

*Machine Learning* aplicado al  
rendimiento académico en educación  
superior: factores, variables y  
herramientas



# *Machine Learning* aplicado al rendimiento académico en educación superior: factores, variables y herramientas

Leonardo Emiro Contreras Bravo,  
Giovanny Mauricio Tarazona Bermúdez, Angie Paola Alemán Cardona,  
Jhonatan Estiven Ruiz Vanegas, Brandon Herney Hernández Arcila,  
Daniela Garzón Trujillo, Óscar Eduardo Sánchez Gómez,  
Marisol Ángel Cifuentes





**UD**  
Editorial

**E2**  
ESPACIOS

© Universidad Distrital Francisco José de Caldas  
© Centro de Investigaciones y Desarrollo Científico (CIDC)  
© Leonardo Emiro Contreras Bravo, Giovanny Mauricio Tarazona Bermúdez, Angie Paola Alemán Cardona, Jhonatan Estiven Ruiz Vanegas, Brandon Herney Hernández Arcila, Daniela Garzón Trujillo, Óscar Eduardo Sánchez Gómez, Marisol Ángel Cifuentes

**ISBN:** 978-958-787-516-4

**ISBN digital:** 978-958-787-517-1

**ISBN ePub:** 978-958-787-518-8

Primera edición, abril de 2023

**Jefe Sección de Publicaciones**

Rubén Eliécer Carvajalino C.

**Gestión editorial**

Rosa Isabel González Moreno

**Corrección de estilo**

Proceditor

**Diagramación**

Ginna Catalina Osorio Mojica

**Editorial UD**

Universidad Distrital Francisco José de Caldas

Carrera 24 n°. 34-37

Teléfono: (601) 3239300 ext. 6202

Correo electrónico: publicaciones@udistrital.edu.co

*Sistema de Bibliotecas de la Universidad Distrital  
Francisco José de Caldas*

*Catalogación en la publicación (CEP)*

Machine learning aplicado al rendimiento académico en la educación superior : factores, variables y herramientas / Leonardo Emiro Contreras Bravo ... [y otros 7 autores]. -- 1a. ed. -- Bogotá : Universidad Distrital Francisco José de Caldas, 2023.

316 páginas; 24 cm. (Colección Espacios)

ISBN: 978-958-787-516-4 ISBN digital: 978-958-787-517-1

ISBN ePub: 978-958-787-518-8

1. Aprendizaje supervisado (Aprendizaje automático) 2. Rendimiento académico -- Mediciones 3. Procesamiento electrónico de datos en la educación 4. Educación superior -- Innovaciones tecnológicas.

I. Contreras Bravo, Leonardo Emiro, autor II. Tarazona Bermúdez, Giovanny Mauricio, autor III. Alemán Cardona, Angie Paola, autora IV. Ruiz Vanegas, Jhonatan Estiven, autor V. Hernández, Arcila Brandon Herney, autor VI. Garzón Trujillo, Daniela, autora VII. Sánchez Gómez, Óscar Eduardo, autor VIII. Ángel Cifuentes, Marisol, autora. Serie.

006.31: CDD 21 edición.

**Todos los derechos reservados.**

Esta obra no puede ser reproducida sin el permiso previo escrito de la Sección de Publicaciones de la Universidad Distrital Francisco José de Caldas. Hecho en Colombia



# Contenido

|                                                                             |           |
|-----------------------------------------------------------------------------|-----------|
| <b>Capítulo 1</b>                                                           |           |
| <b>Contextualización del tema de investigación y su desarrollo</b>          | <b>21</b> |
| 1.1. Introducción                                                           | 21        |
| 1.2. Formulación del problema                                               | 24        |
| 1.3. Justificación del problema                                             | 27        |
| 1.4. Objetivos de la investigación                                          | 30        |
| 1.4.1. Objetivo general                                                     | 30        |
| 1.4.2. Objetivos específicos                                                | 30        |
| 1.5. Metodología                                                            | 31        |
| 1.5.1. Fase I. Fuentes de información                                       | 32        |
| 1.5.2. Fase II. Planificación de la revisión de literatura                  | 32        |
| 1.5.3. Fase III. Criterios de selección de la información                   | 35        |
| 1.5.4. Fase IV. Análisis e interpretación de datos relevantes               | 36        |
| 1.5.5. Fase V. Resultados                                                   | 37        |
| <b>Capítulo 2</b>                                                           |           |
| <b>Rendimiento académico</b>                                                | <b>39</b> |
| 2.1. Generalidades del rendimiento académico                                | 39        |
| 2.2. Panorama de posibles variables que predicen el rendimiento académico   | 41        |
| 2.3. Indicadores del rendimiento académico: calificaciones                  | 44        |
| 2.4. Indicadores del rendimiento académico: pruebas objetivas               | 45        |
| 2.5. Indicadores del rendimiento académico: número de asignaturas aprobadas | 45        |
| 2.6. Indicadores del rendimiento académico: créditos acumulados             | 46        |

|                                                                                                  |            |
|--------------------------------------------------------------------------------------------------|------------|
| <b>Capítulo 3</b>                                                                                |            |
| <b><i>Machine Learning</i></b>                                                                   | <b>47</b>  |
| 3.1. Generalidades del <i>Machine Learning</i>                                                   | 47         |
| 3.2. Clasificación del <i>Machine Learning</i>                                                   | 48         |
| 3.3. Proceso de <i>Machine Learning</i>                                                          | 53         |
| <b>Capítulo 4</b>                                                                                |            |
| <b>Aprendizaje no supervisado: factores y variables que influyen en el rendimiento académico</b> | <b>57</b>  |
| 4.1. Factor sociodemográfico                                                                     | 59         |
| 4.2. Factor de aprendizaje                                                                       | 61         |
| 4.3. Factor académico                                                                            | 62         |
| 4.4. Factor entorno académico                                                                    | 63         |
| 4.5. Factor psicológico                                                                          | 64         |
| 4.6. Factor de gestión académica                                                                 | 64         |
| <b>Capítulo 5</b>                                                                                |            |
| <b>Aprendizaje no supervisado: investigación referencial</b>                                     | <b>69</b>  |
| 5.1. Investigaciones realizadas entre 2010 y 2013                                                | 70         |
| 5.2. Investigaciones realizadas entre 2014 y 2017                                                | 76         |
| 5.3. Investigaciones realizadas entre 2018 y 2020                                                | 93         |
| 5.4. Análisis de las producciones académicas relacionadas con aprendizaje no supervisado         | 114        |
| <b>Capítulo 6</b>                                                                                |            |
| <b>Aprendizaje no supervisado: algoritmos implementados</b>                                      | <b>117</b> |
| 6.1. <i>K-Means</i>                                                                              | 117        |
| 6.2. <i>K-Medoides</i>                                                                           | 120        |
| 6.3. <i>Fuzzy C-Means</i>                                                                        | 122        |
| 6.4. <i>Hierarchical Clustering</i> (HCM/HCA)                                                    | 122        |
| 6.5. <i>Latent Dirichlet Allocation</i> (LDA)                                                    | 124        |
| 6.6. Análisis de componentes principales (PCA)                                                   | 126        |
| 6.7. <i>Gaussian Mixture Model</i> (GMM)                                                         | 127        |
| 6.8. <i>Hidden Markov Model</i> (HMM)                                                            | 128        |
| 6.9. <i>Apriori</i>                                                                              | 129        |
| 6.10. Escalamiento multidimensional (MDS)                                                        | 129        |
| 6.11. <i>Expectation Maximization</i> (EM)                                                       | 130        |
| 6.12. <i>Self-Organization Maps</i> (SOM)                                                        | 132        |

|                                                          |     |
|----------------------------------------------------------|-----|
| 6.13. <i>Agglomerative Hierarchical Clustering</i> (AHC) | 134 |
| 6.14. <i>Temporal Data Clustering</i>                    | 136 |
| 6.15. <i>Relational Association Rules</i> (RAR)          | 137 |
| 6.16. <i>Farthest-First</i>                              | 138 |
| 6.17. <i>Fuzzy K-Means</i>                               | 139 |
| 6.18. <i>K-Modes</i>                                     | 140 |
| 6.19. <i>Kohonen Self-Organizing Map</i>                 | 141 |

## Capítulo 7

### Aprendizaje supervisado: Factores y variables que influyen en el rendimiento académico

|                                     |     |
|-------------------------------------|-----|
| 7.1. Factor académico               | 147 |
| 7.2. Factor sociodemográfico        | 147 |
| 7.3. Factor de aprendizaje en línea | 150 |
| 7.4. Factor de gestión académica    | 150 |
| 7.5. Factor psicosocial             | 152 |
| 7.6. Factor de entorno académico    | 153 |

## Capítulo 8

### Aprendizaje supervisado: Investigación referencial

|                                                                                       |     |
|---------------------------------------------------------------------------------------|-----|
| 8.1. Periodo 2010-2013                                                                | 156 |
| 8.2. Periodo 2014-2017                                                                | 168 |
| 8.3. Periodo 2018-2020                                                                | 182 |
| 8.4. Análisis de las producciones académicas relacionadas con aprendizaje supervisado | 206 |

## Capítulo 9

### Aprendizaje supervisado: Algoritmos implementados

|                                                    |     |
|----------------------------------------------------|-----|
| 9.1. K-vecinos más cercanos (KNN)                  | 213 |
| 9.2. Naive Bayes                                   | 213 |
| 9.3. Máquina de vectores de soporte (SVM)          | 213 |
| 9.4. Árboles de decisión ( <i>decision trees</i> ) | 214 |
| 9.5. Regresión logística                           | 215 |
| 9.6. Algoritmos de regresión                       | 215 |
| 9.7. Regresión lineal                              | 216 |
| 9.8. Regresión polinomial                          | 219 |

|                                                                                                                                            |            |
|--------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Capítulo 10</b>                                                                                                                         |            |
| <b>Redes Neuronales: factores y variables que influyen en el rendimiento académico</b>                                                     | <b>221</b> |
| 10.1. Factor demográfico, económico y social                                                                                               | 225        |
| 10.2. Factor psicológico                                                                                                                   | 226        |
| 10.3. Factor entorno académico                                                                                                             | 226        |
| 10.4. Factor gestión académica                                                                                                             | 227        |
| 10.5. Factor desempeño académico universitario y preuniversitario                                                                          | 228        |
| 10.6. Factor <i>E-Learning</i>                                                                                                             | 229        |
| <b>Capítulo 11</b>                                                                                                                         |            |
| <b>Redes neuronales: Investigación referencial</b>                                                                                         | <b>231</b> |
| 11.1. Investigaciones realizadas entre 2009 y 2013                                                                                         | 232        |
| 11.2. Investigaciones realizadas entre 2014 y 2017                                                                                         | 233        |
| 11.3. Investigaciones realizadas entre 2018 y 2020                                                                                         | 237        |
| 11.4. Análisis de las producciones académicas relacionadas con redes neuronales                                                            | 257        |
| <b>Capítulo 12</b>                                                                                                                         |            |
| <b>Redes neuronales: algoritmos implementados</b>                                                                                          | <b>261</b> |
| 12.1. Red Neuronal Perceptrón Multicapa (MLP)                                                                                              | 262        |
| 12.2. Red Neuronal de Realimentación (FNN)                                                                                                 | 265        |
| 12.3. Red Neuronal Convolucional (CNN)                                                                                                     | 266        |
| 12.4. Red Neuronal Recurrente (RNN)                                                                                                        | 267        |
| 12.5. <i>Bidirectional Long Short-Term Memory</i> (BLSTM)                                                                                  | 267        |
| <b>Capítulo 13</b>                                                                                                                         |            |
| <b>Análisis de <i>software</i> usados/relacionados con <i>Machine Learning</i> aplicado al rendimiento académico en educación superior</b> | <b>269</b> |
| 13.1. Breve resumen de <i>software</i> o lenguajes de programación                                                                         | 269        |
| 13.2. <i>Software</i> : Aprendizaje no supervisado                                                                                         | 276        |
| 13.3. <i>Software</i> : Aprendizaje profundo                                                                                               | 281        |
| 13.4. <i>Software</i> : Aprendizaje supervisado                                                                                            | 283        |
| <b>Referencias</b>                                                                                                                         | <b>285</b> |
| <b>Autores</b>                                                                                                                             | <b>319</b> |

# Índice de tablas

|                  |                                                                                                             |     |
|------------------|-------------------------------------------------------------------------------------------------------------|-----|
| <b>Tabla 1.</b>  | Ecuaciones de búsqueda según el campo de aplicación de <i>Machine Learning</i>                              | 33  |
| <b>Tabla 2.</b>  | Variables que afectan el rendimiento académico                                                              | 43  |
| <b>Tabla 3.</b>  | Autores consultados que han usado el algoritmo <i>K-Means</i> durante la última década                      | 119 |
| <b>Tabla 4.</b>  | Autores consultados que han usado el algoritmo <i>K-Medoids</i> durante la última década                    | 121 |
| <b>Tabla 5.</b>  | Autores consultados que han usado el algoritmo <i>Fuzzy C-Means</i> durante la última década                | 122 |
| <b>Tabla 6.</b>  | Autores consultados que han usado el algoritmo <i>Hierarchical Clustering</i> durante la última década      | 124 |
| <b>Tabla 7.</b>  | Autores consultados que han usado el algoritmo <i>Latent Dirichlet Allocation</i> durante la última década  | 126 |
| <b>Tabla 8.</b>  | Autores consultados que han usado el algoritmo <i>Principal Component Analysis</i> durante la última década | 127 |
| <b>Tabla 9.</b>  | Autores consultados que han usado el algoritmo <i>Gaussian Mixture Models</i> durante la última década      | 128 |
| <b>Tabla 10.</b> | Autores consultados que han usado el algoritmo <i>Hidden Markov Models</i> durante la última década         | 128 |
| <b>Tabla 11.</b> | Autores consultados que han usado el algoritmo <i>Apriori</i> durante la última década                      | 129 |
| <b>Tabla 12.</b> | Autores consultados que han usado el algoritmo <i>Multidimensional Scaling</i> durante la última década     | 130 |
| <b>Tabla 13.</b> | Autores consultados que han usado el algoritmo EM durante la última década                                  | 132 |

|                  |                                                                                                                      |     |
|------------------|----------------------------------------------------------------------------------------------------------------------|-----|
| <b>Tabla 14.</b> | Autores consultados que han usado el algoritmo <i>Self-Organization Map</i> durante la última década                 | 134 |
| <b>Tabla 15.</b> | Autores consultados que han usado el algoritmo <i>Agglomerative Hierarchical Clustering</i> durante la última década | 135 |
| <b>Tabla 16.</b> | Autores consultados que han usado el algoritmo <i>Temporal Data Clustering</i> durante la última década              | 137 |
| <b>Tabla 17.</b> | Autores consultados que han usado el algoritmo <i>Relational Association Rules</i> durante la última década          | 138 |
| <b>Tabla 18.</b> | Autores consultados que han usado el algoritmo <i>Farthest-First</i> durante la última década                        | 139 |
| <b>Tabla 19.</b> | Autores consultados que han usado el algoritmo <i>Fuzzy K-Means</i> durante la última década                         | 140 |
| <b>Tabla 20.</b> | Autores consultados que han usado el algoritmo <i>K-Modes</i> durante la última década                               | 141 |
| <b>Tabla 21.</b> | Autores consultados que han usado el algoritmo <i>Kohonen Self-Organizing Map</i> durante la última década           | 142 |
| <b>Tabla 22.</b> | Investigación documental periodo 2010-2013                                                                           | 156 |
| <b>Tabla 23.</b> | Resultados de la predicción<br>(Naive Bayes, <i>Support Vector Machine</i> , <i>Decision Tree</i> )                  | 160 |
| <b>Tabla 24.</b> | Resultados de la predicción de los modelos MLR, MLP, RBF y SVM                                                       | 162 |
| <b>Tabla 25.</b> | Resultados de la predicción                                                                                          | 163 |
| <b>Tabla 26.</b> | Exactitud de la predicción                                                                                           | 166 |
| <b>Tabla 27.</b> | Investigación documental periodo 2014-2017                                                                           | 168 |
| <b>Tabla 28.</b> | Comparación de algoritmos                                                                                            | 171 |
| <b>Tabla 29.</b> | Resultados de algoritmos de clasificación                                                                            | 173 |
| <b>Tabla 30.</b> | Precisión de los clasificadores                                                                                      | 174 |
| <b>Tabla 31.</b> | Medidas de precisión y sensibilidad                                                                                  | 175 |
| <b>Tabla 32.</b> | Cursos determinantes por semestre                                                                                    | 178 |
| <b>Tabla 33.</b> | Resultados de los modelos aplicados                                                                                  | 178 |

|                  |                                                                               |     |
|------------------|-------------------------------------------------------------------------------|-----|
| <b>Tabla 34.</b> | Tasas de precisión                                                            | 179 |
| <b>Tabla 35.</b> | Investigación documental periodo 2018-2020                                    | 182 |
| <b>Tabla 36.</b> | Resultado del método de clasificación                                         | 192 |
| <b>Tabla 37.</b> | Precisión de modelos                                                          | 193 |
| <b>Tabla 38.</b> | Comparación de algoritmos de aprendizaje automático                           | 200 |
| <b>Tabla 39.</b> | Resultado de validación de los métodos de clasificación                       | 202 |
| <b>Tabla 40.</b> | Resultados del modelo                                                         | 204 |
| <b>Tabla 41.</b> | Validación cruzada con los mejores atributos                                  | 206 |
| <b>Tabla 42.</b> | Algunas arquitecturas de DL                                                   | 262 |
| <b>Tabla 43.</b> | Autores consultados que han usado el algoritmo MLP durante la última década   | 265 |
| <b>Tabla 44.</b> | Autores consultados que han usado el algoritmo FNN durante la última década   | 266 |
| <b>Tabla 45.</b> | Autores consultados que han usado el algoritmo CNN durante la última década   | 266 |
| <b>Tabla 46.</b> | Autores consultados que han usado el algoritmo RNN durante la última década   | 267 |
| <b>Tabla 47.</b> | Autores consultados que han usado el algoritmo BLSTM durante la última década | 268 |
| <b>Tabla 48.</b> | Fuentes bibliográficas que usaron Python                                      | 277 |
| <b>Tabla 49.</b> | Referencias bibliográficas que utilizaron WEKA                                | 277 |
| <b>Tabla 50.</b> | Referencias bibliográficas que utilizaron R                                   | 278 |
| <b>Tabla 51.</b> | Referencias bibliográficas que utilizaron Java                                | 278 |
| <b>Tabla 52.</b> | Referencias bibliográficas que usaron SQL                                     | 279 |
| <b>Tabla 53.</b> | Referencias bibliográficas que usaron MatLab                                  | 279 |
| <b>Tabla 54.</b> | Referencias bibliográficas que usaron C programming                           | 280 |
| <b>Tabla 55.</b> | Otras herramientas de análisis de datos                                       | 280 |
| <b>Tabla 56.</b> | Herramientas tecnológicas                                                     | 282 |





# Índice de figuras

|                   |                                                                                          |    |
|-------------------|------------------------------------------------------------------------------------------|----|
| <b>Figura 1.</b>  | Tendencia en herramientas de análisis de datos                                           | 22 |
| <b>Figura 2.</b>  | Clasificación por sector de los estudiantes de IES                                       | 24 |
| <b>Figura 3.</b>  | Estadísticas de deserción y graduación según el MEN y SPADIES 3.0, Colombia y SNIES 2019 | 25 |
| <b>Figura 4.</b>  | Tasa de graduación, jóvenes de entre los 25 y los 29 años, América Latina y el Caribe    | 26 |
| <b>Figura 5.</b>  | Propuesta de analítica en educación superior                                             | 29 |
| <b>Figura 6.</b>  | Metodología de la investigación. Diagrama de fases                                       | 31 |
| <b>Figura 7.</b>  | Rendimiento académico. Definiciones y actores                                            | 41 |
| <b>Figura 8.</b>  | Clasificación del <i>Machine Learning</i> según su naturaleza                            | 49 |
| <b>Figura 9.</b>  | Campo de las redes neuronales en <i>Machine Learning</i>                                 | 51 |
| <b>Figura 10.</b> | Fases del proceso de <i>Machine Learning</i>                                             | 53 |
| <b>Figura 11.</b> | Agrupación de referencias bibliográficas por año                                         | 58 |
| <b>Figura 12.</b> | Clasificación porcentual de los campos de estudio en los artículos referenciados         | 58 |
| <b>Figura 13.</b> | Variables influyentes en el rendimiento                                                  | 60 |
| <b>Figura 14.</b> | Investigaciones que consideran variables del factor sociodemográfico                     | 60 |
| <b>Figura 15.</b> | Investigaciones que consideran variables del factor de aprendizaje                       | 61 |
| <b>Figura 16.</b> | Investigaciones que consideran variables del factor académico                            | 62 |
| <b>Figura 17.</b> | Investigaciones que consideran variables del factor académico II                         | 63 |

|                   |                                                                                                                                                                                                                                                    |     |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Figura 18.</b> | Investigaciones que consideran variables del factor psicológico                                                                                                                                                                                    | 65  |
| <b>Figura 19.</b> | Investigaciones que consideran variables del factor de gestión académica                                                                                                                                                                           | 65  |
| <b>Figura 20.</b> | Agrupación de artículos por factor                                                                                                                                                                                                                 | 66  |
| <b>Figura 21.</b> | Nube de variables influyentes en el rendimiento académico                                                                                                                                                                                          | 67  |
| <b>Figura 22.</b> | Investigaciones consultadas del campo educación por año de publicación                                                                                                                                                                             | 69  |
| <b>Figura 23.</b> | Características del estilo de aprendizaje de Kolb                                                                                                                                                                                                  | 81  |
| <b>Figura 24.</b> | Rango de tiempo de graduación para el resultado de predicción en cada clúster                                                                                                                                                                      | 85  |
| <b>Figura 25.</b> | Distribución de los resultados finales de los estudiantes                                                                                                                                                                                          | 87  |
| <b>Figura 26.</b> | Esquema propuesto para la evaluación automática de la metacognición del estudiante en el ambiente del aula                                                                                                                                         | 87  |
| <b>Figura 27.</b> | Resultado de las reglas de asociación de la minería evaluada en los diferentes cursos ofertados                                                                                                                                                    | 90  |
| <b>Figura 28.</b> | Comparación de los dos enfoques                                                                                                                                                                                                                    | 91  |
| <b>Figura 29.</b> | Porcentaje de diversos factores y métodos utilizados en el análisis del rendimiento de los estudiantes y porcentaje de estudios de predicción del desempeño según su objetivo y hora de predicción                                                 | 109 |
| <b>Figura 30.</b> | La arquitectura de clasificación CB                                                                                                                                                                                                                | 114 |
| <b>Figura 31.</b> | Investigación referencial y su representación en el mundo                                                                                                                                                                                          | 115 |
| <b>Figura 32.</b> | Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado                                                                                                                                                                       | 116 |
| <b>Figura 33.</b> | Técnicas de UML utilizadas por las investigaciones consultadas                                                                                                                                                                                     | 118 |
| <b>Figura 34.</b> | Algoritmo <i>K-Means</i>                                                                                                                                                                                                                           | 119 |
| <b>Figura 35.</b> | Diferencia entre <i>Mean</i> y <i>Medoid</i>                                                                                                                                                                                                       | 121 |
| <b>Figura 36.</b> | Modelo de agrupación jerárquica                                                                                                                                                                                                                    | 123 |
| <b>Figura 37.</b> | Representación gráfica del modelo de LDA. Las cajas son “placas” que representan réplicas. La placa exterior representa los documentos, mientras que la placa interior representa la selección repetida de temas y palabras dentro de un documento | 125 |

|                           |                                                                                                                                                                                                                                                       |     |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Figura 38.</b>         | Ejemplo EM: componentes y datos de la mezcla. Los datos constan de tres muestras extraídas de cada componente de la mezcla, que se muestran arriba como círculos y triángulos. Las medias de los componentes de la mezcla son -2 y 2, respectivamente | 131 |
| <b>Figura 39.</b>         | Ejemplo del entrenamiento de un mapa auto organizado. La zona azul es la distribución de los datos de entrenamiento, y el disco blanco es el ejemplo de entrenamiento actual para esa distribución                                                    | 132 |
| <b>Figura 40.</b>         | Ejemplo de agrupación jerárquica aglomerativa (AHC) usando el <i>software</i> estadístico XLSTAT en Excel                                                                                                                                             | 136 |
| <b>Figura 41.</b>         | Ejemplo <i>Farthest-First</i> . Los primeros cinco pasos del primer recorrido más lejano de un conjunto de puntos planos                                                                                                                              | 139 |
| <b>Figura 42.</b>         | (Izq.) Ejemplificación Kohonen SOM en 1D. (Der.) Ejemplificación Kohonen SOM en 2D                                                                                                                                                                    | 143 |
| <b>Figura 43.</b>         | Agrupación de referencias bibliográficas por año                                                                                                                                                                                                      | 146 |
| <b>Figura 44.</b>         | Número de artículos que tienen en cuenta factores que influyen en el rendimiento académico                                                                                                                                                            | 146 |
| <b>Figura 45.</b>         | Investigaciones que consideran variables del factor académico                                                                                                                                                                                         | 148 |
| <b>Figura 46.</b>         | Investigaciones que consideran variables del factor sociodemográfico                                                                                                                                                                                  | 149 |
| <b>Figura 47.</b>         | Investigaciones que consideran variables del factor de aprendizaje en línea                                                                                                                                                                           | 150 |
| <b>Figura 48.</b>         | Investigaciones que consideran variables del factor de gestión académica                                                                                                                                                                              | 151 |
| <b>Figura 49.</b>         | Investigaciones que consideran variables del factor psicosocial                                                                                                                                                                                       | 152 |
| <b>Figura 50.</b>         | Investigaciones que consideran variables del factor de entorno académico                                                                                                                                                                              | 153 |
| <b>Figura 51.</b>         | Estados del arte por año                                                                                                                                                                                                                              | 155 |
| <b>Figura 52.</b>         | Proceso de minería de datos y validación cruzada                                                                                                                                                                                                      | 159 |
| <b>Figura 53.</b>         | Exactitud de la predicción                                                                                                                                                                                                                            | 165 |
| <b>Figura 54.</b>         | Comparación de algoritmos                                                                                                                                                                                                                             | 176 |
| <b>Figura 55.</b>         | Comparación de la precisión de los algoritmos                                                                                                                                                                                                         | 195 |
| <b>Figuras 56a y 56b.</b> | Proceso de predicción de deserción universitaria.<br>Precisión de modelos de predicción                                                                                                                                                               | 196 |
| <b>Figura 57.</b>         | Precisión de los modelos de predicción                                                                                                                                                                                                                | 197 |

|                   |                                                                                          |     |
|-------------------|------------------------------------------------------------------------------------------|-----|
| <b>Figura 58.</b> | Comparación de categorías                                                                | 203 |
| <b>Figura 59.</b> | Investigación referencial y su representación en el mundo                                | 207 |
| <b>Figura 60.</b> | Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado             | 207 |
| <b>Figura 61.</b> | Investigaciones que tuvieron como mejor algoritmo al árbol de decisión                   | 210 |
| <b>Figura 62.</b> | Investigaciones que tuvieron como mejor algoritmo a Naive Bayes                          | 210 |
| <b>Figura 63.</b> | Investigaciones que tuvieron como mejor algoritmo a regresión logística                  | 211 |
| <b>Figura 64.</b> | Investigaciones que tuvieron como mejor algoritmo a K-vecino más cercano                 | 211 |
| <b>Figura 65.</b> | Investigaciones que tuvieron como mejor algoritmo a máquina de vectores de soporte (SVM) | 212 |
| <b>Figura 66.</b> | Algoritmos de regresión                                                                  | 216 |
| <b>Figura 67.</b> | Regresión lineal                                                                         | 216 |
| <b>Figura 68.</b> | Ejemplo de regresión polinomial para conjunto de datos                                   | 220 |
| <b>Figura 69.</b> | Agrupación de referencias bibliográficas por año                                         | 221 |
| <b>Figura 70.</b> | Cantidad de artículos encontrados en cada motor de búsqueda                              | 222 |
| <b>Figura 71.</b> | Nube de palabras variables demográficas, económicas y sociales                           | 225 |
| <b>Figura 72.</b> | Investigaciones que consideran variables del Factor psicológico                          | 226 |
| <b>Figura 73.</b> | Nube de palabras variables de entorno académico                                          | 227 |
| <b>Figura 74.</b> | Nube de palabras variables gestión académica                                             | 228 |
| <b>Figura 75.</b> | Nube de palabras variables desempeño académico preuniversitario y universitario          | 229 |
| <b>Figura 76.</b> | Nube de palabras variables <i>E-Learning</i>                                             | 230 |
| <b>Figura 77.</b> | Gráfico de estados del arte por año                                                      | 231 |
| <b>Figura 78.</b> | Investigación referencial y su representación en el mundo                                | 258 |
| <b>Figura 79.</b> | Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado             | 259 |
| <b>Figura 80.</b> | Clasificación según enfoque                                                              | 259 |

|                   |                                                                                           |     |
|-------------------|-------------------------------------------------------------------------------------------|-----|
| <b>Figura 81.</b> | Algoritmos de aprendizaje profundo (redes) utilizadas por las investigaciones consultadas | 262 |
| <b>Figura 82.</b> | El perceptrón multicapa con una sola capa oculta                                          | 263 |
| <b>Figura 83.</b> | Ejemplo de perceptrón                                                                     | 264 |
| <b>Figura 84.</b> | Herramientas tecnológicas usadas                                                          | 281 |
| <b>Figura 85.</b> | Herramientas tecnológicas usadas relacionadas con algoritmos de aprendizaje profundo      | 282 |
| <b>Figura 86.</b> | Frecuencia de utilización de herramientas de análisis                                     | 283 |



# Capítulo 1

## Contextualización del tema de investigación y su desarrollo

### 1.1. Introducción

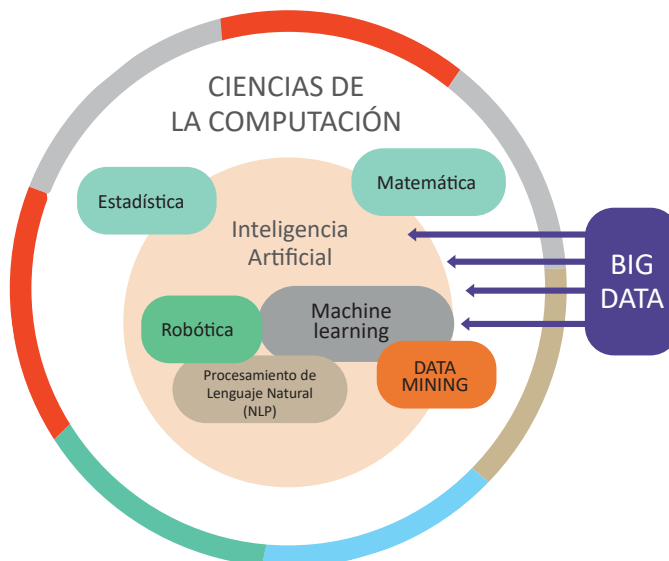
En la coyuntura mundial actual, los estudios del rendimiento académico en la educación superior parecen ser uno de los aspectos más valiosos de estudio debido al dinamismo que experimenta el sector universitario en el marco de una sociedad caracterizada por el rápido avance del conocimiento, la fluidez en la transmisión de la información y los acelerados cambios en las estructuras sociales. En ese contexto, la calificación del capital humano o rendimiento académico adquiere mayor valor, ya que suele ser el principal indicador empleado para medir el éxito o el fracaso de un estudiante, y la eficiencia del proceso educativo de las instituciones de educación superior.

En lo que a calidad de educación superior se refiere, y de acuerdo con los distintos cuestionamientos que se hacen al sector público en cuanto a la relación costo-beneficio en ámbitos sociales, estos estudios han despertado en las instituciones universitarias un interés particular por los resultados académicos de sus estudiantes. Estos resultados comprenden un comportamiento multidimensional: son afectados por múltiples variables sociales e inherentes, tales como los objetivos de la institución, los objetivos particulares de los docentes, el clima social y familiar, etc. Este tipo de estudios y análisis constituyen herramientas sólidas para construir indicadores que orienten la toma de decisiones en educación superior, además de técnicas y metodologías que ofrezcan un alto nivel de exactitud para predecir el rendimiento académico.

*Big Data* (gran volumen de datos) se ha vuelto tendencia a través de los años debido a los métodos usados para analizar los datos: inicialmente los métodos eran puramente estadísticos, pero posteriormente se usaron los algoritmos de *Data Mining* (minería de datos) para entender otras dimensiones de los fenómenos estudiados. Estos estudios se vuelven más complejos con el avance de la ciencia y de la computación y la aparición de la inteligencia artificial (figura 1), el hecho de que una máquina pueda imitar o realizar acciones complejas como las que realizan los humanos. Se generan así tres grandes campos: la robótica (movimientos humanos), el procesamiento por aprendizaje natural (texto y voz) y el *Machine Learning* (ML) (datos).

Las técnicas relacionadas con el *Machine Learning* han tenido su propio auge en los proyectos de investigación debido a su estrecha relación con la inteligencia artificial. En el ámbito académico, existen múltiples referencias bibliográficas que miden el rendimiento académico, principalmente en instituciones de educación superior, y en las cuales se hace uso de diversos algoritmos de técnicas de aprendizaje supervisado y no supervisado. Estos algoritmos permiten obtener información relevante sin utilizar códigos de programación específicos.

**Figura 1.** Tendencia en herramientas de análisis de datos



**Fuente:** elaboración propia.



Hay coincidencias importantes en el estado del arte de múltiples trabajos de investigación que utilizan las técnicas de *Machine Learning*: por ejemplo, en ellos se prioriza la identificación de variables clave con poder de predicción elevado para medir el rendimiento académico de un estudiante y de un correcto ajuste del conjunto de datos que se trabaje. Así mismo, la utilización de técnicas combinadas presenta un alto porcentaje de exactitud y de visualización de variables secundarias que pueden afectar el rendimiento académico.

El *Machine Learning* comprende diversos campos de trabajo: algoritmos supervisados, no supervisados y aprendizaje profundo (redes neuronales). Este último campo, junto con los algoritmos de ensamble o de conjunto, son en los que más existen avances tecnológicos recientemente en cuanto a algoritmos, con el fin de solucionar problemas reales necesarios. El modelo de red neuronal se basa en un conjunto de unidades neuronales simples (neuronas artificiales), que se comportan de forma aproximadamente análoga a los axones de las neuronas en los cerebros biológicos: una neurona se conecta con otras anteriores y posteriores, y estas con muchas otras en diferentes capas. Los enlaces entre las neuronas pueden incrementar o inhibir el estado de activación de las neuronas adyacentes.

De acuerdo con la revisión referencial, se puede afirmar que no se trata de un campo ampliamente aplicado o estudiado en las instituciones de educación superior de América Latina, y los modelos encontrados toman variables distintas: mientras unos utilizan únicamente notas parciales, otros utilizaron conocimientos previos de matemáticas e incluso factores demográficos. Así, es posible crear y aplicar múltiples modelos de *Machine Learning* que tengan en cuenta la mayor cantidad de variables posibles para afectar el rendimiento académico de un estudiante de educación superior (ingeniería). Con esta herramienta es posible tener bases para la toma de decisiones que permitan mejorar la calidad de prestación del servicio educativo y rendimiento de todos los estudiantes.

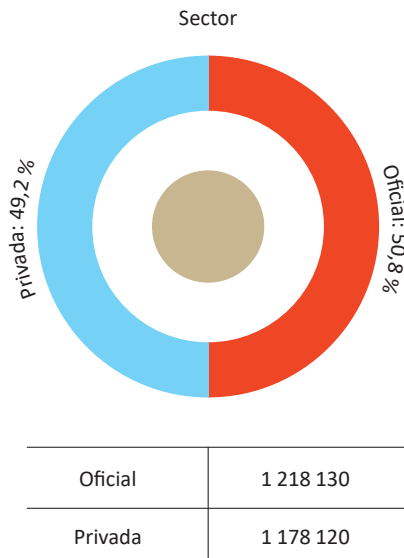
El presente documento de investigación tiene como objetivo principal proporcionar una visión general sobre cómo el *Machine Learning* puede usarse para predecir el rendimiento académico de los estudiantes en educación superior. El documento tiene como base la revisión de la literatura para

identificar las principales variables que afectan el rendimiento académico. Se parte de variables comunes como nivel socioeconómico, el tipo de educación (privada y secundaria), la ciudad de procedencia, el tiempo de desplazamiento desde el hogar hasta la institución de estudio, entre otras; así mismo, se usan los algoritmos más adecuados de aprendizaje no supervisado.

## 1.2. Formulación del problema

En Colombia existen un total de 2 396 259 estudiantes en educación superior (en 2019), de los cuales 1 552 078 (64,77 %) estudian en universidades. Del total de estudiantes en IES, el 50,8 % se encuentra en universidades públicas, y el 49,2 % en universidades privadas (figura 2).

Figura 2. Clasificación por sector de los estudiantes de IES

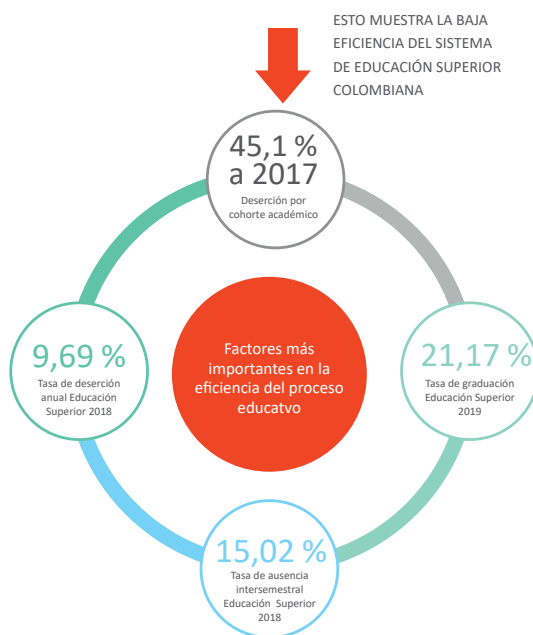


Fuente: Ministerio de Educación (2019a).

La eficiencia del proceso educativo puede ser medido con ciertas métricas: la tasa de deserción anual (conjunto de estudiantes que después de estar matriculados en dos semestres previos se identifican, un año después, como desertores); la tasa de deserción por cohorte (el conteo acumulado de la deserción académica desde primer semestre —, es decir, el momento en el que todos los estudiantes ingresan a la institución al mismo tiempo); la tasa de

graduación (se calcula un fraccionario compuesto por el número total de graduados, y el total de estudiantes que ingresaron en primer semestre); y la tasa de ausencia intersemestral. De acuerdo con el MEN, estas métricas no son las más precisas en Colombia debido a que en el 2017 la tasa de deserción por cohorte fue del 45,1 %. En otras palabras, de un total de 100 estudiantes en una cohorte, casi la mitad de ellos desertó; así mismo, encontramos que una tasa de graduación del 21,17% para 2019: es decir que el porcentaje de estudiantes que ingresa a la educación superior se gradúa en un tiempo superior a los diez semestres académicos (figura 3). Ahora bien, estos indicadores son preocupantes, y se debe decir, además, que en una situación similar, o quizás peor, se encuentran muchos de los países latinoamericanos (Ferreira *et al.*, 2017)

**Figura 3.** Estadísticas de deserción y graduación según el MEN y SPADIES 3.0, Colombia y SNIES 2019

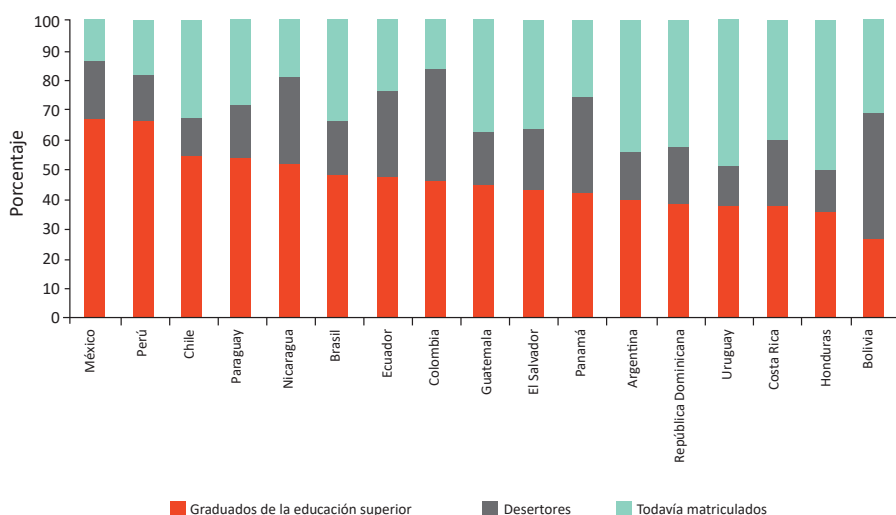


**Fuente:** elaboración propia, adaptado de Ministerio de Educación (2019a, 2019b).

A juzgar por los resultados, el desempeño del sistema es decepcionante. En promedio, alrededor de la mitad de la población de entre los 25 y los 29 años emprendieron un proyecto en la educación superior en algún momento, pero

no finalizaron sus estudios, sea porque aún están estudiando o porque desertaron (figura 4). Según Ferreyra *et al.* (2017), solo en México y Perú, la tasa de graduación está cerca de la estadounidense (que es del 65 %). En los países mostrados, el tiempo transcurrido desde el primer semestre hasta la graduación es, en promedio, un 36 % más de lo requerido. En otros países, algunos estudiantes tardan el doble en graduarse de lo que se supone que deberían. Esto lleva a pensar que en los países de América Latina y el Caribe el tiempo de permanencia de los estudiantes en educación superior es mayor que la de los estudiantes en programas estadounidenses; por tanto, se enfrentan a un costo de oportunidad mayor en términos de ingresos no percibidos.

**Figura 4.** Tasa de graduación, jóvenes de entre los 25 y los 29 años, América Latina y el Caribe



**Fuente:** Ferreyra *et al.* (2017).

Así pues, diferentes autores como García (2016) manifiestan que existe cierta relación entre los indicadores mencionados y el rendimiento académico, y que la eficiencia del sistema de educación superior puede verse mejorada si se toman acciones en cuanto al rendimiento académico (indicador de éxito o fracaso del estudiante), que comprende diferentes variables de tipo cualitativo y cuantitativo.

Con respecto al tema indicadores, en el entorno de la educación en Colombia, en los últimos años, han incrementado los niveles de deserción

por parte de los estudiantes de cursos de educación superior (Guzmán y Durán, 2009). Tanto así que se ha iniciado el estudio de esta problemática desde diferentes universidades como la de Antioquia y la de los Andes. Estas instituciones han tomado como referencia la información académica del Sistema de Información SPADIES (datos abiertos) que permite establecer diferentes categorías para el estudio de la deserción estudiantil.

Por otro lado, las universidades cuentan con una gran cantidad de *software* y plataformas que almacenan datos de los estudiantes y otros generados durante su estadía en la universidad. Estos datos no reciben ningún tratamiento en la búsqueda del mejoramiento del rendimiento académico. De esta manera, se propone un prototipo que permita usar los datos abiertos, dispuestos por entidades del gobierno y las universidades, aquellos dispuestos en las plataformas estudiantiles de administración/admisión, entre otras, con el fin de tratarlos, analizarlos y visualizarlos para así apoyar la toma de decisiones. De esta forma es posible contribuir a la reutilización de la información para llegar a aplicaciones mucho más ricas que aporten mayor conocimiento o que complementan la información ya existente. El reto está en procesar cantidades enormes de datos y convertirlas en decisiones inteligentes y oportunas (Duan y Xu, 2012).

Ante este panorama, surge la siguiente pregunta: ¿es posible integrar datos académicos provenientes de diferentes actores —estudiantes, docentes e instituciones— en un modelo de *Machine Learning* (aprendizaje automático) que permita generar indicadores o clases que faciliten la toma de decisiones acertadas en cuanto al rendimiento académico?

### 1.3. Justificación del problema

Históricamente, el aprendizaje tradicional ha estado centrado en el docente: los alumnos aprendían interactuando con estos, que se enfocaban en interpretar y explicar contenidos. Con el surgimiento de las TIC en educación, el proceso de aprendizaje-enseñanza se descentralizó. Es decir, el proceso de aprendizaje está ahora enfocado en los estudiantes, pues las TIC permiten múltiples interacciones entre los diferentes agentes involucrados: aprendices, instructores y diseñadores de cursos, tutores, contenidos, interfaces, personal administrativo, código, entornos, etc.

Hoy en día, también se dan transformaciones en áreas como la medicina, la economía, la defensa y el comercio, entre otros, como producto de la incursión de la analítica de datos. Tanto ha sido el auge de la analítica que ha permeado el campo de la educación: es posible hoy por hoy procesar grandes volúmenes de datos que contienen información relacionada con los actores del proceso educativo (institución, docentes y estudiantes). Es en esa relación que la ingeniería puede jugar un papel importante en la solución de múltiples aspectos de índole académico-administrativo como la mejora del aprendizaje, la deserción, el abandono y el rendimiento académico.

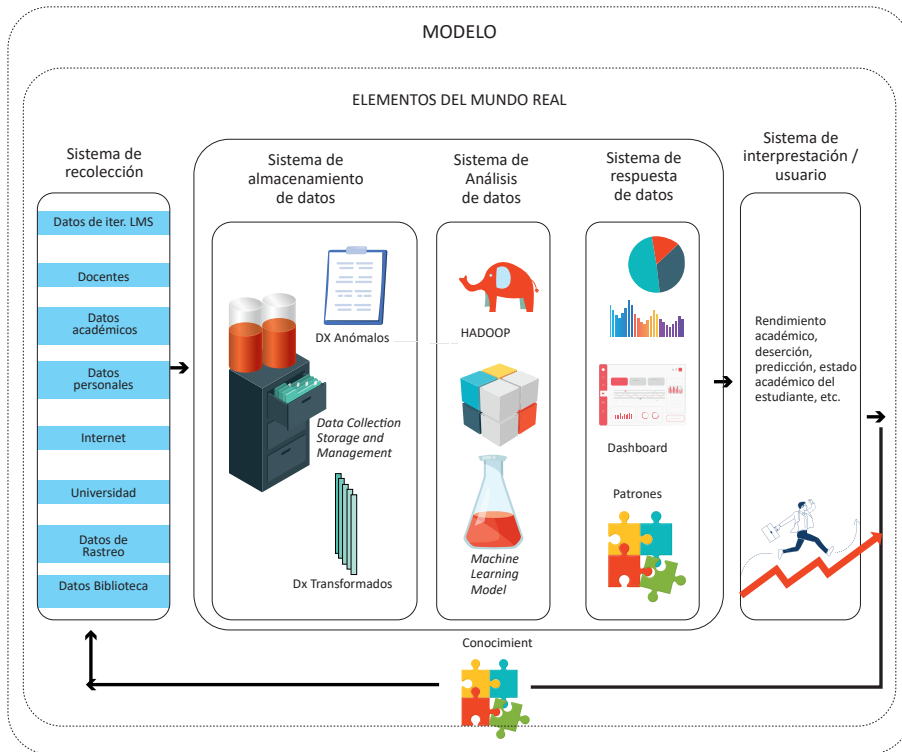
Actualmente, la tendencia en las universidades es trabajar con tecnologías que faciliten y mejoren el desarrollo de las actividades académicas, y la recolección de datos producidos por el estudiante durante el proceso de enseñanza-aprendizaje. Esta información procesada con las herramientas adecuadas permite predecir, prevenir y actuar para mejorar el rendimiento académico y otros aspectos de la vida académica de los estudiantes (Al-Barrak y Al-Razgan, 2016).

En Colombia, la falta de un sistema para analizar y monitorear el progreso del estudiante no es un tema prevalente, como lo debería ser, por dos razones principales: primero, el estudio sobre los métodos de predicción existentes sigue siendo insuficiente para identificar los métodos más adecuados para predecir el rendimiento de los estudiantes; segundo, la falta de investigaciones sobre los factores que afectan los logros de los estudiantes en cursos particulares dentro del contexto de la universidad. Por tanto, se propone una revisión sistemática de la literatura sobre la predicción del rendimiento del estudiante mediante el uso de técnicas de aprendizaje con *Machine Learning* para mejorar los logros de los estudiantes.

Según Lee *et al.* (2006), la analítica de datos se dirige puntualmente a modelos de exploración de datos y herramientas educativas: se trata de una de las técnicas que apoyarán el porvenir de la educación superior y a generar nuevos panoramas y proyectos que mejoren la enseñanza y el aprendizaje de los educandos (figura 5). Por medio de estas herramientas, será posible ayudar a los estudiantes a encontrar qué tan bien lo harán en una determinada asignatura en particular cuando la cursen antes de que se registren en ella, y así evitar problemas posteriores; podría identificarse a los estudiantes en

riesgo o tal vez estimar la cantidad de estudiantes graduados por cohorte y sus futuros o mejores campos de acción al terminar la carrera.

**Figura 5.** Propuesta de analítica en educación superior



**Fuente:** adaptado de Lee *et al.* (2006).

Pese a que el tema del rendimiento académico ya se ha tratado anteriormente, con diferentes técnicas y desde diferentes perspectivas, sigue siendo un campo muy amplio por explorar, pues se pueden generar diferentes modelos que, de resultar efectivos, podrían aportar a la toma eficiente de decisiones orientadas a que el estudiante pueda desarrollar con mayores probabilidades de éxito su ciclo en la universidad. Esto significaría un ahorro de esfuerzos operativos para aquellas personas que contribuyen al proceso de análisis de información para la toma de decisiones dentro de la comunidad académica, lo que a su vez se traduciría en ahorros monetarios e incluso con mejores resultados que los obtenidos con otro tipo de técnicas menos eficientes.

Así pues, este trabajo, al ahondar un poco más sobre la aplicación de herramientas de *Machine Learning* en educación superior, servirá a otros

investigadores, estudiantes, docentes o cualquier persona interesada como material de apoyo para sus futuras indagaciones.

## 1.4. Objetivos de la investigación

A partir del problema identificado en el campo de la educación superior, se plantearon diversos objetivos que permitirán tratar el problema y posiblemente plantear una solución al inconveniente encontrado.

### 1.4.1. Objetivo general

Proporcionar una visión general sobre cómo el *Machine Learning* (aprendizaje automático) puede ser usado para identificar condiciones no satisfactorias y predecir el rendimiento académico de los estudiantes en educación superior.

### 1.4.2. Objetivos específicos

- Identificar las principales variables que afectan el rendimiento académico de los estudiantes en educación superior y a partir de ellas plantear modelos acordes con las condiciones de cada institución.
- Identificar los algoritmos no supervisados de *Machine Learning* existentes para la evaluación del rendimiento académico en educación superior con el fin de poder generar modelos futuros de *clustering*.
- Identificar los algoritmos supervisados y de redes neuronales de *Machine Learning* existentes para la evaluación del rendimiento académico en educación superior, con el fin de poder generar modelos futuros de predicción mediante clases categóricas o numéricas.
- Caracterizar los datos académicos y sus variables claves para la formulación de modelos que faciliten la toma de decisiones derivada del análisis de datos realizado con herramientas de *Machine Learning*.
- Determinar las herramientas de análisis de datos para el cálculo del rendimiento académico, que permitan la viabilidad del uso de los datos académicos para su determinación.

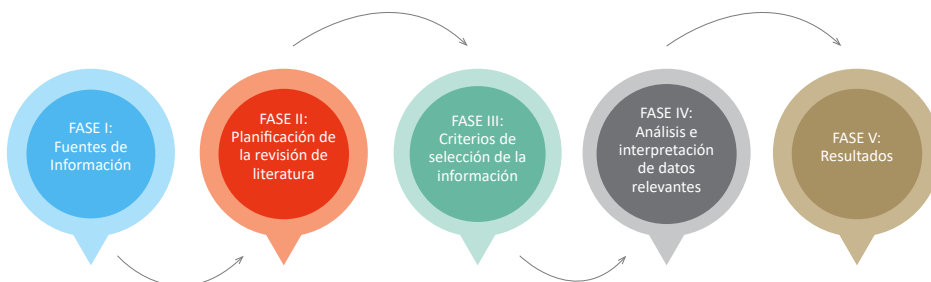


## 1.5. Metodología

Con el fin de llevar a cabo el cumplimiento del objetivo general de esta parte de la investigación, enmarcada dentro del proyecto titulado Integración de datos del ámbito educativo y herramientas de análisis de datos para determinar el rendimiento académico en la educación superior, código 2-15-593-19, financiado por el Centro de Investigaciones y Desarrollo Científico (CIDC) de la Universidad Distrital se planteó una metodología específica enfocada en dos aspectos fundamentales para la elaboración de este documento de investigación: proporcionar una visión general sobre cómo el aprendizaje automático puede usarse para predecir el rendimiento académico de los estudiantes en educación superior, y la búsqueda de teoría sobre los algoritmos de *Machine Learning* aplicados a la predicción del rendimiento académico y a hallazgos recientes sobre este tema.

Para llevar el estudio del estado del arte, y debido a la gran cantidad de investigaciones o trabajos realizados directamente relacionados con el objetivo de esta parte del proyecto, se hace uso del método de revisiones sistemáticas de literatura (RSL). Esta metodología tiene su origen en las revisiones bibliográficas realizadas para ciencias humanas, pero también ha sido utilizada en el campo de la ingeniería (Kitchenham, 2004; Kitchenham y Charters, 2007; Kitchenham *et al.*, 2009). La metodología empleada comprende cinco fases: (1) fuentes de la información, (2) planificación de la revisión de la revisión de literatura (3) criterios de selección de la información, (4) análisis e interpretación de datos relevantes y (5) resultados. Así se muestra en la figura 6.

**Figura 6.** Metodología de la investigación. Diagrama de fases



**Fuente:** elaboración propia.

### 1.5.1. Fase I. Fuentes de información

Las fuentes de información son aquellas en las que se registran contribuciones sobre ciencia, tecnología y educación. Se han incluido tanto bases de datos académicas como motores de búsqueda académicos:

- IEEE: Base de datos relacionada con temas de ciencia e ingeniería.
- Science Direct, Springer Link: Base de datos científica.
- Google Scholar: Buscador de contenido científico académico.
- EBSCO: Base de datos académica sobre ciencia y humanidades.
- ACM Digital Library: Base de datos digital de ACM (Asociación para Maquinaria de Computación). Esta comunidad trabaja temas específicos sobre tecnología e informática.

### 1.5.2. Fase II. Planificación de la revisión de literatura

En esta fase se realiza una revisión de aspectos básicos para tener en cuenta antes de avanzar a las fases de selección de las investigaciones adecuadas para la investigación referencial enfocada en el tema en cuestión.

1. Objetivos de la revisión de literatura: En esta parte del proyecto se pretende el desarrollo de dos objetivos clave del macroproyecto de este trabajo de investigación: identificar la/s teoría/s disponibles acerca del estudio de *Machine Learning* aplicadas al ámbito académico para la predicción del rendimiento académico; y establecer los últimos avances de carácter científico realizados sobre la aplicación de *Machine Learning* al tema en cuestión.
2. Recursos disponibles: Por lo general, el material físico disponible sobre el tema no estuvo al alcance de los investigadores ya que todos los recursos son electrónicos: revistas electrónicas, bases de datos científicas electrónicas, páginas de comunidades que realizan proyectos sobre este (generalmente están ligadas a alguna universidad).
3. Límites temporales: La búsqueda de información contempla el periodo comprendido entre el 2010 y el 2020. Sin embargo, se revisa periódicamente la literatura, para de esta manera tener la información actualizada

y garantizar la calidad del estado del arte. La última revisión se hace el 11 de diciembre del 2020.

4. Definición del protocolo de búsqueda: La búsqueda se llevó a cabo con palabras clave en inglés y español. En español, la búsqueda no es efectiva. Las ecuaciones de búsqueda utilizadas son las que se muestran, dependiendo el campo de *Machine Learning*, en la tabla 1.
5. Consideraciones en la búsqueda: Al momento de realizar la búsqueda de información, todas las palabras clave de las ecuaciones de búsqueda se aplican a todas las fuentes a consultar, aunque a veces la ecuación cambia por el tipo de simbología que se usa en cada motor de búsqueda. Por ejemplo, en lugar de comillas se utilizan paréntesis y símbolos adicionales. Para tener certeza de que la ecuación en el motor de búsqueda utilizado en el momento sea adecuada y la búsqueda sea efectiva, se hace uso de las opciones de búsqueda avanzada.

**Tabla 1.** Ecuaciones de búsqueda según el campo de aplicación de *Machine Learning*

| Campo de aplicación        | Ecuación de búsqueda                                                 |
|----------------------------|----------------------------------------------------------------------|
| Aprendizaje supervisado    | <i>Machine Learning AND Students' Performance</i>                    |
|                            | <i>Educational Data Mining AND Students' Performance</i>             |
|                            | <i>Supervised Machine Learning AND Students' Performance</i>         |
|                            | <i>Prediction of Students Academic Performance</i>                   |
|                            | Modelo predictivo de deserción académica                             |
| Aprendizaje supervisado    | <i>Data Mining Techniques AND Students' Performance</i>              |
|                            | <i>Supervised Learning Algorithms AND Students' Performance</i>      |
|                            | <i>Data Mining for Modeling Students' Performance</i>                |
| Aprendizaje no supervisado | Classification Algorithms                                            |
|                            | <i>Unsupervised Machine Learning AND Academic Performance</i>        |
|                            | <i>"Unsupervised Machine Learning"</i>                               |
|                            | <i>Academic Performance) AND UML – Unsupervised Machine Learning</i> |
|                            | <i>Unsupervised Machine Learning AND Academic OR Performance</i>     |

| Campo de aplicación                     | Ecuación de búsqueda                                        |
|-----------------------------------------|-------------------------------------------------------------|
| Aprendizaje no supervisado              | <i>Higher Education AND Unsupervised Machine Learning</i>   |
|                                         | <i>(Unsupervised) AND (student performance)</i>             |
| Aprendizaje profundo (redes neuronales) | <i>“Deep learning” AND “education” AND “performance”</i>    |
|                                         | <i>“Deep learning” AND “academic performance”</i>           |
|                                         | <i>“deep learning” AND “academic”</i>                       |
|                                         | <i>“neural network” AND “education” AND “performance”</i>   |
|                                         | <i>“neural network” AND “academic performance”</i>          |
|                                         | <i>“neural network” AND “academic”</i>                      |
|                                         | <i>“EDM” AND “prediction”</i>                               |
|                                         | <i>“EDM” AND “academic performance”</i>                     |
|                                         | <i>“Educational Data Mining” AND “prediction”</i>           |
|                                         | <i>“Educational Data Mining” AND “academic performance”</i> |
|                                         | <i>“Learning Analytics” AND “prediction”</i>                |
|                                         | <i>“Learning Analytics” AND “academic performance”</i>      |
|                                         | <i>“Machine Learning” AND “academic performance”</i>        |

**Fuente:** elaboración propia.

Por otro lado, vale la pena aclarar dos términos que podrían generar confusión y que fueron incluidos en la búsqueda de información: el término EDM (*Educational Data Mining*) busca explorar de manera automática grandes bases de datos para lograr detectar patrones o comportamientos en los estudiantes, y el término LA (*learning analytics*) que se basa en el análisis de bases de datos académicas cuya finalidad es tomar decisiones enfocadas en mejorar la calidad del proceso educativo y aumentar las probabilidades de que un estudiante desarrolle con éxito todo el programa estudiantil en el que se encuentra. En este orden de ideas, EDM es parte de LA y, por tanto, cualquier trabajo que se enfoque, bien sea en EDM o en LA, sirve para la recopilación de literatura, siempre y cuando su aplicación vaya dirigida a predecir el rendimiento académico.

Otro aspecto que vale la pena aclarar es que, aunque las variables que se están estudiando en la pregunta de investigación se enfocan en el rendimiento académico y no en la deserción académica, de todas formas, se aceptan dentro de la RSL aquellos trabajos que estudien la deserción académica, pues estos indicadores (la deserción y el rendimiento) están íntimamente ligados. De acuerdo con la revisión de literatura inicial que se hizo de manera exploratoria, la razón principal por la cual se busca predecir el rendimiento académico es para así evitar la deserción de un estudiante que esté en riesgo de hacerlo, ya que, a su vez, el rendimiento académico es la principal variable para determinar si un estudiante es un posible desertor. Un porcentaje considerable de trabajos centran su propuesta en el estudio de la deserción académica en vez de hacerlo en el rendimiento académico. De no incluirse este tipo de trabajos, se estaría dejando por fuera una gran cantidad de información potencialmente útil que puede brindar aportes a la elaboración del estado del arte.

### **1.5.3. Fase III. Criterios de selección de la información**

Inicialmente la búsqueda en las bases de datos acerca de la temática principal *modelos de Machine Learning en educación superior* arroja aproximadamente setecientas investigaciones a partir de las ecuaciones anteriormente mencionadas. Posteriormente se establecen pasos tendientes a filtrar la información y a establecer normas de exclusión. A continuación, se citan algunas:

- Se excluyen aquellos estudios que sean anteriores al 2010.
- Se excluyen aquellos estudios que no relacionen directamente la aplicación de los campos de aprendizaje supervisado, no supervisado y redes neuronales.
- Se excluyen (para el estado del arte) aquellos trabajos que no manejen una estructura o metodología secuencial propia de una investigación cuantitativa.
- Se excluyen aquellos estudios que son una revisión bibliográfica de otros trabajos.

- Se excluyen aquellos artículos que no arrojan resultados específicos del rendimiento de modelos utilizados para predecir el rendimiento académico.
- Se excluyen aquellos estudios que no incluyan dentro del nombre o dentro del resumen las variables indicadas en la ecuación de búsqueda.
- Se excluyen aquellos estudios que no contemplen dentro de su objetivo predecir una variable de tipo categórica o numérica como clase del rendimiento o no usen tareas de agrupación o asociación.

El uso de estas exclusiones hizo que la búsqueda se acotara a cuatrocientos artículos. Muchos de ellos se centraban en la educación básica y media; sin embargo, dado nuestro interés en la aplicación de *Machine Learning* a la educación superior, solo se tuvieron en cuenta aproximadamente trescientos artículos para los temas que se tratan en los capítulos posteriores.

#### **1.5.4. Fase IV. Análisis e interpretación de datos relevantes**

A partir de los artículos ya filtrados, se procedió a agruparlos en cada uno de los campos de división de las aplicaciones del aprendizaje automático de forma general (supervisado, no supervisado y aprendizaje profundo). Por tanto, este estado del arte literario puede no incluir fuentes poco disponibles a través de los argumentos y bases de datos no mencionadas en este documento. Los documentos consultados en esta revisión bibliográfica corresponden a publicaciones que pretendan establecer una predicción dentro del temario de búsqueda a través de múltiples métodos.

El presente estado del arte busca determinar las publicaciones que pretenden establecer una predicción en cuanto al rendimiento académico de los estudiantes a través de diversos algoritmos de *Machine Learning* y en esa medida se enfoca en tratar de resolver las siguientes preguntas:

1. ¿Cuáles son las herramientas y los métodos utilizados para realizar *Machine Learning* en educación superior?
2. ¿Cuáles son los ámbitos de acción de *Machine Learning* en la educación superior?

3. ¿Cuáles son los algoritmos no supervisados en trabajos previos de *Machine Learning* usados en la educación superior?
4. ¿Cuáles son los algoritmos supervisados en trabajos previos de *Machine Learning* usados en la educación superior?
5. ¿Cuáles son los algoritmos de aprendizaje profundo en trabajos previos de *Machine Learning* usados en la educación superior?
6. ¿Cuáles son las herramientas y tecnologías usadas para el análisis de datos relacionadas con *Machine Learning* aplicadas al rendimiento académico?
7. ¿Cuáles son las variables de mayor grado de incidencia en el desempeño académico de los estudiantes, mediante una ponderación asignada en función del número de veces que son incluidas en la construcción y la ejecución de los modelos y algoritmos planteados por diferentes autores?
8. ¿Cómo podría organizarse la información adquirida (modelos, algoritmos) en la construcción y ejecución de los modelos y algoritmos planteados por diferentes autores?

#### **1.5.5. Fase V. Resultados**

Se pretende exponer, discutir y documentar el resultado de la investigación y la aplicabilidad de modelos de *Machine Learning* para el caso de estudio *rendimiento académico en estudiantes de educación superior*. La consecución de esta fase tiene por fin continuar con el desarrollo del macroproyecto *Integración de datos del ámbito educativo y herramientas de análisis de datos para determinar el rendimiento académico en la educación superior*, Código 2-15-593-19, que tiene como fin la implementación de modelos de *Machine Learning* mediante un *software* que permita tomar decisiones y realizar acciones para mejorar el rendimiento académico de los estudiantes de ingeniería de la Universidad Distrital.





# Capítulo 2

## Rendimiento académico

### 2.1. Generalidades del rendimiento académico

El rendimiento académico es considerado por diversos autores como un indicador de tipo multidimensional que depende de aspectos como los objetivos del docente, de la institución, del estudiante, etc. En definitiva, no existe una teoría definitiva o una metodología para medirlo o un indicador para valorarlo.

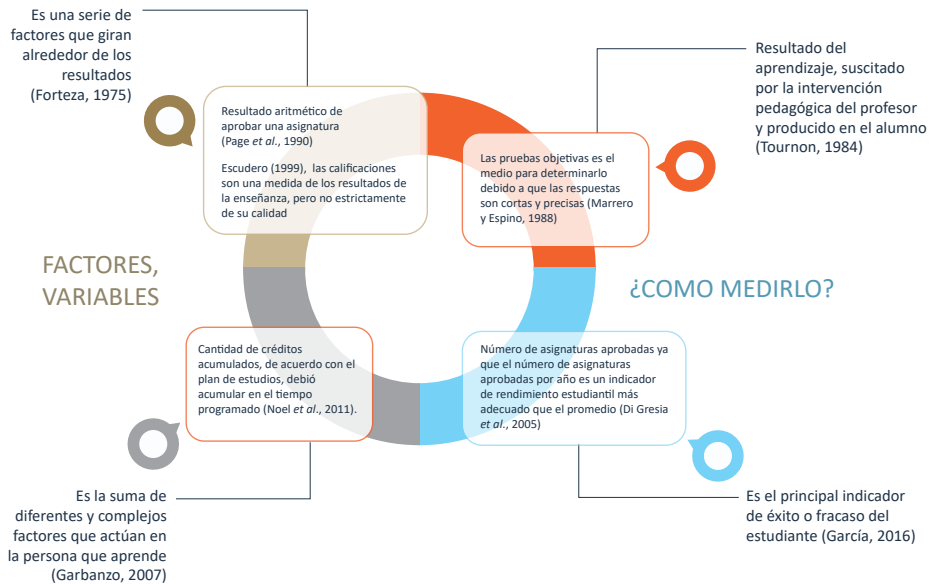
Vale la pena aclarar un poco más el término *rendimiento académico*. Algunos autores han resumido algunas definiciones, como es el caso del estudio de Rojas (2013), que expone diferentes definiciones: el autor muestra que Garbanzo y María (2007) definen el rendimiento académico como la suma de diferentes y complejos factores que actúan en la persona que aprende. Por su parte, Tourón (1985) afirma que el rendimiento académico es resultado del aprendizaje, el producto de la intervención pedagógica del profesor o profesora en el alumno. Rojas (2013) define el rendimiento académico como una serie de factores que giran alrededor de los resultados finales del esfuerzo hecho por la estudiante. En este caso, la definición a tomar en cuenta es la de García, 2016, quien lo considera como el principal indicador de éxito o fracaso del estudiante. Por tal motivo, ha sido considerado como uno de los aspectos importantes a la hora de analizar resultados sobre el proceso de enseñanza-aprendizaje.

Otro aspecto a tener en cuenta es la forma de medición: se pueden interpretar que es solo el resultado aritmético de aprobar una asignatura (Page

*et al.*, 1990); sin embargo, Escudero (1999) señala que las calificaciones son una medida de los resultados de la enseñanza, pero no estrictamente de su calidad. Otra forma de medición serían las pruebas objetivas. Al respecto, Marrero y Espino (1988) destacan que las pruebas objetivas son medios propicios para determinar el rendimiento académico porque permiten respuestas cortas y precisas, sin la influencia subjetiva del profesor. Una tercera forma de medición podría ser a partir del número de asignaturas aprobadas, ya que el número de asignaturas aprobadas por año es un indicador de rendimiento estudiantil más adecuado que el promedio (Porto y Gresia, 2005). Otro indicador que representa el éxito o fracaso de un estudiante es la cantidad de créditos acumulados, pues indica el avance que ha tenido en la carrera, es decir, permite hacer una comparación entre los créditos acumulados por el alumno durante cierto tiempo de estudio y los créditos que, de acuerdo con el plan de estudios, debió acumular en el tiempo programado (Noel *et al.*, 2011). En conclusión, existen diferentes interpretaciones sobre cómo medir el rendimiento académico, pero este trabajo considerará la establecida por Noel *et al.* (2011), Page *et al.* (1990) y Porto y Gresia (2005) (figura 7).

Predecir el rendimiento de los estudiantes se vuelve más desafiante debido al gran volumen de datos en las bases de datos educativas. Es necesario investigar y pretender un modelo (con futuro desarrollo de *software*) que permita implementar diversas herramientas tecnológicas con el fin de predecir y favorecer, por medio de herramientas de aprendizaje automático, la toma de decisiones en el campo educativo tanto para estudiantes como para docentes. Cabe mencionar que, según Johnson *et al.* (2015) en NMC Horizon Report: 2015, la analítica es una tendencia a ser adoptada en la educación superior, tal como lo hacen compañías como Amazon, Netflix y Google para el manejo del marketing y de los usuarios afiliados: aquella puede ayudar a transformar la educación de un sistema de estándar de tamaño único a un sistema flexible, diseñado para satisfacer las necesidades e intereses académicos de los estudiantes.

**Figura 7.** Rendimiento académico. Definiciones y actores



**Fuente:** elaboración propia.

## 2.2. Panorama de posibles variables que predicen el rendimiento académico

Las universidades cuentan con *software* y plataformas que almacenan cantidades enormes de datos de los estudiantes y otros generados durante su estadía en la universidad. En general, estos datos no reciben ningún tratamiento y menos cuando se trata de buscar el mejoramiento del rendimiento académico. De tal manera, es necesario reutilizar la información para crear aplicaciones mucho más ricas, que aporten mayor conocimiento o que complementan la información ya existente.

Existen diversos factores que permiten medir la eficiencia del proceso educativo, muchos de ellos de carácter multidimensional, como es el caso del fenómeno de retención —la diferencia entre número de estudiantes que ingresan en primer semestre y los graduados por año (Salcedo, 2010)—; y del rendimiento académico definido como el principal indicador de éxito o fracaso del estudiante. Por tanto, su determinación ha generado controversia, ya que no existe una teoría definitiva acerca de una metodología para

su medición o un indicador para su valoración. Al ser multidimensional, el rendimiento académico depende de múltiples aspectos tales como los objetivos del docente, de la institución, del estudiante, etc. Así mismo, requiere realizar una integración de las diferentes técnicas y metodologías con el propósito de predecir el rendimiento académico de una manera más confiable (Khan y Choi, 2014).

Por tanto, plantear un modelo implica analizar y pensar diversas variables interrelacionadas para tratar de predecir el rendimiento académico y la deserción estudiantil, entre otros aspectos de la vida académica del estudiante. Ahora bien, aunque según la literatura la gran mayoría de variables del sistema son de tipo cuantitativo y solo unas pocas de tipo cualitativo (redes sociales, por ejemplo), al modelar el fenómeno, no se pretende convertir al estudiante en un ser al que se le puede medir su actuar de forma numérica. Por el contrario, lo que se busca con este modelo es la interpretación de unos resultados numéricos (de lo cuantitativo a lo cualitativo) con el fin de establecer un plan que permita identificar otras cualidades del estudiante, así como tratar de mejorar por medio de sugerencias su actuar académico durante la estadía en la universidad. Esta integración de datos y tecnologías no puede convertirse en una predicción definitiva del estudiante, por cuanto solo informará sobre la probabilidad de ocurrencia de algún suceso (pérdida de materias, año de culminación de sus estudios, etc.) y no dará un veredicto sobre cómo será la vida académica de un estudiante en determinada carrera universitaria.

Los atributos que se han utilizado con frecuencia, según Shahiri *et al.* (2015), son el promedio acumulado de calificaciones (CGPA) y la evaluación interna. Según la revisión de bibliográfica de este último autor, realizada entre el 2002 y el 2012, diez de treinta artículos han utilizado CGPA como atributo principal para predecir el rendimiento de los estudiantes. Así mismo, a través del análisis de correlación de coeficientes, el resultado muestra que CGPA es la variable de entrada más significativa —0,87—, en comparación con otras variables. Otras variables son las agrupadas como evaluación interna; es decir, quices, trabajos de clases, evaluaciones parciales, trabajos de laboratorio, asistencia, etc.

Las variables también identificadas en estudios para predecir el rendimiento académico es la evaluación demográfica (que incluye género, edad, antecedentes familiares y discapacidad, entre otros) y externa de los estudiantes. Cuando hablamos de ella, nos referimos a obtener una nota específica para un tema especial aplicado al estudiante. La razón por la cual se hace distinción en el género del estudiante radica en que el género marca diferencias en el proceso de aprendizaje, tal como lo demuestra en su trabajo Meit *et al.* (2007), que concluye que el género es uno de los atributos importantes que influyen en el rendimiento de los estudiantes. Algunas de las variables que podrían requerir el modelo se muestran en la tabla 2.

**Tabla 2.** Variables que afectan el rendimiento académico

|                         |                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------------|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Factores Preuniversidad | Factores académicos           | Puntaje global ICFES (Colombia) , Año de presentación de prueba ICFES (<2014 o >2014), Puntaje ICFES para área de Matemáticas, Puntaje ICFES para área de Física, Puntaje ICFES para área de Química, Puntaje ICFES para área de Lenguaje, Puntaje ICFES para área de Biología, Puntaje ICFES para área de Ciencias sociales, Puntaje ICFES para área de Filosofía, Puntaje ICFES para área de Ciencias naturales, Puntaje ICFES para área de lectura crítica, Puntaje ICFES para área de Sociales y ciudadanas, Puntaje ICFES área de inglés |
|                         | Factores demográficos         | Edad, género, estado civil, tipo de colegio donde culminó la educación secundaria, experiencia laboral, tiempo transcurrido antes de ingresar a la universidad y carácter del bachillerato                                                                                                                                                                                                                                                                                                                                                    |
|                         | Factores sociales             | Nivel educativo de los padres, número de miembros de la familia y ocupación de los padres                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|                         | Factores socioeconómicos      | Lugar de procedencia, región de procedencia, ingresos familiares, número de autos, casa propia o arrendada, tipo de inscripción y estrato                                                                                                                                                                                                                                                                                                                                                                                                     |
| Factores de Universidad | Factores de gestión académica | Becas obtenidas, año de ingreso a la universidad pénsum, número de créditos número de créditos cursados, número de créditos aprobados, nota final para cada asignatura, número de asignaturas cursadas, número de asignaturas perdidas, número de asignaturas ganadas, número de veces que ha perdido una asignatura, número de asignaturas repetidas y número de asignaturas canceladas                                                                                                                                                      |
|                         | Factores tecnológicos         | Posibilidades de acceso a internet, N.º de accesos a LMS, tiempo de uso total de LMS, interacciones en línea, N.º de recursos accedidos, N.º de documentos descargados, tiempo por actividad, frecuencia de uso, coordenadas x e y del mouse, tiempo de clic en cada actividad, cantidad de clic, documentos accedidos, participación en quizzes, tiempo que el estudiante demora para resolver una pregunta y participación en foros                                                                                                         |

|                         |                                    |                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Factores de Universidad | Factores de biblioteca             | Frecuencia de acceso, tipos de área de búsqueda y número de libros en la biblioteca                                                                                                                                                                                                                                                                                  |
|                         | Factores institucionales           | Tamaño del salón de clases, organización y planificación del personal, horarios de los cursos, tipo y tamaño de la institución, procesos de funcionamiento tamaño del salón de clases e infraestructura, políticas                                                                                                                                                   |
|                         | Factores pedagógicos               | Número de estudiantes por docente, número de estudiantes matriculados, número de repetidores, métodos y materiales didácticos utilizados, motivación de los estudiantes, tiempo dedicado por los profesores a la preparación de sus clases, formación académica del profesor, años de experiencia del profesor, evaluación docente, tipo de docente y área académica |
|                         | Factores intelectuales y afectivos | Concentración, memoria, comprensión verbal, razonamiento, fluidez verbal sociabilidad, seguridad, integración, relación padre-hijo y relación profesor-alumno                                                                                                                                                                                                        |

**Fuente:** adaptado de García (2016), Estrada y Quintero (2015), Tomás *et al.* (2014) y Oladejo (2010).

## 2.3. Indicadores del rendimiento académico: calificaciones

Como se ha mencionado, el rendimiento académico se determina tomando en cuenta variables cualitativas y cuantitativas. De igual manera, no existe una teoría definitiva acerca de una metodología para su medición o un indicador para su valoración de manera objetiva y fiable, aunque existen algunos indicadores que se han venido usando en algunas investigaciones. Las notas son quizás la forma más fácil de medición, pues representan una medición cuantitativa o cualitativa del rendimiento. Una nota alta puede indicar un excelente rendimiento académico y una nota baja todo lo contrario. Sin embargo, Page *et al.* (1990) indican que las calificaciones comprenden un valor relativo como medida de rendimiento, ya que no existe un criterio estandarizado para todos los centros, para todas las asignaturas, para todos los cursos y para todos los profesores. Así mismo, Escudero (1999) señala que las calificaciones son una medida de los resultados de la enseñanza, pero no estrictamente de su calidad, pues están condicionadas no solo por la calidad de los alumnos, sino también por el criterio y el rigor personal del profesor a la hora de diseñar la enseñanza, y valorar y calificar el aprendizaje y el rendimiento académico.

## **2.4. Indicadores del rendimiento académico: pruebas objetivas**

Las pruebas objetivas son aquellas que requieren por parte del estudiante respuestas claras y precisas sobre un hecho particular. Estas pruebas pueden estar alojadas en plataformas LMS, aunque también pueden ser más tradicionales. Estas pruebas tienen un alto grado de validez debido a que cumplen específicamente los propósitos para los que fueron elaboradas (Page *et al.*, 1990).

## **2.5. Indicadores del rendimiento académico: número de asignaturas aprobadas**

Teniendo en cuenta la definición de rendimiento académico de García (2016), que lo considera como el indicador del éxito/fracaso de un estudiante, tener en cuenta el número de asignaturas aprobadas puede servir como medición. Este indicador muestra la cantidad de materias aprobadas por un estudiante durante un determinado periodo de tiempo (la relación entre asignaturas aprobadas sobre el total de asignaturas matriculadas en un semestre académico). Un número alto de asignaturas aprobadas indica un alto porcentaje de éxito, y por tanto de mejor rendimiento académico; en caso contrario, si el resultado es un número bajo, se podría decir que hay un bajo desempeño académico. Peor aún, si no aprobó ninguna de las materias, se concluiría que el estudiante “fracasó” académicamente. El estudio realizado por Porto y Gresia (2005) concluye que el número de asignaturas aprobadas por año es un indicador de rendimiento estudiantil más adecuado que el que tiene en cuenta el promedio, ya que sintéticamente la cantidad de materias aprobadas comprende mayor variabilidad que el indicador relacionado con las notas.

## **2.6. Indicadores del rendimiento académico: créditos acumulados**

Este indicador permite determinar la relación entre cantidad de créditos acumulados por el estudiante durante cierto tiempo de estudio y los créditos que, de acuerdo con el plan de estudios, debió acumular en el tiempo

programado. De manera similar, un valor alto indicaría buen rendimiento; en el caso contrario, se mostraría un bajo rendimiento académico. Al igual que el anterior indicador, este representa el éxito o fracaso de un estudiante, pues indica el avance en la carrera. Es decir, permite hacer una comparación entre los créditos acumulados por el alumno durante cierto tiempo de estudio y los créditos que, de acuerdo con el plan de estudios, debió acumular en el tiempo programado (Rodríguez *et al.*, 2011).



# Capítulo 3

## *Machine Learning*

### 3.1. Generalidades del *Machine Learning*

*Machine Learning* es un subconjunto de la inteligencia artificial que consiste en la construcción de un modelo matemático basado en datos de muestra conocidos como “datos de entrenamiento”. *Machine Learning* se usa para hacer predicciones o tomar decisiones sin que un algoritmo se programe de manera explícita para realizar una tarea o ejecutar determinada función (IBM, 2020).

*Machine Learning* (ML) también se puede percibir como un conjunto de herramientas que, de una u otra forma, busca aprendizaje computacional a partir de datos sin que la máquina reciba un código específico para el problema. Teniendo en cuenta la definición de Hackeling (2014), el ML es el diseño y el estudio de programas computacionales que usan experiencia para decisiones futuras. Así mismo, Huddleston y Brown (2018) definen el foco de *Machine Learning* en el uso de algoritmos para mejorar la descripción, la predicción y el desempeño descriptivo del contexto real; este campo, durante las últimas dos décadas, ha tenido avances significativos en su desarrollo (Prenkaj *et al.*, 2020).

A su vez, se trata de una rama en evolución. *Machine Learning* se centra en los algoritmos computacionales diseñados para emular la inteligencia humana pues aprenden del entorno circundante. Se les considera el caballo de batalla de la nueva era del *Big Data*. Las técnicas basadas en el aprendizaje automático se han aplicado con éxito en diversos campos: desde el

reconocimiento de patrones, la visión por computadora, la ingeniería de naves espaciales, las finanzas, el entretenimiento y la biología computacional hasta las aplicaciones médicas y biomédicas (Naqa y Murphy, 2015).

La inteligencia artificial a su vez es percibida como un área de la informática, pero tiene que ver con el desarrollo de programas inteligentes o que puedan hacer cosas inteligentes. Por tanto, el *Machine Learning* o aprendizaje automático es un subcampo que específicamente busca un modelo o procedimiento que haga el mejor uso de los datos históricos compuestos por entradas y salidas para predecir hábilmente salidas nuevas a partir de entradas nuevas que no eran conocidas (Zhang, 2017).

Los campos de la minería de datos y del aprendizaje automático están entrelazados. La minería de datos utiliza algoritmos de aprendizaje automático para interrogar grandes bases de datos y descubrir conocimientos ocultos en los datos, mientras que muchos algoritmos de aprendizaje automático emplean métodos de minería de datos para preprocesar los datos antes de aprender las tareas deseadas. Sin embargo, debe tenerse en cuenta que el aprendizaje automático no se limita a resolver problemas similares a los de las bases de datos, sino que también se extiende a la resolución de desafíos complejos de inteligencia artificial: aprende y se adapta a una situación que cambia de forma dinámica, como las prácticas de oncología y radioterapia (Naqa y Murphy, 2015).

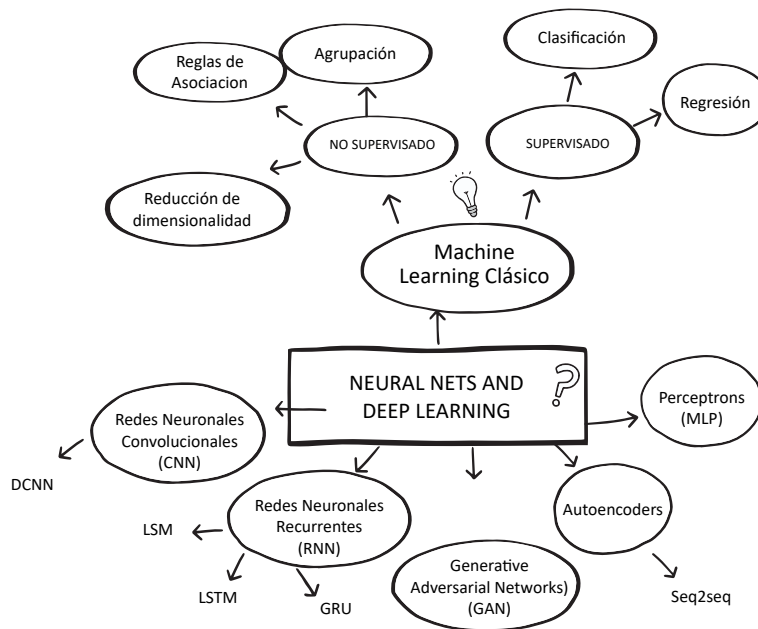
El uso de las distintas técnicas de *Machine Learning* con fines educativos es prometedor en cuanto al descubrimiento de patrones. Sin embargo, la aplicación del aprendizaje automático en la educación se enfrenta varios desafíos que deben abordarse (Prenkaj *et al.*, 2020). El primero de ellos tiene que ver con la falta de información disponible y existente, en especial en los países en desarrollo; el segundo, los baches en los sistemas de alerta temprana robustos que identifiquen patrones indeseables; y el tercero, la clasificación de estudiantes de acuerdo con sus problemas y capacidades.

### 3.2. Clasificación del *Machine Learning*

El aprendizaje automático se puede clasificar según la naturaleza del etiquetado de datos: puede ser supervisado, no supervisado o semisupervisado, como se muestra en la figura 8. El aprendizaje supervisado se utiliza para

estimar un mapeo desconocido (entrada, salida) a partir de muestras conocidas (entrada, salida), y se debe tener en cuenta que la salida está etiquetada (por ejemplo, con las categorías *clasificación* y *regresión*). En el aprendizaje no supervisado, solo se proporcionan muestras de entrada al sistema de aprendizaje (por ejemplo, agrupación y estimación de la función de densidad de probabilidad). El aprendizaje semisupervisado es una combinación de ambos —supervisado y no supervisado—: la parte de los datos está parcialmente etiquetada y la parte etiquetada se usa para inferir la parte no etiquetada (por ejemplo, sistemas de recuperación de texto/imágenes).

**Figura 8.** Clasificación del *Machine Learning* según su naturaleza



**Fuente:** adaptado de Gomes (2019).

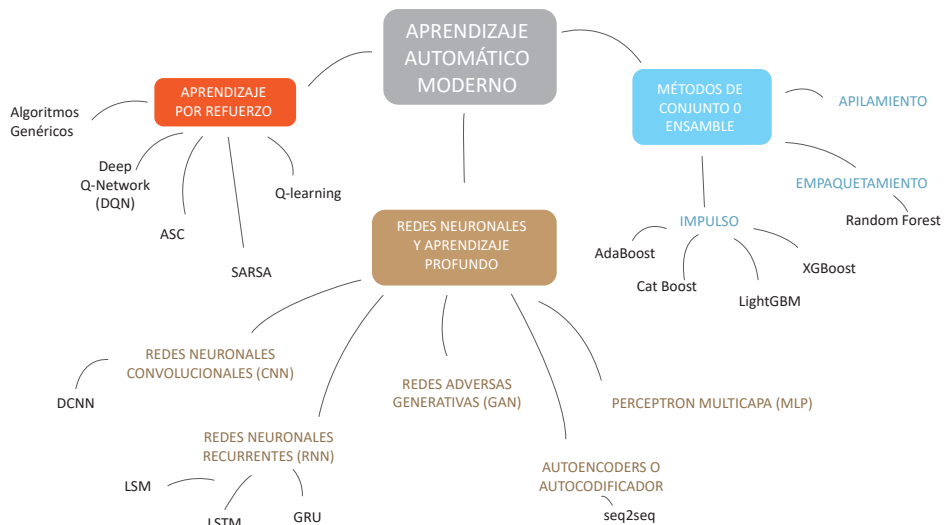
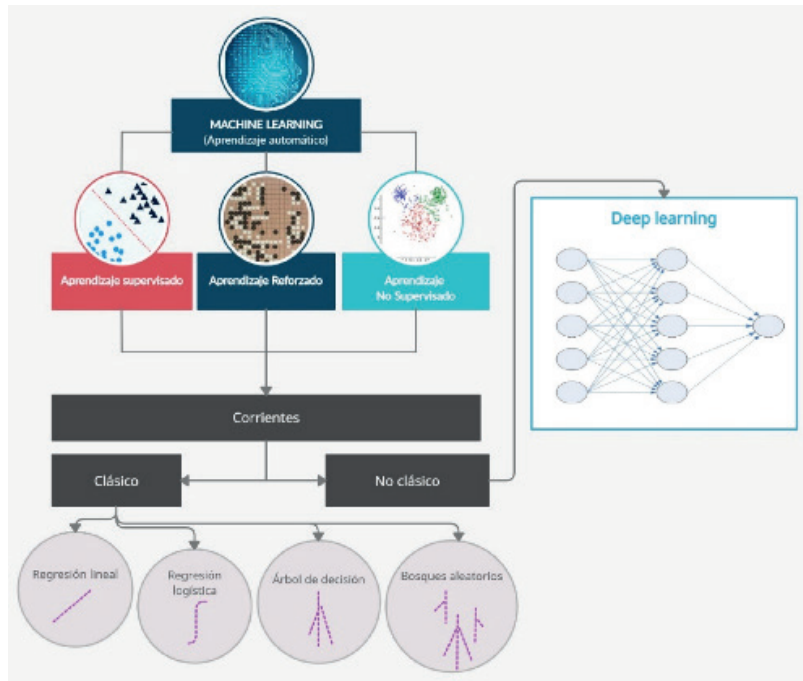
Desde la perspectiva del aprendizaje de conceptos, el aprendizaje automático se puede clasificar en aprendizaje transductivo y aprendizaje inductivo. El aprendizaje transductivo implica la inferencia de casos de entrenamiento específicos a casos de prueba específicos con el uso de etiquetas discretas —como en agrupamiento— o de etiquetas continuas —como en aprendizaje múltiple—. Por otro lado, el aprendizaje inductivo tiene como objetivo

predecir los resultados de los insumos que el alumno no ha encontrado antes. En este sentido, Gomes (2019) defiende la necesidad de un sesgo inductivo en el proceso de entrenamiento para permitir que un algoritmo de aprendizaje automático se generalice más allá de la observación invisible.

El *Machine Learning* se clasifica de acuerdo con la manera en que los algoritmos “aprenden” (los tipos de aprendizaje existentes son: aprendizaje supervisado, no supervisado y reforzado) y de acuerdo con el paradigma (modelos clásicos y no clásicos del ML). Dentro de los modelos clásicos de ML están los métodos de regresión, clasificación, clustering, reducción de la dimensionalidad, entre otros (Salgado Gomez y Velasco Franco, 2020).

Posteriormente aparecen otras formas de desarrollo de ML, como las redes neuronales profundas. Si bien las redes neuronales hacían parte del ML clásico —cuando se trataba de modelos de una sola capa oculta dentro de la red de neuronas (llamado perceptrón monocapa), estos fueron mejorando de tal manera que hoy en día una red neuronal puede comprender diversas capas ocultas y así permitir resolver problemas muy complejos. Un modelo de red neuronal con varias capas ocultas se llama “perceptrón multicapa”. Gracias a la aplicabilidad e impacto que tienen estos modelos, forman parte del paradigma no clásico del ML, y específicamente de una corriente llamada *deep learning* que engloba a las redes neuronales artificiales profundas. La palabra *deep* traduce “profundo” y hace referencia a que los modelos de red neuronal de esta vertiente no son tan sencillos, sino que comprenden varias capas ocultas (sean pocas o muchas). Los problemas más comunes que implican el uso de *deep learning* son de regresión y de clasificación (Prenkaj *et al.*, 2020). A continuación, se muestra de manera gráfica la clasificación del ML (figura 9).

**Figura 9.** Campo de las redes neuronales en *Machine Learning*



Fuente: Barrios (2022).

Las redes neuronales artificiales (RNA o ANN por sus siglas en inglés) son estructuras abstractas inspiradas en mecanismos y en la estructura física

del conocimiento y el aprendizaje biológico. Para su diseño se toma como base el funcionamiento de la neurona biológica (Zambrano Matamala *et al.*, 2011); es decir, de la composición de neuronas repartidas en capas o *layers*, y de la interconexión de las neuronas entre sí y su capacidad para compartir información. La estructura de una neurona natural comprende dendritas que son las entradas de información, el núcleo que procesa la información y los axones que la transmiten. Los axones también se conectan con otras neuronas a través del proceso conocido como sinapsis (Cuevas-Tello, 2018). Las redes neuronales artificiales están compuestas por nodos (o neuronas) interconectados, y cada nodo de la red es una neurona. A su vez, cada neurona es una función matemática que ajusta los valores de los datos que pasan a través de ella. También se definen como conjuntos de entidades de procesamiento altamente interconectados. Este modelo está inspirado en el sistema nervioso central biológico, más específicamente en el cerebro. La neurona, por su parte, acepta entradas y responde con salidas (Rastrollo-Guerrero *et al.*, 2020). Por lo general, entre más capas tenga una red neuronal, más información y funciones complejas puede manejar (Hernández-Blanco *et al.*, 2019). Estas están incluidas en el campo del *Machine Learning*.

Una única neurona es incapaz de resolver por sí sola un problema no lineal, como por ejemplo un condicional tipo XOR. Por ello hay conjuntos de neuronas que funcionan a manera de red y trabajan en conjunto (Cuevas-Tello, 2018). Para aprender, las redes neuronales utilizan un algoritmo que les permite ponderar el error. Este algoritmo se llama *back propagation* o algoritmo de retropropagación. En el caso de estas redes, durante el aprendizaje se minimiza la función de error entre la salida deseada y la salida real de la red (Zambrano Matamala *et al.*, 2011).

Una red neuronal tiene un número fijo de entradas y salidas, y este número corresponde al número de neuronas en las capas de entrada y salida, respectivamente. Sin embargo, el número de capas ocultas (profundidad), así como el número de neuronas por capa oculta (anchura) son parámetros que no se definen de la misma manera (Hernández-Blanco *et al.*, 2019).

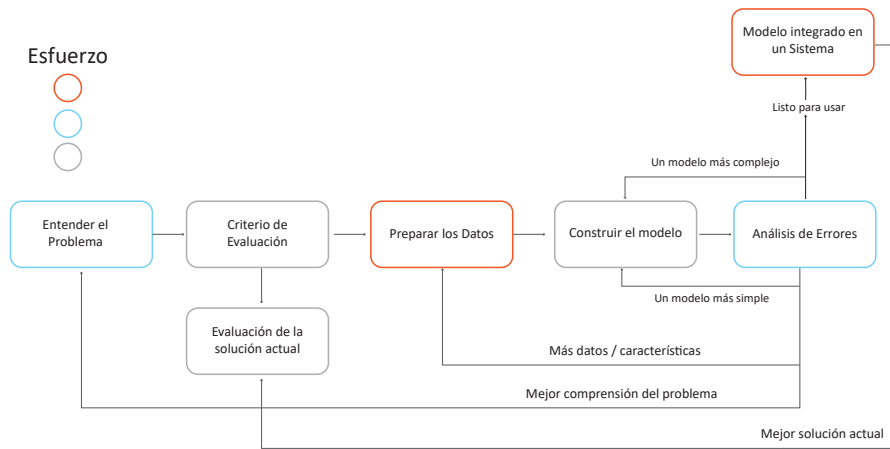
Para el entrenamiento de una red hace falta buscar el ajuste de los parámetros para cada neurona; el problema es que algunas redes profundas pueden tener millones de estos parámetros (Hernández-Blanco *et al.*, 2019).

Entonces, es necesario medir la salida real contra la deseada mediante el algoritmo de *back propagation*. Este método usa entre sí un método de optimización no lineal conocido como *gradiente estocástico del descenso* (Hernández-Blanco *et al.*, 2019).

### 3.3. Proceso de *Machine Learning*

Según Heras (2020) este proceso es muy similar al proceso de *Data Mining* o minería de datos. Recordemos que el *Machine Learning*, en cierta medida, tiene su origen en esta área de la ciencia de datos, más adelante tanto potencializada como soportada en diversos campos como la ingeniería, las matemáticas, la estadística, el aprendizaje computacional, la visualización y la gestión de bases de datos, entre otros. Los pasos del proceso son: entender el problema, definir un criterio de evaluación, evaluación de la solución actual, preparar los datos, construir el modelo, análisis de errores y modelo integrado en un sistema (figura 10).

**Figura 10.** Fases del proceso de *Machine Learning*



**Fuente:** Heras (2020).

Según IArtificial.Net (2018), un proceso de *Machine Learning* puede resumirse en siete fases:

**Fase I: Comprender el problema:** como primer paso, es fundamental entender y comprender el problema que se quiere resolver y los datos que se tienen a disposición. Comprender el problema requiere de una gran cantidad de tiempo, sobre todo si el problema se da en un segmento de mercado o en una industria en la que hay poco conocimiento.

**Fase II: Definir un criterio de evaluación:** trata principalmente del cálculo de error. Normalmente se usa la magnitud cruzada para problemas de clasificación y el error cuadrático medio para problemas de regresión. Sin embargo, es recomendable usar otras medidas como la exhaustividad y la precisión.

**Fase III: Evaluación de la solución vigente:** generalmente, el problema que se quiere solucionar con *Machine Learning*, ya se está solucionando, pero de otra manera. En este paso se mide el rendimiento de la solución vigente (a partir del esquema de evaluación seleccionado), y se compara con el rendimiento del modelo propuesto a partir de *Machine Learning*; de esta manera se puede saber si vale la pena implementar el modelo propuesto o si se continua con la solución vigente.

**Fase IV: Preparar los datos:** la disposición de los datos es una de las fases del *Machine Learning* que requiere mayor esfuerzo. Los principales desafíos en esta fase son:

**Datos inconclusos:**

- Se deben eliminar los datos inconclusos. La opción más sencilla es quedarse con los datos incompletos solo si se trata de una minoría. Sin embargo, si disponemos de una gran base esta no es la mejor opción dado que habría muchos datos desechados por obligación.
- Atribuir con un valor razonable: Generalmente si hay un valor faltante, automáticamente se puede agregar un valor que tenga sentido. Por ejemplo, en una encuesta, si algún usuario no ha registrado su edad, se puede usar la edad media de todos los usuarios encuestados que compartieron ese dato.
- Atribuir con un modelo de aprendizaje automático: En un nivel más alto, es posible construir un modelo de *Machine Learning* que, a través de



predicciones, permita saber cuál es el valor faltante si aprende de los casos en los que se dispone de datos.

- Hacer uso de alguna técnica de *Machine Learning* que permita ejecutar y manejar datos incompletos.

**Datos combinados de varias fuentes:** en la mayoría de los casos, es necesario combinar datos de diferentes fuentes. Algunos datos provienen de múltiples bases, otros de hojas de cálculo de Excel, ficheros, entre otros. Es recomendable combinar los datos de tal forma que los algoritmos de *Machine Learning* logren comprender y ejecutar toda la información de entrada.

**Ajustar los datos en un formato adecuado:** al considerar el uso de librerías de *Machine Learning* disponibles en diferentes herramientas, en primer lugar, se recomienda ajustar en un formato específico a los datos. Estas librerías permiten que los datos tengan forma de matriz o de tensor. Un tensor es la generalización de una matriz.

**Normalización de la información de entrada:** al ejecutar algoritmos de *Machine Learning* es muy útil normalizar la información de entrada con el fin de facilitar el aprendizaje a la técnica que se ejecute. Se entiende por normalizar el proceso de ajuste de los datos de entrada a una escala semejante.

**Fase V: Construir el modelo:** una vez que se hace una preparación de los datos, se reduce el esfuerzo, pues ya existen varias librerías de *Machine Learning* abiertas. Durante esta fase, se elige el tipo de técnica de *Machine Learning*.

**Fase VI: Análisis de errores:** en esta fase es necesario realizar un esfuerzo promedio. Analizar errores permite comprender cuáles son los ajustes necesarios para mejorar los resultados de *Machine Learning*. En particular las opciones serán:

- Usar un modelo más robusto.
- Aplicar un modelo más sencillo.
- Analizar la necesidad de contar con más datos o más características.
- Desarrollar una alta comprensión del problema e intuir de mejor manera cuál es el paso por seguir.

**Fase VII: Modelo integrado en un sistema:** una vez que se comprenda por completo el error, se debe comparar con el error propuesto desde *Machine Learning*. Si es lo que se requiere, se implementa el modelo del *Machine Learning* propuesto. La fase de integración de un modelo de *Machine Learning* en un sistema requiere un esfuerzo más alto. Se requiere de lo siguiente:

- Iterar de forma automática los pasos para ajustar los datos de entrada.
- Conexión con otras partes del sistema para alcanzar sinergia.
- Implementar y ejecutar los resultados del algoritmo en todo el sistema.
- Reconocer los errores del modelo automáticamente.
- Alertar a las diferentes partes del sistema si los errores reconocidos crecen exponencialmente con el tiempo.
- Reformular y ejecutar el modelo de *Machine Learning* propuesto con información de entrada nueva, bien sea automáticamente o de forma manual.

## Capítulo 4

# Aprendizaje no supervisado: factores y variables que influyen en el rendimiento académico

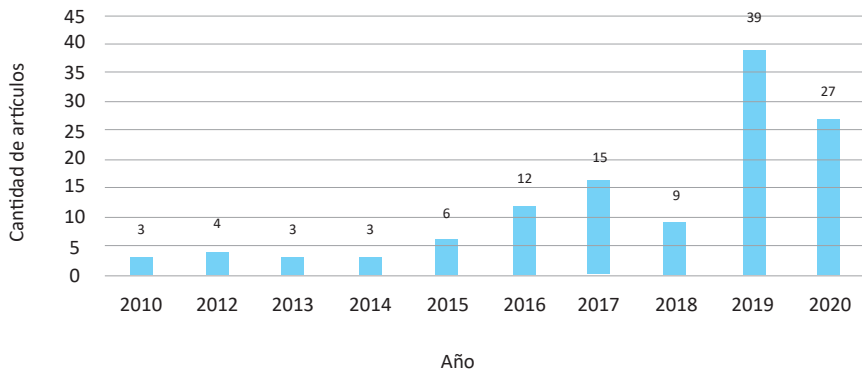
Antes de proceder a establecer las variables y factores que se han tenido en cuenta en los trabajos sobre rendimiento académico que usan algoritmos de *Machine Learning* no supervisados, daremos un vistazo a la información recopilada.

A partir de la información recopilada y filtrada a partir de los criterios mencionados para el periodo 2010-2020, y teniendo en cuenta solo aquella información que implicara el uso del aprendizaje no supervisado para determinar el rendimiento académico de los estudiantes en educación superior, fue posible generar —como se muestra la figura 11— un panorama general de las múltiples aplicaciones de esta técnica de análisis de datos. Así mismo, la cantidad de artículos encontrados al aplicar las ecuaciones de búsqueda es de 122 trabajos (figura 11).

Se analizaron los campos de estudio principales del aprendizaje automático no supervisado consultado en las seis bases de datos mencionadas. Sin embargo, para cumplir el objetivo propuesto, esta investigación se enfocó en la búsqueda en artículos de educación centrados en factores como rendimiento, comportamiento y aprendizaje de forma general. Estos temas son los principales del 49,18% de los artículos analizados; el porcentaje restante corresponde a artículos enfocados en salud, tecnología, ingeniería, ciencias básicas y administración/economía. Fue evidente que estas son

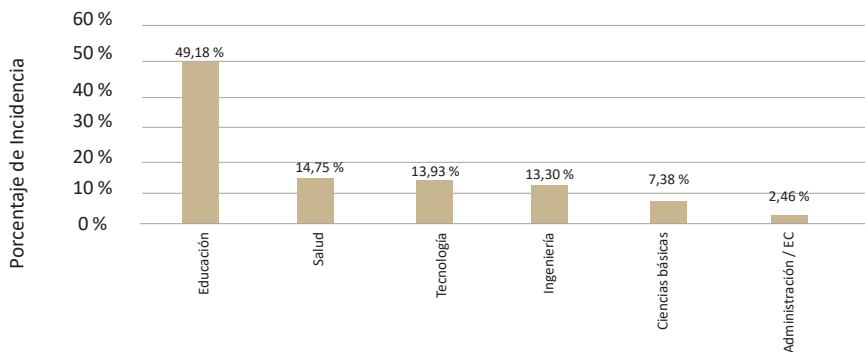
las tendencias claras de investigación y aplicabilidad a nivel mundial en el aprendizaje automático no supervisado (figura 12).

**Figura 11.** Agrupación de referencias bibliográficas por año



Fuente: elaboración propia.

**Figura 12.** Clasificación porcentual de los campos de estudio en los artículos referenciados



Fuente: elaboración propia.

Así, se confirma que el presente tema de estudio, la *construcción de modelos de Machine Learning con aprendizaje no supervisado para determinar el rendimiento académico en estudiantes de educación superior*, es altamente relevante en el ámbito educativo, ya que no hay un número significativo de investigaciones en las bases de datos que analicen el rendimiento académico en la educación superior a través de modelos de aprendizaje automático no supervisado.

Con respecto al contexto académico, los estudiantes universitarios no solo están sometidos a la evaluación continua de sus maestros y autoridades, sino

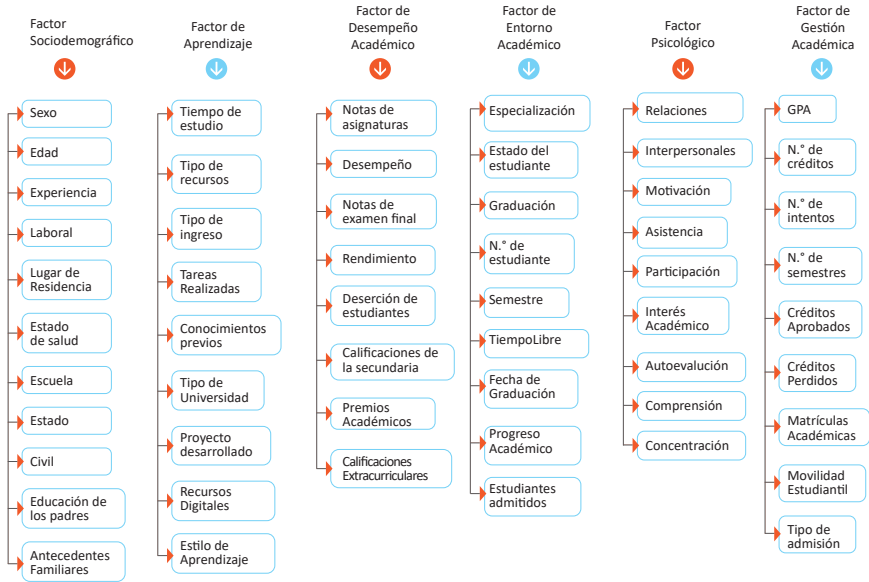
también a la de los miembros del grupo estudiantil al cual pertenecen y con el que se relacionan cotidianamente. Así, se amplía la cantidad de variables que se deben analizar para poder determinar causales directas e indirectas en el rendimiento académico del estudiante y cuáles pueden ser los factores que lo lleven a desertar o a abandonar su formación académica; podemos, rápidamente enumerar algunos de ellos: factor sociodemográfico, factor de aprendizaje, factor de desempeño académico, factor de entorno académico, factor psicológico y factor de gestión académica. No obstante, aunque la gran mayoría de variables son de tipo cuantitativo, y solo unas pocas de tipo cualitativo (interés académico del estudiante, autonomía, comunicación, argumentación, por ejemplo), no se pretende medir el actuar del estudiante bajo un parámetro numérico.

En este capítulo se presenta de forma detallada un conglomerado de variables tratadas por diversos autores de múltiples referencias bibliográficas encontradas en las bases de datos mencionadas anteriormente. Estas variables se trabajan con el propósito de predecir el rendimiento y la deserción académica de estudiantes en educación superior mediante la construcción de modelos de aprendizaje automático no supervisado, variables que se clasifican en seis grupos principales como se muestra en la figura 13.

## **4.1. Factor sociodemográfico**

Diversos estudios centrados en el ámbito educativo han analizado la relación que existe entre variables de tipo sociodemográfico y el rendimiento o la deserción académica de estudiantes en instituciones de educación superior (figura 14). La revisión del estado del arte muestra que este factor agrupa la dimensión social, la evolución y las características generales de los individuos, y que existen dos grandes categorías que agrupan estos factores: las características inherentes de los estudiantes —sexo, edad, estado civil, estado de salud, entre otras—; y las variables socioeconómicas —experiencia laboral, lugar de residencia, escuela, educación de los padres, antecedentes familiares, entre otros—.

**Figura 13.** Variables influyentes en el rendimiento



Fuente: elaboración propia.

**Figura 14.** Investigaciones que consideran variables del factor sociodemográfico

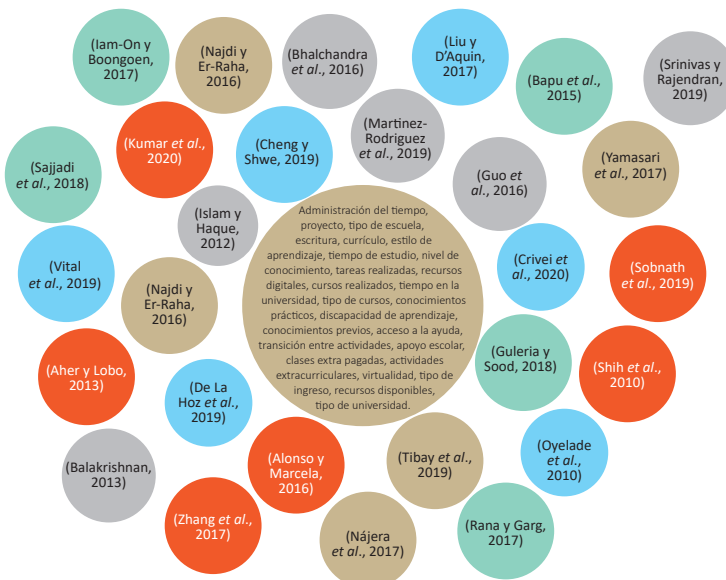


Fuente: elaboración propia.

## 4.2. Factor de aprendizaje

El factor de aprendizaje tiene que ver con todas aquellas circunstancias que en mayor o menor medida condicionan el proceso de aprendizaje —lo favorecen o lo dificultan—. Se considera factores de aprendizaje aquellos directamente relacionados con el alumno, el docente y la institución de educación superior: los tiempos de estudio, los tipos de cursos, los tipo de ingresos, las tareas realizadas, el tipo de universidad, los recursos digitales, los recursos físicos, entre otros (figura 15). El factor de aprendizaje permite medir la interacción de los estudiantes con las plataformas de educación virtual que las universidades ponen a disposición (Virtual, 2021). La evolución de los sistemas educativos ha determinado que la implementación de actividades en línea mejora el nivel de enseñanza si se lo compara con el modelo tradicional, ya que estos están determinados por operaciones del pensamiento, el funcionamiento de los hemisferios cerebrales, la capacidad de concentración, la memoria, las técnicas y estrategias adecuadas. En estos procesos influye la actitud, la motivación, la voluntad y las relaciones personales (Monroy y Pina, 2014).

**Figura 15.** Investigaciones que consideran variables del factor de aprendizaje



**Fuente:** elaboración propia.

### 4.3. Factor académico

El bajo rendimiento académico es un problema que enfrentan estudiantes y profesores en diferentes niveles educativos de las instituciones de educación superior. El rendimiento académico, sea cual sea, representa el aprendizaje alcanzado por el estudiante y el nivel de eficacia en la consecución de los objetivos curriculares en diversas asignaturas. El factor académico muestra estas representaciones y alcances. Algunos autores afirman que los estudiantes con bajo rendimiento académico tienen dos dificultades primordiales: no saben estudiar y no saben aprender. Por ello, el docente debe enfocar su trabajo en el mejoramiento de las habilidades para aprender de los estudiantes. En este factor se presentan variables como las notas de asignatura, las notas de aprendizaje, el desempeño, la deserción, los premios académicos, las calificaciones extracurriculares, las calificaciones de las secundarias, entre otros (figura 16).

**Figura 16.** Investigaciones que consideran variables del factor académico



**Fuente:** elaboración propia.



#### 4.4. Factor entorno académico

El factor entorno académico se relaciona con las características del sistema educativo del que forman parte los estudiantes; se trata de las características funcionales y estructurales de los centros educativos: ubicaciones físicas, contextos y culturas en las que los estudiantes aprenden. Además, también muestra los modos de interacción y las formas en las que los maestros organizan el entorno educativo para facilitar el aprendizaje. Este factor agrupa variables como especialización, graduación, estado del estudiante, número del estudiante, tiempo libre, fecha de graduación, progreso académico, estudiantes admitidos, entre otros (figura 17).

**Figura 17.** Investigaciones que consideran variables del factor académico II



**Fuente:** elaboración propia.

## 4.5. Factor psicológico

Muchos estudios analizan la relación entre la motivación y el rendimiento, dos variables que muestran el interés del estudiante y sus aspiraciones. En la medida en que un estudiante muestra más interés por lo que realiza y sus aspiraciones se ajustan a sus posibilidades, estará más motivado y este hecho redundará en un mejor aprovechamiento académico. Por otro lado, el nivel de ansiedad y otras características personales del estudiante podrían ser facilitadores o inhibidores del rendimiento.

El rendimiento académico es también la resultante del complejo mundo que envuelve al estudiante, determinado por una serie de aspectos cotidianos. Cuando hablamos del factor psicosocial, nos referimos a uno que considera las conexiones que se dan entre la persona y la sociedad, ya que es evidente que ejercen una clara influencia sobre los individuos. Incluye variables como interés, tiempo dedicado, autorregulación, esfuerzo, capacidad de trabajo, intensidad de estudio, competencias, aptitudes, personalidad, atención, motivación, memoria, medio relacional, recursos tecnológicos, entre otros. Estas variables afectan directamente el desempeño académico de estudiantes universitarios (figura 18).

## 4.6. Factor de gestión académica

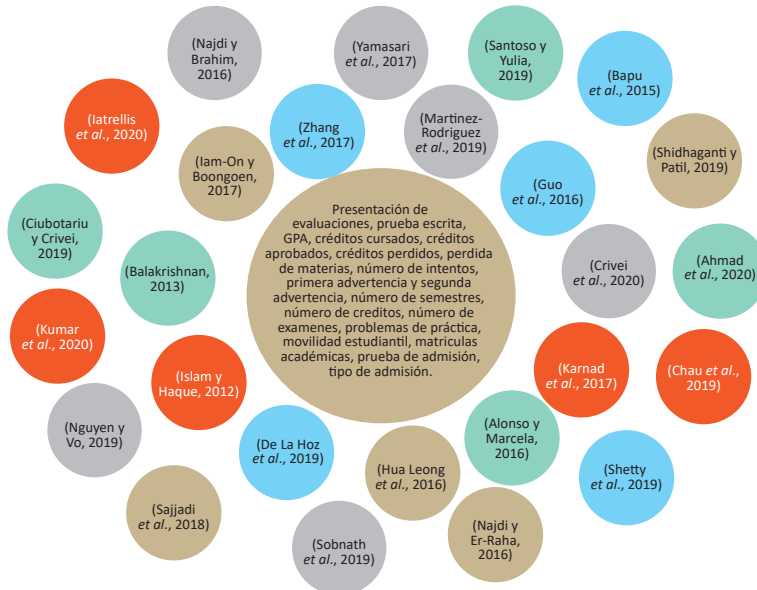
El factor de gestión académica da luces acerca de los fenómenos de facilitación y mejora de los procesos formativos que imparten las instituciones de educación de educación superior, así como la definición, la implementación y el seguimiento de procesos de mejoramiento continuo que permitan responder y anteponerse a las necesidades formativas de los estudiantes, a las exigencias del entorno, del mercado laboral y la política pública. Este factor permite analizar la evolución del estudiante durante su etapa universitaria, en qué áreas se presentan dificultades y cuáles son las causas. Así, es posible brindar acompañamiento por parte de bienestar institucional del instituto de educación superior, y mejorar las expectativas y los deseos de superación para mejorar su calidad de vida. Las variables contempladas en este factor fueron GPA, número de créditos, número de semestres, créditos aprobados, movilidad académica, entre otros (figura 19).

**Figura 18.** Investigaciones que consideran variables del factor psicológico



Fuente: elaboración propia.

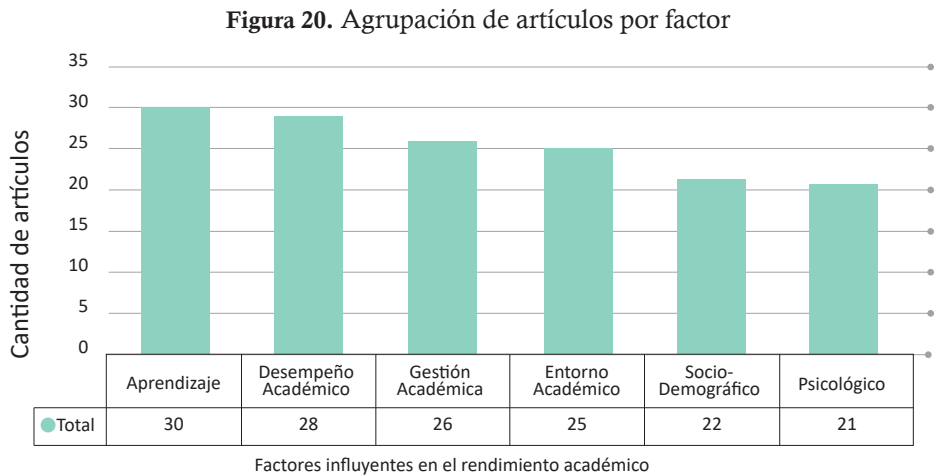
**Figura 19.** Investigaciones que consideran variables del factor de gestión académica



Fuente: elaboración propia.

A partir de la clasificación de las variables por factores realizada previamente, podemos afirmar que, en el universo de investigaciones estudiadas, las variables del factor de aprendizaje son las más utilizadas durante la última década: representan el 50 % de los artículos de educación referenciados (figura 20). Esto puede deberse a que la mayoría de los procesos educativos se ven afectados mayormente por escenarios que se relacionan indirectamente con el rendimiento académico de estudiantes universitarios.

Al factor de aprendizaje le sigue muy de cerca el factor académico, que fue usado en el 46,66 % de las investigaciones realizadas; este factor se compone en su mayoría de variables que representan la forma tradicional de evaluar el proceso formativo de los estudiantes, además de las capacidades innatas de ellos como forma de expresión, talentos, entre otros.



**Fuente:** elaboración propia.

En cuanto al factor de gestión académica, las variables más representativas en las investigaciones analizadas son GPA, créditos aprobados, créditos cursados, número de intentos, número de semestres, advertencias de salida, entre otras. Este factor representa el 43,33 % de las referencias bibliográficas que, junto con el factor de aprendizaje, rigen o determinan la metodología de calificación y la continuidad del proceso de aprendizaje de los estudiantes de educación superior. Finalmente, los otros tres factores, entorno académico, sociodemográfico y psicológico, tienen una relevancia similar en

comparación con los factores anteriormente mencionados, lo que significa que cada uno de los diferentes autores consideran estas variables como de importancia similar. Es importante aclarar que existe una fuerte interrelación o dependencia entre los factores, y que cualquier alteración de uno de ellos, por mínima que sea, puede repercutir en los valores de las demás variables. Una alteración puede afectar el rendimiento académico del estudiante e influir en la deserción de las instituciones de educación superior.

Con respecto a las variables que influyen en el rendimiento académico, la más representativa es GPA (*Grade Point Average* o promedio de calificaciones acumuladas). Esta representa el 23,33% del total de artículos. Se trata de un método utilizado a nivel mundial para evaluar y categorizar el desempeño académico de los estudiantes durante su etapa formativa, y le siguen variables influyentes como el sexo (hombre o mujer), la edad, los puntajes de asignaturas, tipo de escuela (privada, pública o mixta), proyecto (enfoques aplicativos), asistencia, entre otros.

- Con el fin de visualizar las variables más usadas en el sector educativo mencionadas anteriormente, se construyó una nube de palabras (figura 21) con el *software* Word Art.

**Figura 21.** Nube de variables influyentes en el rendimiento académico



**Fuente:** elaboración propia.

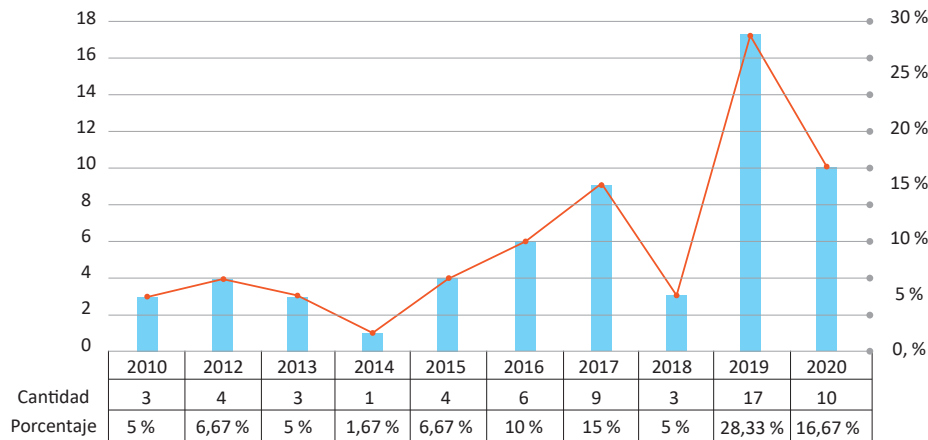


# Capítulo 5

## Aprendizaje no supervisado: investigación referencial

A continuación, se realiza una breve descripción y un análisis detallado de cada uno de los artículos investigados durante el periodo 2010-2020. En este análisis se hace énfasis en los algoritmos de aprendizaje automático no supervisado aplicados, las variables utilizadas en los modelos, la eficiencia de las técnicas o métodos aplicados, el lenguaje de programación o *software* utilizado y los resultados finales de cada estudio. Estos artículos se agruparon por periodos de tiempo —2010 a 2013, 2014 a 2017 y 2018 a 2020), como se muestra en la figura 22.

**Figura 22.** Investigaciones consultadas del campo educación por año de publicación



**Fuente:** elaboración propia.

## 5.1. Investigaciones realizadas entre 2010 y 2013

Shih *et al.* (2010) proponen un algoritmo de agrupamiento simple y sin supervisión para modelos ocultos de Markov (HMM) que pueden sacar a la luz tácticas de aprendizaje de los estudiantes. Las variables utilizadas se basan en el número total de secuencias realizadas y la ganancia de aprendizaje previa a la prueba. Una de las causales principales del rendimiento es el acercamiento a tutores de *software* y la capacidad para aprender a partir de intentos persistentes de resolver problemas. Esta causa sugiere un paradigma para los sistemas de tutoría que enfatiza los intentos y proporciona canales o ejemplos trabajados solo cuando sea estrictamente necesario; sin embargo, hay otras explicaciones factibles y se requiere una exploración más extensa de este tema. La interpretación general de estos resultados es que los estudiantes aprenden más cuando usan tácticas de tipo persistencia, siempre y cuando no adivinen repetidamente. Curiosamente, esta interrelación es independiente de la elección del umbral. La explicación más probable es que las acciones muy cortas o largas resultan en cantidades mayores de información sobre el estudiante, y por tanto las acciones que se reclasifican por pequeños cambios son relativamente poco importantes.

Por su parte, Oyelade *et al.* (2010) describen un sistema para analizar los resultados de los estudiantes de educación superior de una institución privada en Nigeria con base en clústeres. Implementaron el algoritmo de aprendizaje no supervisado K-Mean junto con algoritmos estadísticos estándar para organizar las puntuaciones de los estudiantes de acuerdo con su nivel de rendimiento académico. El fin era que los planificadores académicos pudieran tomar una decisión efectiva, semestre por semestre, y mejorar así los futuros resultados académicos. Además, se eligió la medida de la distancia euclidiana de similitud para ser utilizada en el análisis de las puntuaciones de los estudiantes. El conjunto de datos contiene información sobre setenta y nueve estudiantes de nueve cursos ofrecidos en un semestre; de este modo, se introdujeron variables al modelo como número de estudiantes, puntuaciones del estudiante y número total de cursos de un semestre. Este análisis mostró que veinticinco estudiantes tuvieron un desempeño “muy bueno” (62,22 %), mientras que quince estudiantes tuvieron un desempeño “muy justo” (45,73 %). Los veintinueve estudiantes restantes tuvieron un



desempeño “bueno” (53,03 %). Por tanto, este algoritmo de agrupación sirve como un buen punto de referencia para monitorear la progresión del rendimiento de los estudiantes en la educación superior.

Ramadass y Abhraham (2010) utilizaron mapas autoorganizados (SOM) para entender la toma de decisiones sobre el rendimiento de los estudiantes durante el proceso de admisión. SOM se toma como arquitectura al utilizar las redes neuronales (ANN). La calificación educativa incluye especialización en pregrado (UG), porcentaje de UG y porcentaje de secundaria superior, entrevista personal y experiencia laboral previa e información personal (género, estado (BC/MBC/SC/ST/OC), ubicación y antecedentes familiares (negocio). Se tomó una matriz de ciento dos neuronas y se entrenó con doscientas variables de entrada. Las variables principales utilizadas fueron la calificación educativa que incluye especialización en pregrado (UG), porcentaje de UG y porcentaje de secundaria superior; la elegibilidad que incluye puntaje KMAT/PGCET, entrevista personal y experiencia laboral previa; e información personal que incluye género, estado (BC/MBC/SC/ST/OC), ubicación y antecedentes familiares (negocio). Este modelo sin supervisión alcanza el 80 % de la precisión total. Posteriormente, el modelo se comparó con el resultado original y se encuentra una tasa de error del 20 %. Se ha observado que la puntuación de pregrado, KMAT y antecedentes de comercio cumplen un papel importante en su éxito. Además, el mayor porcentaje de graduados sin mayor comercio, buena puntuación en KMAT con experiencia laboral mejora el éxito de los estudiantes.

Por otro lado, Trandafilu *et al.* (2012) presentan un proceso de análisis exhaustivo realizado sobre los datos de estudiantes a partir de técnicas de aprendizaje automático para descubrir perfiles de estudiantes a partir de datos de evaluación de cursos y para generar asociaciones entre asignaturas en función del desempeño del estudiante. Los autores emplean agrupaciones de maximización de expectativas para dividir a los estudiantes en distintos perfiles que muestran algunas características interesantes cuando se analizan cualitativamente mediante los algoritmos *Apriori* y Árboles de decisión. Utilizan variables como ID del alumno, atributo de evaluación, asistencia, proyecto vinculado y laboratorio de cada curso. Con ellos se busca predecir el rendimiento y el desempeño en las evaluaciones de los cursos futuros y

el número de estudiantes que continuarán los estudios de maestría en la universidad. Como conclusión, al aplicar la agrupación sobre los conjuntos de datos del estudiante, se determinó que muchos de los estudiantes tienen problemas con varios cursos, y que solo el 11 % de ellos logra obtener muy buenos resultados; en este contexto, los autores proponen que la universidad debe intentar integrar la tesis del diploma con las prácticas debido a los resultados relevantes obtenidos en este estudio, que se basan en una exactitud del 97 % del algoritmo *Apriori*.

Sabitha y Mehrotra (2012) proponen un modelo que puede mejorar la búsqueda y la entrega de un objeto de aprendizaje relevante con base en preferencias y en un mayor ranking y agrupación de objetos de aprendizaje; se utilizaron las técnicas de aprendizaje no supervisado *K-Means* y *Self-Organizing Maps* (SOM) para agrupar objetos de aprendizaje personalizados a partir de valores categóricos. Antes de la aplicación, los autores mencionan que las instituciones académicas y de investigación archivan un gran número de documentos como planes de estudio, material de estudio y recursos relacionados con la investigación. Los profesores e investigadores necesitan almacenar y utilizar estos documentos durante un período de tiempo más largo. Para lograrlo se requiere convertir estos recursos educativos en objetos de aprendizaje y almacenarlos de manera estructurada y significativa a través de un sistema de gestión de aprendizaje (LMS). Se enriquece así la enseñanza clásica, y por tanto es necesario implantar un sistema de recomendación en el dominio del aprendizaje electrónico para ampliar la funcionalidad de los sistemas de gestión del aprendizaje con base en normas y proporcionar la recuperación basada en el usuario. Se introducen variables introductorias al modelo como pertinencia temática: tema, subtema, autor; relevancia personal: edad, nivel educativo; pertinencia situacional: tiempo, espacio y geoaprendizaje. El análisis de conjuntos de datos se realizó utilizando el kit de herramientas TANAGRA para así definir un número óptimo de cuatro clústeres. Según los resultados experimentales, los objetos de aprendizaje, junto con los objetos de conocimiento se almacenan en el LMS, y las preferencias del alumno y el perfil del alumno se almacenan en el modelo del alumno. Finalmente, los autores concluyen que al agrupar los

objetos de aprendizaje podemos mejorar los procesos educativos y reducir la dificultad de encontrar un recurso educativo particular para el usuario.

Ramadass y Abraham (2012) hacen uso de técnicas de aprendizaje no supervisado con el fin de predecir cómo mejorar el rendimiento académico de los estudiantes, así como el rango de las instituciones educativas. Diseñaron un sistema controlador de bucle abierto (OLC) con *Self Organizing Map* (SOM); estudiaron su naturaleza y precisión en el campo educativo; y lo adaptaron a un *Kohonen Self Organizing Map* (KSOM). Un KSOM de dos capas está diseñado con diez vectores de entrada que representan los criterios de admisión, y tres unidades competitivas de salida que representan el rendimiento del estudiante durante la admisión sobre su grado. Una vez que los estudiantes de un patrón similar se agrupan en las unidades de salida, se aplica el algoritmo de extracción de reglas para extraer los conocimientos adquiridos en las unidades de salida; estos pueden actuar como reglas de control, y de esta forma las reglas que se extraen están disponibles para controlar el proceso de evaluación del rendimiento del estudiante. Para dicho estudio, se utilizan variables introductorias como puntaje académico, ciudad de vivienda, estudiantes favorecidos, especialización, experiencia laboral. Los resultados experimentales mostraron que el algoritmo de aprendizaje no supervisado KSOM es capaz de producir reglas interpretables con altas capacidades de predicción, y que la precisión de los sistemas se prueba con conjuntos de datos disponibles conocidos. Además, se corroboró que la precisión que permite ese algoritmo es alta y que puede mejorarse aún más al ajustar los parámetros de entrada.

Finalmente, el esquema de evaluación propuesto por los autores proporciona los detalles sobre la capacidad de los estudiantes, y, por tanto, las directivas del instituto/colegio toman conciencia de la actitud de sus estudiantes y les permiten actuar de manera proactiva para guiarlos en un camino adecuado. Por tanto, el sistema de control KSOM hace que la institución reconozca que:

- Los estudiantes de naturaleza similar deben agruparse.
- Los estudiantes que tienen buen puntaje académico logran alta tasa de éxito.

- Los estudiantes de ciudad se desempeñan mejor en comparación que los de áreas rurales.
- Los estudiantes desfavorecidos necesitan atención adicional.
- Los estudiantes que no tienen mayores conocimientos en comercio, matemáticas y otras áreas principales de la educación superior, pero que tienen experiencia laboral, también se desempeñan bien.

Islam y Haque (2012) utilizaron una mezcla o híbrido de algoritmos supervisados y no supervisados —como el algoritmo árbol de decisión y *K-Means*— con el fin de ayudar a los profesores a reducir la tasa de abandono escolar a un nivel significativo y mejorar el rendimiento de los estudiantes; así mismo, con esos dos algoritmos fue posible predecir el GPA del estudiante y tomar medidas pragmáticas para mejorar el rendimiento global de las clases. El algoritmo árbol de decisión se utiliza para entrenar a la máquina y posteriormente utilizar el algoritmo *K-Means* para agrupar los datos de estudiantes en tres clases, alta, media y baja, de acuerdo con su nuevo grado. El nuevo grado se calcula a partir del grado del semestre anterior, proceso que implica una evaluación externa y una evaluación interna. Posteriormente se clasifica al estudiante en una de las tres categorías. Este proyecto logró mejoras en el rendimiento de los estudiantes, así como reducir la proporción de fracaso mediante la adopción de medidas adecuadas en el momento oportuno para mejorar la calidad de la educación.

Aher *et al.* (2013) realizaron una combinación del algoritmo de agrupamiento (*K-Means*) y la regla de asociación *Apriori* para predecir cómo un completo sistema de recomendaciones puede influir en la decisión que tome un estudiante a la hora de escoger un curso. En este caso, los estudiantes son identificados a partir de las variables interés de aprendizaje por el estudiante y tipo de lenguaje de programación escogido. La muestra total de la investigación comprende treinta y ocho estudiantes y ocho cursos: C-programming (C), Visual Basic (VB), Active Server Pages (ASP), Computer Network (CN), Network Engineering (NE), Operating System (OS), Distributed System (DS) y Data Structure (DS-I). Los resultados evidencian que inicialmente existe mucha volatilidad en las decisiones de cada estudiante, así que se decide utilizar las técnicas de clasificación no supervisadas para identificar

patrones poco comunes, y posteriormente se realiza el mismo ejercicio después de hacer una exhaustiva recomendación de los diferentes cursos ofertados, lo que permitió identificar que la percepción de los estudiantes puede verse altamente influida por la información que reciben y por el estímulo inicial de tomar uno de estos cursos.

Balakrishnan (2013) identificó patrones definitorios en el comportamiento del estudiante en su interacción con el MOOC (un curso masivo abierto en línea) y su propensión a permanecer en el curso. Su pretensión era predecir si era probable que un estudiante permaneciera en el curso en una siguiente semana. Los MOOC suelen contener cortos videos de conferencias (de diez a quince minutos de duración), así como exámenes y tareas asignadas para evaluar la comprensión de los estudiantes. En este caso se utilizaron modelos ocultos de Markov (HMM) como medio para entender el comportamiento del estudiante en el tiempo, ya que el estado oculto puede modelar características latentes del estudiante que influyen en su voluntad de perseverar, y luego poder inferirlas de sus interacciones observables con el MOOC. Además, un HMM para MOOC nos permite inferir el comportamiento de un estudiante en el siguiente paso de tiempo con base en su estado anterior y sus acciones actualmente observables.

Dado que hay muchos tipos diferentes de acciones observables, exploramos dos alternativas para crear un modelo compuesto que incorpore múltiples características. En un primer método, los autores construyen una matriz de características multidimensionales de valor continuo para los estudiantes, a través de las secciones de tiempo del curso, y se cuantificó este espacio de características con el algoritmo *K-Means Clustering* o discretización de productos cruzados, en un número discreto de estados observables. En el segundo método, se utilizó un enfoque de conjunto de apilamiento: se entrenaron varios HMM individuales que cada uno considera una sola característica observable. Luego, se pasan los resultados a un regresor logístico que los autores entrenaron para predecir si un estudiante será retenido en el siguiente paso del curso. Como datos introductorios al estudio, tenemos 29 882 estudiantes matriculados, pertenecientes a once conferencias, cada una dividida en varios videos de diez a veinte minutos de duración. Cada estudiante se enfrenta a problemas de práctica de opción múltiple sin clasificar, cuatro

tareas asignadas, y cada uno de ellos tiene asignada una cantidad diferente de problemas de programación. Finalmente, y de acuerdo con los autores, se concluye que el problema del estudiante que nunca participa puede deberse a factores externos; este último fenómeno sugiere que, por la razón que sea, los estudiantes están perdiendo la voluntad de completar el curso.

## 5.2. Investigaciones realizadas entre 2014 y 2017

Durairaj y Vijitha (2014) desarrollaron un modelo de confianza con base en técnicas de minería de datos. Su objetivo era construir una herramienta de gestión estratégica al sistema educativo. Se recopilaban los datos en tiempo real de estudiantes universitarios. Estos datos describen las relaciones entre el comportamiento de aprendizaje de los estudiantes y su rendimiento académico en diferentes calificaciones de asignaturas en términos semestrales (el análisis de rendimiento se llevó a cabo utilizando la herramienta Weka). Así mismo, se realizaron entrevistas a expertos para obtener información necesaria para resolver diferentes problemas de agrupamiento de datos y para construir una base de conocimientos. Inicialmente se usó el algoritmo de agrupamiento *K-Means*, que predijo el porcentaje de aprobado y el porcentaje de fracaso de los estudiantes en general. Estos resultados fueron patentes en un examen en particular. Los resultados pueden ser utilizados para el proceso de toma de decisiones por la dirección de una institución educativa.

Li y Shi (2015) discuten la relación entre los dormitorios de los estudiantes y su rendimiento académico a partir de algoritmos no supervisados como el algoritmo *K-Means*, el cual se utiliza principalmente en el proceso de análisis de este estudio. Para ello, se utilizaron 687 muestras entre estudiantes de segundo y tercer año de tres carreras diferentes, y 171 dormitorios. Además, para la aplicación del modelo se introducen variables como calificaciones académicas y calificaciones extracurriculares. Se obtuvieron cinco clústeres con una medición del 35,32 %. A partir del resultado del análisis de datos con la aplicación java y el *software* Weka, los autores concluyen que las calificaciones de los estudiantes están influenciadas por el arreglo del dormitorio.

Bapu *et al.* (2015), en su proyecto de investigación, utilizaron métodos de minería de datos, aprendizaje automático e inteligencia artificial para monitorear a los estudiantes en cursos de aprendizaje electrónico con el fin de

dar a los educadores mejores herramientas para ayudar a sus estudiantes. Los autores pretenden mejorar el monitoreo de los estudiantes, automatizar algunos de los trabajos de los educadores, consolidar todos los datos generados por un entrenamiento y crear un sistema de tutoría inteligente basado en un innovador motor de comportamiento basado en HTM. El monitoreo de los estudiantes tiene el objetivo de evitar que estos se queden atrás de sus compañeros y deserten, lo que se puede prever con en análisis de datos. Las variables utilizadas fueron frecuencia de inicio de sesión, ultimo inicio de sesión, tiempo que el estudiante estuvo en línea, número de lecciones leídas, número de lecciones descargadas como PDF para leer posteriormente, número de recursos adjuntos a una lección consultada, número de exámenes, crucigramas, asignaciones, etc.; hechos, nota media obtenida, número de temas del foro leídos, número de temas del foro creados y número de respuestas a temas del foro. Los algoritmos ejecutados son expectation maximisation, *Hierarchical clustering*, simple *K-Means*, and *X-means*. La investigación arrojó información sorprendente en parte porque solo se encontraron diferencias cuantitativas, pero no cualitativas, entre la actividad de los estudiantes. Parecía que una sola característica, una especie de índice de su actividad global, sería casi suficiente para describir los datos. Adicionalmente, los clústeres observados para los entrenamientos no mostraron nada más relevante que una simple distinción entre estudiantes activos y menos activos, además de variaciones según el número de clústeres elegidos.

Bhalchandra *et al.* (2016) sugieren algunos enfoques para encontrar patrones en los logs y actuaciones de los estudiantes. Los datos de esta investigación se recolectaron de encuestas y del registro de rendimiento —se manejaron 360 registros y cuarenta y seis campos de datos de los estudiantes—. Los datos son sociales, intelectuales, demográficos, habituales, de salud y económicos. Se implementaron algoritmos de agrupamiento jerárquico utilizando SPSS. Las variables utilizadas fueron código del curso, nombre, sexo, estado civil, edad, domicilio, teléfono móvil, correo electrónico, pasante de grado, nombre del collage de grado, educación del padre, tamaño de la familia, relación familiar, razón para elegir este curso, y algunas ideas en mente como aspectos sociales de los estudiantes, la ventaja económica de los estudiantes para leer detenidamente la educación, los deseos y ambiciones



personales y, finalmente, los recursos disponibles que pueden ayudar a los estudiantes a tener éxito. Estos datos fueron evaluados con el método de aprendizaje no supervisado *Hierarchical Clustering*. Se encontró que, en el primer nivel de agrupación, el entorno familiar del estudiante —información que incluye los ingresos familiares, el apoyo y las relaciones familiares— importa mucho. Los pasatiempos —ver películas entre semana, por ejemplo— y otros aspectos de este tipo tienen por lo general impacto negativo en el rendimiento. Los atributos como la educación de los padres, el trabajo a tiempo parcial, el modo de viaje, el tiempo de autoestudio, el estado civil, y el tiempo con los amigos también afectan el rendimiento. El bajo rendimiento o el fracaso se asocia con la razón para elegir este curso y el tiempo libre con los amigos.

Por su parte, Najdi y Brahim (2016) aplican técnicas de aprendizaje sin supervisión a los datos educativos para la extracción de tipologías y perfiles de estudiantes. Utilizan un algoritmo *K-Means* y lenguaje de programación R con el fin de identificar grupos homogéneos de estudiantes de acuerdo con su rendimiento académico. Además, analizan la duración de los estudios del programa de grado. El conjunto de datos utilizado en este estudio es el conjunto de calificaciones oficiales de los estudiantes para los seis semestres y el grado final de la licenciatura. Se usan las variables grado de la escuela secundaria (antes de inscribirse en el programa universitario), nota del estudiante en el semestre, número de estudiantes matriculados por semestre, promedio general de estudiantes con todos los años de estudios, calificación global de la licenciatura y número de matriculados con todos los años de estudios en el programa de licenciatura.

Este grupo de variables arrojó tres clústeres principales que tienen las siguientes características:

- Clúster 1: Este grupo se compone del 28 % de la población. Se puede describir como el grupo de estudiantes “débiles”. Los estudiantes tienen un bajo nivel de notas promedio para todos los semestres y un mayor número de inscripciones.
- Clúster 2: Este grupo está conformado por el mayor número de estudiantes del grupo, representa el 50 % de los estudiantes graduados.



- Clúster 3: Es el grupo de estudiantes “más exitosos”; representa el 22 % de los estudiantes graduados. La información que arrojan sus variables es favorable y constante a lo largo de los semestres.

Por tanto, este análisis de agrupación ha producido resultados relevantes y ayudará a los responsables de la toma de decisiones de las universidades a obtener una visión más profunda de las características de aprendizaje de los estudiantes, supervisar el rendimiento de los estudiantes durante cada semestre del programa de grado y adaptar los enfoques pedagógicos y estrategias para mejorar el rendimiento y la calidad de la educación en las universidades marroquíes.

Battaglia *et al.* (2016) comienzan su análisis describiendo un método no-jerárquico, el algoritmo *K-Means* que como herramienta para separar fácilmente a los estudiantes de una clase en grupos que pueden ser reconocidos y caracterizados por rasgos comunes en sus respuestas con base en un cuestionario de tipo mixto (es decir, con preguntas abiertas y cerradas). Después, describen un método de análisis diferente, basado en la agrupación jerárquica (conocido como *Hierarchical Clustering*) que ayuda al investigador a encontrar grupos de estudiantes visualizados a través de un dendograma. Este muestra fácilmente los vínculos entre parejas y grupos de estudiantes con base en distancias mutuas. La variable principal serán las respuestas obtenidas a partir del cuestionario. Con el fin de estudiar la coherencia de los resultados obtenidos mediante el uso de métodos jerárquicos y no jerárquicos, se compararon los resultados entre sí. Se encontró mucha similitud en los clústeres generados; sin embargo, algunos de ellos (resultado del uso del algoritmo *K-Means*) están más divididos, y pueden ser, por tanto, mejor caracterizados.

Alonso y Marcela (2016) proponen un modelo basado en el tipo de aprendizaje para entender de qué manera los estudiantes, especialmente los que cursan las asignaturas asociadas con el área de matemáticas, aprenden. Su investigación se basa en los conceptos y teorías de Kolb, quien plantea que el aprendizaje es un fenómeno que se da a partir de la experiencia, las necesidades y las metas del sujeto. Este fenómeno, además, comprende cuatro etapas: la experiencia concreta, la observación reflexiva, la conceptualización

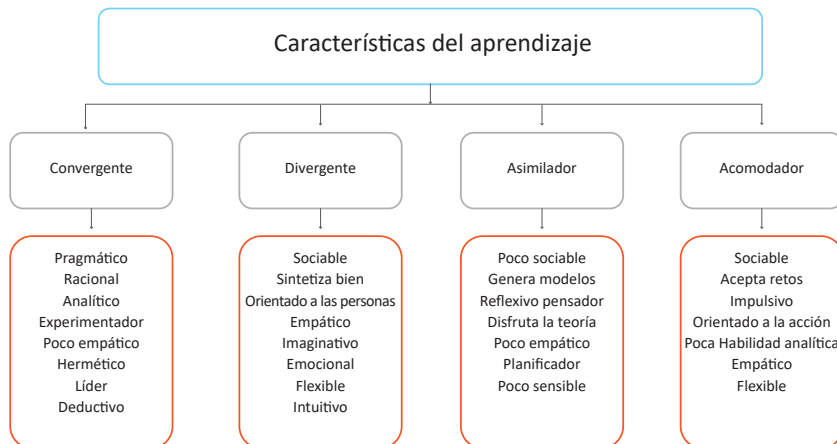
abstracta y, por último, la experimentación activa, junto con otros aspectos de clasificación. La muestra es de los estudiantes de primer semestre de los programas de la facultad de ingeniería, entre 2013 y 2014, de la Universidad Francisco de Paula Santander-Ocaña. Se introdujeron variables como estilo de aprendizaje, pérdida de alguna materia y promedio del semestre.

El estudio utiliza técnicas de aprendizaje no supervisado y un algoritmo *K-Means* para Weka, llamado simpleKmean. Los resultados muestran cuatro clústeres: el primero comprende un 57 % de información al estilo de aprendizaje de Felder y Silverman multiestilo, el 47 %, de información con el estilo Kolb divergente, y un 90 % de información con desempeño bajo, un estilo más adecuado para personas del área de humanidades y las artes; en el segundo clúster predomina el estilo de Felder y Silverman, con 26 %, el estilo acomodador de Kolb, con 41 %, y de desempeño alto con un 67 %, pues el aprendizaje es más visual, hay planeación para resolver problemas; el tercer clúster comprende un 27 % de información con el estilo Felder y Silverman, un 64 % con el estilo de Kolb divergente, y un 64 % de información de desempeño alto gracias al estilo más visual, además de capacidades de análisis de situaciones para la toma de decisiones; y, finalmente, el cuarto clúster comprende un 21 % de información con el estilo Felder y Silverman multiestilo, un 16 % con el estilo de Kolb divergente, y un 31 % de desempeño bajo por la utilización combinada de diversas técnicas que no permiten que la concentración y la disposición del estudiante se focalice (figura 23).

Hua Leong *et al.* (2016) modelan estados ocultos de compromiso de los estudiantes que trabajan en ejercicios de programación en un sistema de tutoría inteligente. Utilizan variables como pose de cabeza, número de pulsaciones de tecla, duración máxima de la pausa de pulsación de tecla, cantidad de errores de compilación, duración fuera de la pantalla y cantidad de intentos de compilación. Usaron técnicas de aprendizaje automático sin supervisión, como el algoritmo *Hidden Markov Model* (HMM) —que se usa con el fin de modelar los niveles de participación de los estudiantes y se ajusta a los registros de acción y sensor de los estudiantes—; así, fue posible traducir las acciones y posturas de la cabeza de los estudiantes en estados de compromiso ocultos y probabilidades de transición. Finalmente, los resultados muestran que los estados de compromiso de los estudiantes se pueden

agrupar en tres distintos: el estado comprometido (estado 1), con una probabilidad del 72 %; el estado inicial (estado 2), con 68 %; y el estado desconectado (estado 3), con 31 %. Los autores encontraron que los estudiantes que están comprometidos tienden a permanecer comprometidos, mientras que los estudiantes que están desconectados son más propensos a cambiar a los otros dos estados. Sugieren al potencial para el uso de intervenciones de aprendizaje como pista para llevar a los estudiantes a un estado más duradero y comprometido. Sin embargo, según los autores, esta hipótesis requerirá más experimentos para ser validada.

**Figura 23.** Características del estilo de aprendizaje de Kolb



**Fuente:** adaptado de Alonso y Marcela (2016).

En su estudio, Guo *et al.* (2016) presentan una arquitectura de aprendizaje profundo para predecir el rendimiento de los estudiantes. Esta arquitectura aprovecha los datos no etiquetados y prediseñados en capas ocultas de entidades de forma inteligente. Los autores usan aprendizaje no supervisado en un discreto autocodificador automático disperso, y datos sin etiquetar para descubrir características mediante un algoritmo de retropropagación. Así, les fue posible minimizar el error de reconstrucción de los datos. En una etapa posterior, se usan algoritmos de aprendizaje supervisado para ajustar los parámetros, se utilizan algoritmos como Naive Bayes, MLP (Multilayer Perceptron), SVM (Support Vector Machine) y SPPN (Student Performance

Prediction Network); el rendimiento promedio obtenido de cada algoritmo fue de 20,7%; 41,4%; 48,4% y 77,2%, respectivamente.

En esta investigación se usaron variables relacionadas con antecedentes y datos demográficos de los estudiantes: sexo, edad, estado de salud, estado familiar, etc.; datos de estudios anteriores: puntaje de ingreso a la escuela secundaria, GPA de la escuela primaria, etc.; datos de evaluación de la escuela: tipo de escuela, clasificación de la escuela, etc.; datos del estudio: puntaje de cada curso en la escuela secundaria (examen intermedio, examen final, promedio); y datos personales: personalidad, atención, datos relacionados con la psicología. En general, el algoritmo propuesto (SPPN) se utiliza para modelar la función de actividad de las neuronas ocultas con el fin de inicializar todas las redes completas a valores apropiados en la variable de calificación final, que comprende cinco rangos diferentes, entre 0 y 100.

Rana y Garg (2016), por su parte, evalúan el rendimiento del estudiante a través de diferentes algoritmos de clasificación como *K-Means* y *Hierarchical Clustering Algorithm*. Los autores usaron estos algoritmos sobre el conjunto de datos (primeros cinco estudiantes) de los estudiantes de BE (tecnología de la información) del tercer semestre de UIET para la asignatura electrónica digital. Usaron las siguientes variables: rendimiento de estudiantes en la calificación en pruebas en la sesión 1, la sesión 2, un quiz y el examen de universidad. Esta información se agrupó previamente de la siguiente forma: los estudiantes con puntaje de 80% o superior son excelentes en sus estudios; los estudiantes con puntaje de entre 50% y 79% son buenos; y los estudiantes con menos del 50% son pobres. Esta clasificación permite hacer una predicción clara sobre cuáles estudiantes necesitan trabajar duro. La utilización de los algoritmos busca identificar patrones de mejoramiento en el rendimiento o conductas que puedan afectar la forma de presentación las pruebas.

El propósito principal del artículo de Imtiyaz y Singhal (2016) es identificar las respuestas no genuinas, y establecer un modelo de evaluación eficaz que utilice el análisis de pares de configuraciones. Diseñaron una encuesta que permite a los estudiantes evaluar individualmente a cada maestro. Esta encuesta supone que cada estudiante evalúa sin sesgo y con total honestidad. El objetivo de este trabajo es identificar las respuestas imprecisas o fraudulentas dadas por los estudiantes, y eliminar la incertidumbre en los

resultados finales. Este trabajo combina la técnica de aprendizaje automático de enfoque de agrupamiento *K-Means* con una teoría matemática difusa de análisis de conjuntos para evaluar la calidad de la enseñanza en institutos de educación superior. Así, sería posible eliminar la incertidumbre en los resultados y garantizar que solo se tengan en cuenta respuestas auténticas para los cálculos. Las variables utilizadas fueron número de estudiantes, calificación brindada, nota alcanzada y coherencia en la respuesta.

En su trabajo, Najdi y Er-Raha (2016) intentan extraer tipologías y perfiles de estudiantes de posgrado, con base en resultados educativos y su relación con el tiempo de estudio hasta la titulación. Los autores utilizaron el algoritmo de aprendizaje no supervisado *K-Means*. Se utilizó además una herramienta que consta de dos partes: interfaz de usuario web y el servidor web. Estos dos componentes están controlados por el código escrito con R y Shiny. Este análisis se realizó en un conjunto de datos correspondientes a las calificaciones del estudiante, además del número de inscripciones para los semestres del programa de grado. Más específicamente, se analizaron dos mil registros de estudiantes y las variables notas, talentos, conocimientos prácticos en un campo en particular, antecedentes familiares, escuela. Después de ajustar y someter los datos, se crearon tres clústeres: el primero corresponde al 28 % de la población, se trata de un grupo de estudiantes débiles, con bajo rendimiento en lo que respecta a resultados educativos de cabina y la duración del estudio; el segundo clúster está conformado por el mayor número de estudiantes del grupo, representa al 50 % de los estudiantes graduados, que tienen un buen nivel de longitudes de estudio, con resultados medios; y, finalmente, el tercer clúster corresponde al grupo de estudiantes más exitosos, representa al 22 % de los estudiantes graduados, los que tienen muy buen índice de resultados educativos de cabina y buenas longitudes de estudio. La herramienta web implementada ha permitido agrupar a los estudiantes de forma accesible, y los decisores universitarios tomar decisiones pensadas. Permite seleccionar variables, elegir el número de clústeres e interactuar con la visualización resultante que dentro de la institución educativa permitirá mejorar la comprensión de las diferentes características de aprendizaje de los estudiantes graduados y podría utilizarse

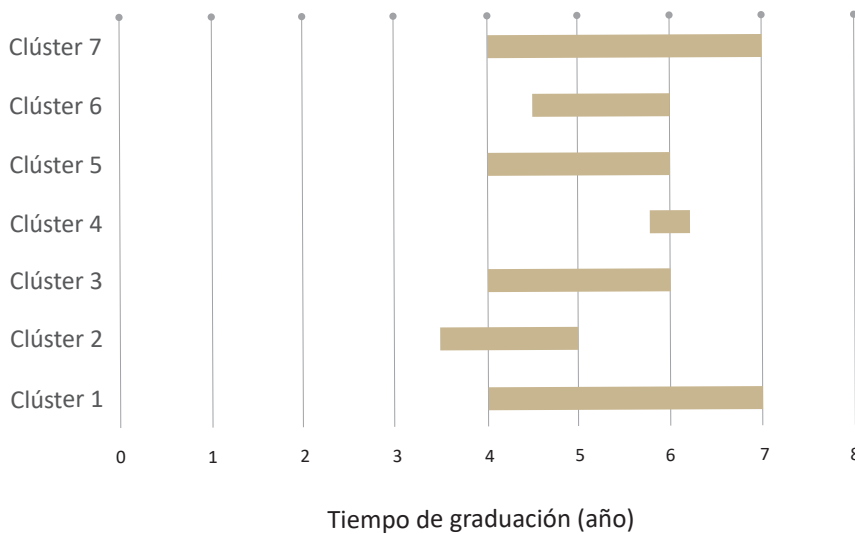
para adaptar los enfoques y estrategias de enseñanza de acuerdo con los perfiles de estudiantes identificados.

Karnad *et al.* (2017) sugieren un modelo de agrupación de puntajes ponderados que se puede utilizar para evaluar y validar la resolución de problemas y las habilidades de pensamiento de los estudiantes en función de su rendimiento académico. Los autores usaron el algoritmo de aprendizaje no supervisado *K-Means*, y agruparon a los estudiantes en tres clústeres: clúster de alta cualificación (HSC), clúster de habilidades adecuadas (ASC) y clúster de habilidades inadecuadas (ISC). Para esta investigación se usaron las variables número único de identificación del estudiante; y calificaciones y puntajes obtenidos en cada curso optado por el estudiante en los diferentes semestres en su segundo y tercer año de ingeniería. Posteriormente, los datos se refinaron mediante la clasificación de las empresas a las que accedieron los estudiantes: nivel 1, nivel 2 y estudiantes que no recibieron una oferta de trabajo. Esta categorización se construyó con factores como el perfil del trabajo, la marca de la empresa y el paquete salarial ofrecido. La precisión del modelo para identificar a los estudiantes que requirieron entrenamiento de destrezas fue del 95,83 %, y la precisión del modelo para identificar al estudiante que requirió entrenamiento de destrezas fue del 88,46 %. Finalmente, el modelo de agrupación identificó a los estudiantes que tenían pocas habilidades y requerían capacitación especial para aumentar su empleabilidad. Se comprobó que la mayoría de los estudiantes tenían aptitudes adecuadas, pero necesitaban capacitación para obtener mejores oportunidades de empleo.

Cahaya *et al.* (2017) hacen uso del algoritmo de aprendizaje no supervisado *K-Medoids* para predecir la duración de un tiempo de estudio o permanencia de los estudiantes en las universidades. Este algoritmo clasificó el tiempo de estudio de los estudiantes hasta la graduación mediante el uso de veinticinco características: las calificaciones de veinticinco asignaturas en los primeros cuatro semestres. También se utilizaron transcripciones de doscientos cuarenta estudiantes graduados, y para este caso, se considera el tiempo de graduación para etiquetar el clúster. Las variables introductorias fueron notas de veinticinco asignaturas, semestre y designación al finalizar el cuarto semestre. Se obtuvieron como resultado siete clústeres, con una precisión de predicción media del 99,58 % y una desviación estándar de 0,0358 (figura

24). Del estudio se concluye que el abandono académico o la ampliación de la finalización de la tesis son los otros factores que contribuyen al intervalo de tiempo de graduación en cada clúster. Sin embargo, muchos factores son desconocidos incluso al final del estudio, como la existencia de privilegio universitario para que un estudiante tome un permiso académico, la posibilidad de retomar las materias perdidas por los estudiantes para finalizar, y el factor de condición social del estudiante. Sin embargo, estos factores pueden influir en el tiempo de graduación del estudiante, pero no se consideran en este estudio. Por tanto, para trabajos futuros, los autores consideran uno o más de los factores como las características para agrupar a los estudiantes.

**Figura 24.** Rango de tiempo de graduación para el resultado de predicción en cada clúster



**Fuente:** Cahaya *et al.* (2017).

Liu y D'Aquin (2017) presentan un estudio que aplica el aprendizaje no supervisado para identificar cómo las características demográficas de los estudiantes y su participación en actividades de aprendizaje en línea pueden afectar su rendimiento académico. Para ello, utilizaron el algoritmo de agrupación *K-prototypes*, el cual integra *K-Means* y *K-Modes*. Este método usa la medida de distancia euclidiana cuadrada en atributos numéricos y la medida de disimilitud simple coincidente en atributos categóricos para encontrar



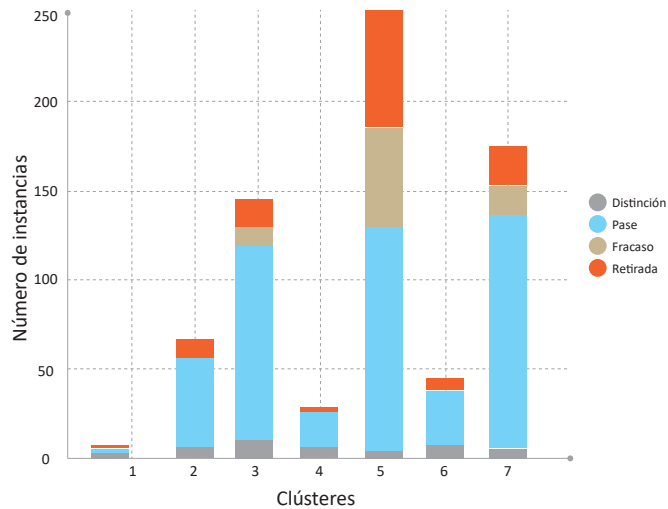
los centroides más cercanos de objetos, pues el objetivo de este estudio es utilizar tanto los datos demográficos de los estudiantes como sus interacciones con un sistema de entorno de aprendizaje virtual; esta información se compone de atributos numéricos y categóricos. Para llevar a cabo el estudio, se introducen variables al modelo como género del estudiante, región geográfica donde vivió el estudiante mientras tomaba el curso, nivel educativo más alto que un estudiante obtuvo al ingresar al curso, nivel de privación del lugar donde vivió el estudiante durante el curso, rango de edad del estudiante y discapacidad.

El conjunto de datos de aprendizaje abierto contiene información de 748 estudiantes que han tomado el curso en dos términos diferentes. De entre ellos, treinta y seis estudiantes han tomado el curso dos veces, algunos porque fracasaron y otros se han retirado del curso por primera vez. Los resultados clasifican a los estudiantes en cuatro niveles: distinción, pase, fracaso y retirada. Se concluyen que los grupos “exitosos” presentan un mayor porcentaje dado que los estudiantes viven en las zonas más privilegiadas que los grupos “fracasados”. Además, los grupos “exitosos” comprenden un mayor porcentaje de estudiantes ya que estos tienen mejores niveles de educación superior que los grupos “fracasados” (figura 25). Conocer a estos grupos de estudiantes con resultados de aprendizaje exitosos o pobres puede ayudar al profesorado a diseñar cursos en línea que se adapten a las necesidades de diferentes estudiantes. Esta configuración puede ayudar a los estudiantes en la selección de cursos en línea que son apropiados para ellos.

Hidayah *et al.* (2016) proponen una evaluación de metacognición en el ambiente de aprendizaje del aula parcialmente automática. Se implementa un mecanismo de aprendizaje no supervisado, y los autores ratifican que es necesario supervisar el desarrollo de tales habilidades cognitivas para asegurar los logros de los estudiantes universitarios. El estudio se realiza a través de un esquema que se divide en tres etapas, como se ve en la figura 26.

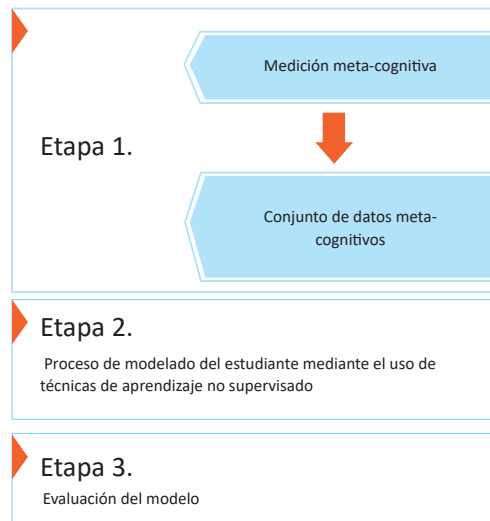


**Figura 25.** Distribución de los resultados finales de los estudiantes



**Fuente:** Liu y D'Aquin (2017).

**Figura 26.** Esquema propuesto para la evaluación automática de la metacognición del estudiante en el ambiente del aula



**Fuente:** Hidayah *et al.* (2016).

En la segunda etapa se realizó el proceso de modelado del estudiante con la ayuda de dos algoritmos de aprendizaje no supervisado: *K-Means* y EM. Este

último es un método de agrupamiento probabilístico. Se recogieron hasta 105 instancias como conjunto de datos, y se usaron ocho atributos que consisten en subcomponentes metacognitivos: conocimiento declarativo (DK), conocimiento procesal (PK), conocimiento condicional (CK), planificación (P), estrategias de gestión de la información (IMS), monitoreo de comprensión (CM), estrategias de depuración (DS) y evaluación (E). El experimento demostró que, para el conjunto de datos metacognitivos, el valor óptimo de clústeres es dos, sobre todo teniendo en cuenta el uso de ambos algoritmos de agrupamiento. La K óptima se eligió en función del error, que es el promedio de la distancia observada entre la muestra de prueba con centroides de clúster. El valor de la función de error disminuyó cuando el número de grupos era de más de dos. Finalmente, los dos clústeres obtenidos identifican dos grupos de estudiantes: uno que funciona bien en cuanto a su conciencia metacognitiva, y el otro que necesita más orientación y asesoramiento para el logro de mejores resultados y así evitar fracasos.

Yamasari *et al.* (2017) proponen una etapa de extracción de características que se implementa en el preprocesamiento de datos. También proponen aplicar el algoritmo no supervisado *Fuzzy C-Means* (FCM), que permite que la extracción se realice sobre la base de la categoría y la taxonomía de Bloom, mediante la recopilación de datos de los estudiantes. La muestra de esta investigación toma a ochenta y cinco estudiantes de primaria. Según lo especificado por la taxonomía de Bloom, hay tres niveles de competencia del estudiante: C1, C2 y C3. Cada uno de esos niveles se basa en tres preguntas de evaluación. Al mismo tiempo, estas se utilizan como variables introductorias para el modelo: puntuación, tiempo de ejecución y ayudante usado. En este caso, la puntuación es la respuesta a cada pregunta; el tiempo de ejecución es el tiempo tomado para responder cada pregunta; y ayudante usado es la condición que un estudiante utiliza para acceder a la ayuda. Los resultados experimentales muestran que la extracción de características con base en la categoría (CBE\_FCM) y la taxonomía de Bloom (BTBE\_FCM) posibilitan el mejoramiento del proceso de agrupación con base en el rendimiento del estudiante, el nivel de precisión y el tiempo de ejecución. En este sentido, el FCM original tiene la menor precisión —aproximadamente 64,7%—, mientras que BTBE\_FCM es el de mayor precisión —alrededor de

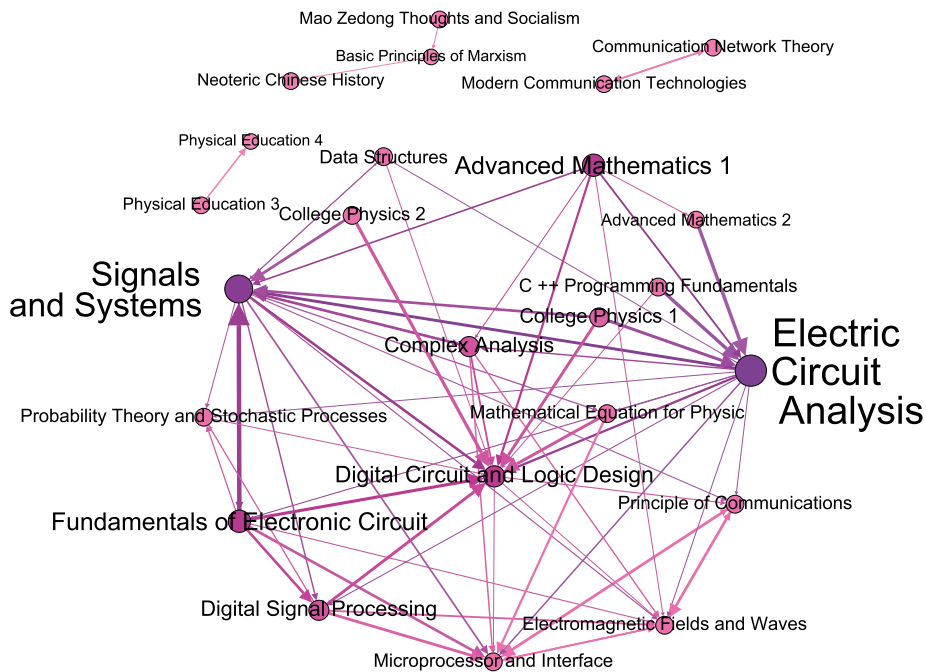
69,4%—. Finalmente, uno de los objetivos de este estudio es determinar el número de grupos de estudiantes que deben desarrollarse de acuerdo con su comprensión de las materias. Esta información es necesaria para los profesores porque, al usar esta información, pueden observar la agrupación de los logros del estudiante en función de sus características. En cuanto a trabajos futuros, los autores mencionan que los métodos propuestos pueden mejorarse al añadir más peso a cada característica con base en el nivel de correlación del rendimiento del estudiante.

Zhang, Yin *et al.* (2017) proponen un modelo basado en tres características esenciales: apoyo-confianza-interés. La investigación analiza el aprendizaje de los estudiantes de los grados 9 y 11 en la carrera de Ingeniería de la Comunicación de la Universidad de Beijing de Correos y Telecomunicaciones. Analiza las características de distribución de puntuación y las deficiencias clave de los métodos de segmentación de puntuación existentes (variable: GPA) a través del algoritmo *K-Means*. Los resultados obtenidos muestran que cursos como Análisis de Circuitos Eléctricos y Señales y Sistemas se rigen bajo las reglas de asociación más relevantes dentro del proceso de aprendizaje de los estudiantes con miras al desarrollo de su futuro desarrollo profesional. El contexto nos indica que estos dos cursos son los cursos básicos de Ingeniería de la Comunicación, y se imparten en el segundo y tercer trimestre, respectivamente. Estos dos cursos pueden reflejar, en gran medida, si los estudiantes tienen el talento requerido para esta profesión. Al mismo tiempo, el conocimiento en estos dos cursos básicos se utilizará con frecuencia en los cursos profesionales posteriores, por lo que las puntuaciones en estos dos cursos afectarán las puntuaciones de los cursos posteriores (figura 27).

Zhang *et al.* (2017) presentan un enfoque no supervisado para caracterizar los comportamientos de aprendizaje de los estudiantes en un entorno de aprendizaje abierto. Caracterizando entonces los comportamientos primarios de cada grupo en relación con la capacidad de los estudiantes para construir modelos correctos, así como sus ganancias de aprendizaje derivadas de sus puntajes pre y posttest. El estudio se realizó en un aula de sexto grado (niños de los once a los trece años). Se recolectó información diariamente durante tres semanas, en la clase de ciencias —una hora de clase—, y se

usaron las variables tiempo (en segundos) que un estudiante dedica a leer páginas de contenido de dominio o CT, el porcentaje de ediciones de modelos conceptuales o computacionales en las acciones totales de un estudiante, el porcentaje de las acciones de edición del modelo conceptual o computacional admitidas por un estudiante, Las transiciones de una categoría de actividad de aprendizaje a otra, y el algoritmo de *K-Means*. Los resultados son el sustento de un estudio combinado de los comportamientos y del rendimiento de los alumnos que demuestra que los grupos son indicativos del rendimiento de los estudiantes en el modelado y las ganancias de aprendizaje pre y post. Se demuestra que los estudiantes que tenían una buena comprensión de las tareas de aprendizaje y resolvieron problemas de manera estratégica se desempeñaron mejor que los estudiantes *ad hoc*. Finalmente, se mostró que la mayoría de los estudiantes mejoraron de manera autónoma sus comportamientos de aprendizaje y el uso de estrategias en unidades posteriores.

**Figura 27.** Resultado de las reglas de asociación de la minería evaluada en los diferentes cursos ofertados



**Fuente:** Zhang *et al.* (2017).

Rana y Garg (2017) discuten importantes algoritmos de clasificación y agrupamiento de *Machine Learning* que se pueden aplicar para evaluar y predecir el desempeño académico de un estudiante. El rendimiento de los estudiantes de Electrónica Digital del Instituto Universitario de Ingeniería y Tecnología (UIET) y de la Universidad de Panjab (PU) se calcula con algoritmos de aprendizaje sin supervisión como *K-Means* y *Hierarchical Clustering Algorithms*. Además, para el aprendizaje supervisado, se aplica el algoritmo M5P. También se realiza una comparación entre *Classification Via Clustering* y *Classification Via Regression* con la herramienta de WEKA. Se trata, en síntesis, de la utilización de dos enfoques para analizar y predecir el desempeño de BE (Tecnología de la Información) de estudiantes de tercer semestre (figura 28). Los datos académicos de 58 estudiantes del conjunto de datos se distribuyen en ocho atributos o variables introductorias como grado nominal, asistencia, institución, área urbana, área rural, notas totales, notas altas y notas bajas. De esta manera, fue posible establecer dos clústeres para ambos algoritmos de agrupamiento. Hay que anotar que la precisión de *K-Means* disminuye cuando el algoritmo pasa de ejecutar ocho atributos a seis; sin embargo, pero el caso de *Hierarchical Clustering*, la precisión aumenta.

**Figura 28.** Comparación de los dos enfoques

| Características /nombres      |           |                           |                                 |                                       |
|-------------------------------|-----------|---------------------------|---------------------------------|---------------------------------------|
| Instancia                     | Atributos | Modo de prueba            | Tiempo para construir el modelo | Exactitud tomando todas las variables |
| ↓                             |           |                           |                                 |                                       |
| Classification Via Clústering |           |                           |                                 |                                       |
| 58                            | 8         | Validación cruzada K-Fold | 0                               | 34,48 %                               |
| ↓                             |           |                           |                                 |                                       |
| Classification Via Regression |           |                           |                                 |                                       |
| 58                            | 8         | Validación cruzada K-Fold | 0,04                            | 93,10 %                               |

**Fuente:** Rana y Garg (2017).

Según los resultados experimentales, la precisión en *Classification Via Regression* es más que la de *Classification Via Clustering*. Sin embargo, el tiempo necesario para construir un modelo para el conjunto de datos dado en *Classification Via Clustering* es menor (0 seg.) que el necesario para *Classification*

*Via Regression* (0,04 seg.). Según los autores, la investigación podría extenderse a varias materias los estudios de los estudiantes en su cuarto año.

Nájera *et al.* (2017) presentan un estudio experimental que toma como objeto a estudiantes asociados y maestros de tutoría, y analizan sus habilidades y afinidades con métodos de agrupamiento de k-medios, maximización de expectativas (EM), y lo más lejos primero (FF). El estudio recoge datos de 1199 estudiantes universitarios y treinta y cinco profesores. Los resultados alcanzaron el 100% de la compatibilidad entre los clústeres. Se utilizó la maximización de la expectativa y el primero más lejano. Las variables usadas fueron edad, sexo, estado civil, comunicación (habilidad para la presentación, discusión o argumentación); interpersonal (conceptos fácilmente aplicados, valores, herramientas en la realidad natural y social); intrapersonal (hacer planes para lograr metas realistas); autodirección; digital (uso de herramientas tecnológicas para facilitar el proceso de aprendizaje); autonomía (capacidad de automotivación a pesar de la dificultad de ciertas actividades); y afinidades (disfrute activo de redes sociales). El objetivo de la investigación era identificar grupos similares de estudiantes a ser asignados para tutoría con habilidades y afinidades similares.

Los resultados experimentales muestran que EM y FF crearon clústeres con 100% de compatibilidad entre tutores y estudiantes independientemente del clúster, mientras que *K-Means* fue capaz de crear solo dos clústeres con compatibilidad inferior al 100%. Así mismo, *K-Means* mostró una baja dispersión para crear clústeres para estudiantes y tutores. La propuesta final considera que es indispensable formar grupos con afinidades y habilidades comunes entre estudiantes y docentes, y que la formación de grupos así podría mejorar el rendimiento de los estudiantes si se asigna un tutor adecuado a cada grupo.

En su artículo, Iam-On y Boongoen (2017) pretendieron establecer el enfoque de agrupación como una guía práctica para explorar las categorías y características de los estudiantes. En él proporcionan una visión descriptiva del rendimiento académico de los estudiantes admitidos entre 2007 y 2009. El propósito de esta investigación es descubrir conocimientos, patrones interesantes y relaciones que contribuyen al abandono escolar. La información administrada se clasificó a través de las siguientes variables: número de

identificación del estudiante, nivel de grado del estudiante, título de licenciatura, número de identificación del departamento académico, género del estudiante, número de identificación de la provincia de origen del estudiante, año de admisión del estudiante, código que indica el tipo de admisión, código que representa el estado del estudiante, fecha de terminación —graduado o deserción—, número de identificación de un solicitante de ingreso a la universidad, calificación general de la escuela del estudiante, año académico en el que el estudiante toma un curso específico, semestre en el que el estudiante toma un curso específico, código que denota curso académico, calificación del curso y descripción del tipo de entrada. Una vez obtenida la colección inicial de datos de los estudiantes, se procesa de tal manera que se resuelven los errores y los valores faltantes.

Para este estudio se utilizaron técnicas de agrupamiento no supervisadas como *K-Means*. Se revela que existen dos grupos al momento de la admisión a la universidad: uno corresponde a solicitantes con buenas calificaciones escolares, y el otro, a aquellos con perfiles de calificaciones de moderadas a pobres. Los autores traducen la existencia de los dos grupos de la siguiente manera: los estudiantes con un buen historial académico-escolar siguen teniendo buenos resultados en el primer año, la mayoría de los solicitantes con bajo rendimiento escolar pueden sufrir abandono escolar después del primer año, y algunos pueden adaptarse al sistema académico universitario y sobrevivir.

### 5.3. Investigaciones realizadas entre 2018 y 2020

Sajjadi *et al.* (2018) utilizan técnicas de agrupamiento no supervisadas y el algoritmo *K-Means* para predecir el estado de graduación de las carreras declaradas en cinco departamentos de la California State University Northridge (CSUN), así como el algoritmo logistic regression, que se basa en un número mínimo de cursos de división inferior en cada especialización. Además, hicieron uso de los clústeres detectados para identificar cursos de cuello de botella ocultos. Los datos introductorios abarcan un período de diez años, entre 2004 y 2014, y a un total de 9088 registros de estudiantes. Los datos además conciernen únicamente a los cursos necesarios para cada carrera: Administración, Marketing, Derecho Empresarial, Ingeniería Civil



y Eléctrica. Se usan las variables graduación, número de semestres en la mayor, número de créditos para la mayor, número de créditos de transferencia obtenidos y los nombres de los cursos.

Como resultado, los autores identifican grupos fuertemente predictivos en cada uno de los cinco departamentos. El número óptimo de clústeres fue de dos en todos los departamentos. Además, se comparó el desempeño de un clasificador de regresión logística entrenado en el conjunto completo de características, con el desempeño de un clasificador que utilizó información de comembresía de los clústeres en una tarea de clasificación. Así, fue posible predecir si el estudiante se había graduado con esa especialización. En este sentido, los clasificadores basados en clúster destacan sorprendentemente bien con respecto a los modelos de regresión logística, tal como se evidencia a continuación, en las puntuaciones de precisión. Finalmente, se concluye que los estudiantes en el mismo grupo tienden a abandonar al mismo tiempo después de obtener las mismas calificaciones en los mismos cursos

Howard (2018) utilizó un marco de aprendizaje automático no supervisado para explorar un conjunto de datos relacionados con la producción evaluada por los estudiantes de un programa en negocios durante cuatro semestres sucesivos. El enfoque de este trabajo son los datos, y está también respaldado por pruebas para comprender el alcance del aprendizaje de los estudiantes de una clase de tributación de clase ejecutiva como una herramienta para fines de presentación de informes de acreditación. El aprendizaje emiratí de la tributación es importante porque los Emiratos Árabes Unidos, para ese entonces, estaban introduciendo un régimen tributario más completo, con un impuesto del 5% del IVA que estaría vigente desde 2018. Para este estudio, los autores hicieron uso de técnicas como *Hierarchical Cluster Analysis* (HCA), *Multi-Dimensional Scaling* (MDS) y *Self-Organizing Map* (SOM), y de variables introductorias como proyecto, grupo, trabajo, tiempo, estudiante, experiencia, gobierno, riqueza, ingresos, provisión, pago y servicio, impuesto sobre la renta, país, deducción, medio de crédito, entre otras. Los resultados del análisis de datos asistido por ordenador señalaron cuatro factores: dos relacionados con los impuestos y dos relacionados con el aprendizaje. Estos factores son la teoría de impuestos y la práctica fiscal, junto con el aprendizaje práctico y teórico. Según el autor, la investigación



motivó un paradigma teórico fundamentado que explicaba el alcance del conocimiento adquirido por el alumno de la clase de tributación.

Guleriay y Sood (2018) realizaron la clasificación de los datos del estudiante en términos de una evaluación interna hecha por los miembros de la facultad y la visualización de un conjunto de datos educativos con Python. Usaron múltiples modelos de predicción de clasificación con base en datos y resultados comparativos de modelos de clasificación. Se comparan los modelos de clasificación que utilizan Python y que pueden transformar el aprendizaje. Se considera que el modelo usado, SVM, es que tiene la mejor medida de precisión para el análisis predictivo y la clasificación del rendimiento general de la clase. Se conoce como máquinas de vectores de soporte. Sin embargo, a este le siguen los algoritmos CART, un clasificador de árbol de decisión y KNN, el algoritmo K-Vecino más cercano para el reconocimiento de patrones, el cual se utiliza para el análisis de clasificación y regresión. Las variables tipo independiente utilizadas para este estudio fueron comprensión, actitud y asistencia para tres parámetros en particular: fundamentos de TI, sistemas operativos y cursos *dot net*. El conjunto de datos de entrada muestra que los estudiantes han obtenido calificaciones internas a partir de estos tres parámetros para los cursos, y que los datos pueden ser desestructurados o semiestructurados, lo que genera incertidumbre en cuanto a la predicción del desempeño general de los estudiantes y su máximo interés por determinada asignatura.

Martinez-Rodriguez *et al.* (2019) evaluaron la correlación lineal de forma independiente: los hábitos de estudio y la autoestima ante el desempeño escolar. El estudio se realizó con 153 estudiantes universitarios de nuevo ingreso, pertenecientes al área administrativa y social de la Universidad Autónoma de Baja California. Es una propuesta que se centra en perfiles académicos creados automáticamente mediante el uso del algoritmo *K-Means*, el cual tiene un enfoque cuantitativo. El objetivo principal es estimar la relación existente entre factores conductuales de personalidad del estudiante y su rendimiento académico, específicamente los factores de hábitos de estudio y autoestima. Las variables de estudio evaluadas frente a los hábitos de estudio fueron: administración del tiempo, concentración, memoria, motivación para estudiar, presentación de evaluaciones y relaciones interpersonales.

En cuanto a la autoestima, se evaluaron escuela, familia y general (nivel de confianza al tomar decisiones). Finalmente se evidencia que el 80 % de los estudiantes con un perfil académico adecuado obtuvieron un desempeño bueno y sobresaliente, y solo el 68 % y el 61 % de los estudiantes con un perfil académico medio y deficiente obtuvieron un desempeño *bueno y sobresaliente*, respectivamente.

Shetty *et al.* (2019) buscan predecir las probabilidades de que un estudiante consiga empleo en una empresa con base en sus resultados académicos, logros, experiencia laboral (pasantías) y otras características relevantes. Se hace un esfuerzo por entrenar y realizar el estudio empírico a través múltiples algoritmos de aprendizaje automático supervisado: regresión logística, Naive Bayes, Support Vector Machine, K-Nearest Neighbor y Decision Tree. Los modelos de aprendizaje automático seleccionados también se comparan por su eficiencia mediante el empleo de las técnicas de extracción de características supervisadas y no supervisadas, como PCA y LDA. Se diseñó un formulario de Google para recopilar los datos de todos los estudiantes de programas como Informática e Ingeniería, y el Departamento de Colocación de la Universidad REVA, entre 2015 y 2019. Los resultados muestran que el modelo de árbol de decisión con  $K = 10$  para validación cruzada, y el método no supervisado PCA han superado a todos los otros modelos con una precisión de 72,83 %, apoyo satisfactorio y la recuperación durante la experimentación.

El objetivo principal del trabajo de Bey *et al.* (2019) es introducir un proceso de tres fases y un estudio de caso en el que se utilicen técnicas de agrupamiento no supervisadas para identificar automáticamente el comportamiento de programación de los alumnos. El estudio de caso tiene lugar en un curso de programación de Shell en el que se analizan datos de cien estudiantes exitosos sobre sus trayectorias de comportamiento, a través del algoritmo *Gaussian Mixture Models for Clustering*. Como resultado, los autores lograron extraer una lista de características que ayudan a mejorar la calidad del proceso de identificación automática de perfiles de los estudiantes y algunas trayectorias de comportamiento de estudiantes en relación con su desempeño en el examen final. Estas características son percibidas a través de las siguientes variables: número de envíos, tiempo medio entre dos envíos,

número medio de cambios y porcentaje de errores sintácticos, desviación estándar temporal y desviación típica del código. Este proceso arrojó tres clústeres que permiten identificar que los estudiantes exitosos pueden pasar tiempo para pensar en la solución de un problema antes de enviar la primera versión de su programa, y luego realizar regularmente cambios menores. Estos estudiantes adoptan este comportamiento durante toda la duración del curso. Por otro lado, los estudiantes con menor rendimiento regularmente ejecutan muchas presentaciones sin pasar mucho tiempo entre la codificación. De forma más general, los resultados muestran que las cuatro primeras secuencias representan el comportamiento del 71,70 % de los estudiantes en este clúster (alto rendimiento). Además, el 78,73 % de los estudiantes del segundo clúster siguen dos trayectorias particulares: omiten el tiempo necesario para analizar la codificación previa antes del primer envío.

De La Hoz *et al.* (2019) desarrollaron una metodología para identificar y pronosticar diferentes usuarios en ambientes netamente virtuales enfocados en educación. Analizan el uso de la plataforma por parte de los estudiantes y cómo influye en el rendimiento en los exámenes. Para esto, se aplicaron técnicas de aprendizaje automático supervisado y no supervisado como algoritmo *Fuzzy K-Means*, K-vecinos cercanos y PAM (partición alrededor de medoides). Además, se usaron las variables tiempo de estudio de los recursos educativos obligatorios, número de veces que el estudiante ingresó a la plataforma, tiempo de estudio en aspectos definidos como complementarios, el desempeño del estudiante en el examen parcial de investigación de operaciones, el desempeño del usuario en el examen final de Investigación de operaciones, el nivel de conocimiento final alcanzado por el usuario. El análisis permitió clasificar a los usuarios en tres categorías —para la clasificación se usó la metodología *Fuzzy K-Means* con el objetivo de delimitar zonas de transición entre los diferentes niveles de conocimiento—: el perfil grupo uno (nivel alto) es ocupado por usuarios con resultados promedio en los exámenes y que utilizan poco tiempo de estudio para aprender las asignaturas relacionadas con los objetivos globales; el perfil grupo dos (nivel medio) lo ocupan usuarios con un alto rendimiento en el resultado promedio obtenido en el examen, estos dedican el tiempo mínimo para aprender las asignaturas de estudio relacionadas con los objetivos globales; y el perfil grupo tres (nivel

bajo) lo ocupan usuarios que presentaron un bajo rendimiento, con notas bajas en los exámenes. Del total de 403 muestras, se seleccionaron 258 como datos de entrenamiento y 145 para evaluación. Los resultados del proceso de agrupación evidencian cómo el algoritmo KNN exhibe los mejores resultados de predicción con un 91 % de precisión, mejora los clasificadores de discriminación lineal y los de forma cuadrática con una precisión de 85 % y 84 %, respectivamente.

Aissaoui *et al.* (2019) proponen un enfoque para identificar el estilo de aprendizaje automáticamente con base en los comportamientos de los estudiantes existentes y utilizando técnicas de minería de uso web y algoritmos de aprendizaje automático. La información de los alumnos se extrajo del archivo de registro de un sistema de aprendizaje electrónico. Para evaluar el rendimiento del clasificador se utilizó el método de matriz de confusión. Esta información se clasificó en las siguientes variables: video, presentaciones, demostraciones, ejercicios, PDF, tareas, anuncios, referencias, ejemplos, material practico, foros, lista de temas, imágenes, gráficos, email y secuencia agrupadas en dieciséis combinaciones de estilos de aprendizaje basadas en el modelo de estilo de aprendizaje de Felder y Silverman (procesamiento (activo/reflexivo), percepción (sensible/intuitivo), entrada (visual/verbal), comprensión (secuencial/global). Inicialmente se utilizó el algoritmo *K-Modes* para agrupar las secuencias de los alumnos en combinaciones de estilos de aprendizaje, y luego se utilizó el ingenuo clasificador de Bayes para predecir el estilo de aprendizaje de un estudiante en tiempo real. La precisión del modelo aplicado en R es de 89,06 %. Así, se concluyó que el tipo de aprendizaje influye de manera gradual en la comprensión de los estudiantes, y que los contenidos programáticos de las asignaturas deben tener en cuenta la información y el tipo de aprendizaje más adecuado para llegar a los estudiantes.

Joshika y Rajeshwari (2019) utilizan la técnica de aprendizaje no supervisado *K-Means Clustering* a través de la herramienta Weka para identificar el efecto de tener una enfermedad en el rendimiento académico de los estudiantes: al estar enfermos carecerán de concentración en los estudios y, si no lo están, su desempeño nunca se verá afectado. Para esto, se analizan algunos de los factores o variables de entrada que afectan sus estudios: sexo del estudiante, edad del estudiante, tiempo libre después de la escuela, estado de

salud actual y grado del primer periodo. La salud se toma como eje X y el grado como eje Y; el K-centroide calculado es 11. Los autores clasifican la salud de los estudiantes de 1 a 5. Los estudiantes cuyas calificaciones son 1 a 3 no podrán concentrarse mucho en sus estudios porque están enfermos, y estos, además, no están utilizando el tiempo libre en casa para estudiar; sin embargo, comparativamente hay una pequeña diferencia entre los estudiantes cuyos padres van a trabajar y no pueden cuidar de ellos. Estos son los estudiantes que más se ven afectados. Por otro lado, los resultados muestran que los estudiantes que pueden estudiar durante su tiempo libre en casa también han obtenido una puntuación alta (por encima de 11). Esto significa que los estudiantes han hecho un uso eficiente del tiempo libre mediante el aprendizaje de nuevos conocimientos. Finalmente se concluye que es muy importante que los estudiantes utilicen su tiempo libre de manera efectiva para que puedan obtener un conocimiento más general. Además, es importante que los estudiantes se mantengan sanos para hacer su trabajo rutinario de una manera efectiva y entusiasta.

Cheng y Shwe (2019) plantean una encuesta que se lleva a cabo rutinariamente en estudiantes cerca de terminar un título de pregrado y contiene información sobre las diversas experiencias durante el aprendizaje en Cal Poly Pomona. Este estudio agrupa los resultados de aprendizaje de los estudiantes y distingue a aquellos con mayor similitud a través de variables como nombre del estudiante, tiempo pasado en la universidad, participación en el campus, historial de empleo, calificaciones con licencia, autoevaluación de los resultados del estudiante y el algoritmo de aprendizaje no supervisado Hierarchical Clustering. Un total de tres clústeres principales y tres pares de resultados similares fueron identificados entre los once resultados de los estudiantes —cada uno con 512 respuestas— utilizando el modelo de clúster jerárquico. Específicamente, los resultados del estudiante a&e, h&j, y f&i se consideran mejor al estar juntos, pues a partir de esta relación se generan estrategias para mejorar los resultados del estudiante. Se espera que estos resultados proporcionen información adicional al profesorado y al administrador a medida que desarrollen cursos que traten las necesidades de los estudiantes. Adicionalmente, estos resultados obtenidos pueden ser fusionados para un enfoque dirigido hacia el diseño de cursos optimizados.

Nguyen y Vo (2019) proponen un método de agrupamiento de datos educativos basado en la recuperación de información con el fin de identificar la lista de estudiantes en problemas del ranking de todos los estudiantes cuando se los agrupo por tiempo. Este estudio usa las técnicas de aprendizaje no supervisado como *K-Means*, *Kernel K-Means* y *Temporal Data Clustering (DTM) Kernel*. El método resultante, *DTM\_kernel*, puede proporcionar mejores clústeres para la recuperación final con base en la relevancia, el examen de los estudiantes con todo su rendimiento de estudio en el tiempo; sin embargo, el método *Kernel K-Means* no tiene en cuenta ningún concepto temporal en su proceso de aprendizaje, mientras que el *DTM* genera una cadena temporal de clústeres. Por otro lado, *K-Means* del núcleo devuelve una partición óptima que es esférica en el espacio de funciones y se convierte en arbitraria en el espacio de datos. Por otro lado, *DTM* tiene asociaciones objeto-clúster en cada ranura de tiempo. Dicho estudio hace uso de variables introductorias sobre el estado de los estudiantes: graduarse, estudiando, primera advertencia, segunda advertencia, parada de estudio y número total de estudiantes. Finalmente, los autores utilizan recall, precision y F-measure entre 0 y 1 para registrar los resultados experimentales. Para el conjunto de datos “2006”, hay una mejora de alrededor del 13 % en recall, alrededor del 3 % en precision, y alrededor del 5 % en F-measures; para el conjunto de datos “2007”, cerca del 6 % en recall, cerca del 4 % en precision, y cerca del 5 % en F-measure; para el conjunto de datos “2008”, cerca del 3 % en recall, cerca del 3 % en precisión, y alrededor del 4 % de media.

Chau *et al.* (2019) proponen un método de análisis de datos educativos a través de la creación de un modelo que priorice datos temporales en el rendimiento del estudio de cada estudiante. Así, es posible formar un patrón a partir de los grupos a los que se pertenecen a lo largo del tiempo. La propuesta tiene que ver con que, durante el transcurso del desarrollo profesional de cualquier estudiante puede pasar por múltiples estados que no se miden comúnmente con el paso del tiempo. Los autores proponen un método de análisis de clústeres temporales basado en *DTM* para la identificación temprana de estudiantes en problemas. Este método compara los algoritmos *LNA* y *K-Means*, y utiliza los datos de calificaciones de los estudiantes de tercer año matriculados entre 2006 y 2008, correspondientes al rendimiento del



estudio de cuarenta y tres cursos requeridos por el programa, organizados en las siguientes variables: graduarse, estudiando, promedio, estudiantes por encima del promedio, estudiantes por debajo del promedio, estudiantes con un rendimiento hacia arriba, hacia abajo o sin cambios desde el punto anterior en el tiempo al siguiente. Este resultado puede darnos valores de *recall* y F-medida más altos que aquellos de diferentes métodos que no tomaron en cuenta la conexión temporal entre las actuaciones de estudio de los estudiantes. También se confirma que el método propuesto puede proporcionar una solución eficaz a la tarea, y a la vez puede preparar y dar apoyo valioso a estos estudiantes tan pronto como sea posible para su éxito final en el estudio.

El objetivo principal del estudio de Shidhaganti y Patil (2019) es que sea más fácil para los estudiantes de un instituto conectarse de manera virtual a sus clases. En esta investigación se propone una arquitectura para construir una interfaz de chat donde los individuos de ideas afines, agrupados con base en el algoritmo de mapas autoorganizados SOM y *K-Means*, puedan interactuar. Para el caso particular del estudio planteado, se mostró que SOM eran más efectivo en comparación con *K-Means* debido a que SOM fue capaz de clasificar a los estudiantes en nueve grupos, en comparación con *K-Means* agrupamiento: en este caso, solo cinco grupos de estudiantes fueron reconocidos. Esta identificación de estudiantes se realizó a partir de dos variables: el interés académico del estudiante y su promedio de calificaciones. Así, fue posible diseñar una aplicación de uso institucional que clasifique a los estudiantes y ayude a los docentes de la institución a gestionar de mejor manera el proceso educativo y a medir el rendimiento de las clases. De esta manera, los clústeres obtenidos ayudan a los estudiantes no solo a conectarse entre grupos de su similitud, sino también a otros grupos que pueden diferir solo por unos pocos atributos.

Shukla y Malviya (2019) tienen como objetivo construir un modelo mejorado de clasificación y predicción del rendimiento académico de los estudiantes en el campo de la minería de datos educativos. Para llevar a cabo el estudio, se aplican diferentes técnicas de agrupamiento y clasificación de aprendizaje automático; el primer paso es aplicar los algoritmos canopy, EM, *Farthest-First*, *Hierarchical* clusterer and Simple *K-Means* en los conjuntos de datos del estudiante, y comparar el rendimiento de varias técnicas

de agrupamiento con el fin de identificar el de mejor precisión. El segundo paso es aplicar el algoritmo Naive Bayes para definir la etiqueta de clase de los grupos agrupados y así predecir la etiqueta de clase de las nuevas instancias de los conjuntos de datos de los estudiantes. La metodología utilizada comprende cinco etapas: recopilación de datos, procesamiento previo de datos, agrupación de datos, clasificación de datos e interpretación de los resultados. En este proceso, se hace uso de variables introductorias como la información sobre el progreso académico y la información sobre el progreso de la colocación. Estos datos se recopilan de los estudiantes de diversos cursos con diferentes atributos como aptitud, técnica, gestión, comunicación, entrenamiento, cursos vacacionales y rendimiento. Entre los resultados, la técnica de agrupamiento EM se identificó como la mejor técnica, con 92,5 % de precisión, para lograr clasificar los grupos de instancias de los estudiantes. Esta valiosa información mejora el procesamiento de técnicas Naive Bayes y, como resultado, su precisión aumentó de 95,99 % a 98,75 %. Por tanto, el enfoque combinado de agrupación EM y clasificación Naive Bayes se añade para modificar el modelo con algunas nuevas etapas. Según los autores, este modelo modificado en el artículo ha mejorado la eficiencia en la clasificación y predicción de las posibilidades de colocación de los estudiantes durante la selección del campus. Se concluye que la minería de reglas de asociación y otros lenguajes de aprendizaje automático, técnicas y herramientas pueden utilizarse para aplicar este modelo a una gran escala de datos de los estudiantes.

Tibay *et al.* (2019) describen el funcionamiento óptimo de las escuelas a partir del enfoque DEA (análisis envolvente de datos) que mide la eficiencia del desempeño de las organizaciones para identificar la eficiencia del desempeño de la dirección escolar (SBM) de una división que comprende las escuelas primarias y secundarias, y que está bajo el Departamento de Educación (DepEdde) en Filipinas. Las escuelas eficientes se utilizaron como punto de referencia para la mejora de las escuelas ineficientes: estas últimas se someten a un proceso de agrupación de acuerdo con características similares. Se utilizó el algoritmo de agrupación de *K-Means* con el fin de agrupar a las escuelas según su perfil respectivo, mediante variables de entrada como liderazgo y gobernanza; currículo e instrucción; responsabilidad



y mejora continua; y gestión de recursos. Con base en investigaciones bibliotecarias, libros, revistas, artículos, tesis, disertaciones y publicaciones en línea, se demostró que el enfoque DEA proporciona los resultados de la distribución a escala: el 87,18 % de las escuelas pueden incrementar su desempeño al disminuir el costo de los insumos, mientras que el 12,82 % de las escuelas operaban en óptimas condiciones. Finalmente, se demuestra que el 77 % de las escuelas que tienen altos resultados de SBM, solo el 36,67 % son eficientes y el 63,33 % son ineficientes; esto solo demuestra que el que las escuelas tengan altos resultados de SBM no significa que estén operando de manera eficiente.

Vital *et al.* (2019) analizan el desempeño del estudiante con algoritmos de aprendizaje automático (HMM y K-medios) estadísticos y no supervisados. Los autores pretenden identificar otras variables adicionales a través del agrupamiento, sin importar la causa de la aprobación y el fracaso de los estudiantes. Este estudio tiene en cuenta diferentes factores como el tamaño de la familia, el consumo de alcohol en los días laborables y fines de semana, dirección (rural/urbana), sexo (hombre/mujer) y la regularidad del estudiante. Se utilizaron las siguientes variables educativas y sociales: escuela del estudiante, sexo del estudiante, edad del estudiante, tipo de dirección del hogar del estudiante, estado de convivencia de los padres, educación de la madre, trabajo de la madre, educación del padre, trabajo del padre, tutor del estudiante, tamaño de la familia, calidad de las relaciones familiares, razón para elegir la escuela, tiempo de viaje de la casa a la escuela, tiempo de estudio semanal, número de fallas pasadas de clase, apoyo escolar extraescolar, educación familiar (apoyo), actividades extracurriculares, clases extra pagadas, acceso a Internet en el hogar, guardería asistida, quiere recibir educación superior, relaciones románticas, tiempo libre después de la escuela, asistir-fuera-con-amigos, consumo de alcohol durante el fin de semana, consumo de alcohol en el día laboral, estado de salud actual, número de ausencias escolares, calificaciones en el primer período, calificaciones en el segundo período, calificación final, calificación y resultado. El análisis muestra que de la mayoría de los estudiantes que fracasaron en el pasado, el 12,7 % (50) tuvo solo un fracaso, y el 8,4 % tuvo más de un fracaso. Así como el estudio de clúster *K-Means* da los buenos resultados y el análisis se basa en

sus centroides. Las proyecciones del estudio de conglomerados jerárquicos describen que los principales factores combinados son el tamaño de la familia, el día de trabajo, el consumo de alcohol, el fin de semana, la dirección (rural o urbana), el sexo (masculino o femenino) y el número de ausencias.

Ciubotariu y Crivei (2019) examinaron y analizaron dos modelos de aprendizaje automático no supervisado, mapas autoorganizados (SOM) y minería de reglas de asociación relacional (RAR) en el contexto del análisis de conjuntos de datos educativos para predecir el rendimiento académico de los estudiantes y detectar relaciones ocultas entre las calificaciones recibidas por los estudiantes durante el semestre y su categoría de examen final en una disciplina académica determinada. Analizaron, además, si los modelos de aprendizaje no supervisados pueden revelar alguna información sobre la calidad de los procesos educativos. Dado que no es fácil predecir la calificación exacta del examen final para un estudiante teniendo en cuenta la incertidumbre en los procesos de aprendizaje y evaluación, los autores consideraron en este trabajo cuatro categorías de calificaciones finales: (1) excelente (que se señala en el estudio como *E* y que representa los grados finales 9 y 10); (2) bueno (que se señala como *G* y representa los grados finales 7 y 8); (3) satisfactorio (que se señala como *S* y representa los grados finales 5 y 6); y (4) no satisfactorio (que se señala como *F* y representa el grado final 4).

El conjunto de datos contiene las calificaciones obtenidas por 905 estudiantes en un curso de pregrado de Informática en la Universidad de Babes-Bolyai. Los datos fueron recogidos durante seis años académicos (2011-2017) en las sesiones ordinarias y de nuevo durante exámenes. El estudio comprende cuatro atributos o variables de entrada que representan las puntuaciones obtenidas por los alumnos en varias evaluaciones durante el semestre académico: puntuación de seminario (*a1*), puntuación de proyecto (*a2*), puntuación de estado de proyecto (*a3*) y puntuación de prueba escrita (*a4*). Los resultados experimentales muestran una precisión de los modelos de entre el 68 % y el 71 %. El algoritmo RAR fue el de mayor potencial, ya que fue capaz de expresar patrones interesantes en los conjuntos de datos académicos, lo que permite una mejor visión del problema de la predicción del rendimiento académico. Además, se pudo concluir que las calificaciones recibidas por los estudiantes durante el estudio pueden ser relevantes para

predecir su rendimiento final. Sin embargo, se observaron varios valores atípicos en el conjunto de datos. Estos casos anómalos pueden deberse a (1) un pequeño número de evaluaciones de los estudiantes durante el semestre (atributos); (2) el proceso de aprendizaje de los estudiantes, que no es continuo durante el semestre académico; (3) la diferencia entre las normas de evaluación de los instructores del laboratorio y las actividades del seminario. En consecuencia, un mayor número de evaluaciones durante el semestre académico sería útil para estimular a los estudiantes a estudiar durante el semestre y no solo para el examen final.

El trabajo de Ahmad *et al.* (2019) tenía como fin identificar las características más significativas de los atributos de los estudiantes del archivo de registro con el algoritmo de aprendizaje no supervisado Principal Component Analysis (PCA). Inicialmente se seleccionaron ciertas características como vista del curso, notas, ejercicios, ejemplos y asignaciones con PCA, para ser alimentadas en el mapa autoorganizado (SOM) y así agrupar a los estudiantes en función del comportamiento a través de las siguientes variables: presentar las obras según lo solicitado; ver y descargar la información sobre la tarea, contiene la información del entorno de aprendizaje, contiene la información del entorno de aprendizaje, permitir ver y descargar la información sobre el ejercicio, contiene la información del entorno de aprendizaje, recurso, permitir al alumno enviar el funciona como solicitud, permitir ver y descargar material de aprendizaje, ver la discusión que se produce entre el miembro de la clase y los profesores, permitir ver y descargar la información sobre el ejemplo, permitir ver y descargar la información sobre el cuestionario, permitir que el estudiante envíe el funciona como solicitud, permitir ver y descargar la información relativa a la prueba. Se asignaron actividades por semanas y se hizo seguimiento a los estudiantes periódicamente. El resultado muestra que la técnica es capaz de agrupar el comportamiento de navegación del estudiante en dos grupos de clúster: alto comportamiento de navegación y bajo comportamiento de navegación. De esta forma es fácil identificar que los estudiantes agrupados en el clúster de alto comportamiento tenían un interés más amplio en las temáticas planteadas, trataban aspectos externos y profundizaban su aprendizaje de forma constante, casi el opuesto al clúster de bajo comportamiento.

Sobnath *et al.* (2019) realizan una investigación acerca de las universidades y colegios del Reino Unido. Estas acogen cada año a unos treinta mil estudiantes con necesidades especiales, y el estudio demuestra que la tasa de abandono escolar de los estudiantes con discapacidad es mucho mayor: del 31,5 % frente al 12,3% de los estudiantes sin discapacidades. Por esto, los autores proponen MALSEND, una plataforma conceptual de inteligencia humano-máquina (HMI) que utiliza algoritmos de aprendizaje para acomodar y analizar conjuntos de datos de las características de las actividades de las universidades para descubrir tendencias en el rendimiento con respecto a las áreas temáticas para estudiantes autistas, disléxicos y estudiantes con trastorno hiperactivo con déficit de atención. Para este propósito, se utilizaron algoritmos como *K-Means* para identificar conglomerados y patrones insospechados; reducción de la dimensionalidad o análisis principal de componentes (PCA). Para reducir el número de variables significativas al simplificar los datos sin perder información se usó Python. Finalmente, se utilizó otro método de aprendizaje no supervisado conocido como algoritmo de aprendizaje de reglas de asociación para descubrir relaciones interesantes entre otros atributos.

Las variables tenidas en cuenta fueron rango de edad, sexo, estado (tiempo completo, tiempo parcial, estudiantes a distancia) y tipo de discapacidad de aprendizaje (autista, TDAH, disléxico, disgrafía). Además, se incluyen el tipo de ingreso (básico, nivel A, diploma), el nivel de los estudiantes (puntos y asignaturas), calificaciones del módulo, número de concursos, número de créditos, tipo de módulo, código y descripción del curso, código del módulo y nivel (pregrado o posgrado). Además, las calificaciones de 1er, 2do, 3er año o resultados de posgrado y la información de los egresados (trayectoria profesional, puesto de trabajo después de la graduación). Finalmente, según los autores se espera que la plataforma de aprendizaje automático genere nuevos conocimientos mediante la identificación de patrones que podrían ayudar a resolver algunos de los desafíos sociales como las altas tasas de abandono escolar, la baja empleabilidad o la depresión entre los estudiantes. De hecho, un análisis más profundo de estos datos puede llevar a identificar los factores de éxito social y a evaluar las fortalezas y debilidades de un estudiante; se podría entender su aprendizaje y su desarrollo emocional para

introducirllos lentamente en entornos sociales más complejos y variados a lo largo del tiempo.

Srinivas y Rajendran (2019) pretenden identificar las fortalezas, debilidades, oportunidades y amenazas actuales (FODA) de la universidad de California mediante la integración de cuatro técnicas diferentes: modelado de temas, análisis de sentimientos, causa raíz y análisis. Este trabajo utiliza el modelo de aprendizaje no supervisado *Latent Dirichlet Allocation* (LDA). El objetivo de este artículo es mejorar la satisfacción de los estudiantes mediante la identificación de la fuerza, la debilidad, la oportunidad y la amenaza de una universidad a través de la minería de conocimiento de las reseñas de estudiantes en línea. El enfoque propuesto utilizó un rastreador web para extraer miles de archivos de revisión de estudiantes disponibles públicamente de fuentes en línea. Las variables utilizadas fueron tasa de graduación, puntajes promedio del SAT de los estudiantes admitidos, matrícula, inscripción total de estudiantes y proporción de estudiantes por docente y tipo de universidad. Además, se identificaron doce temas significativos que pueden tener implicaciones gerenciales en los órganos de gobierno de la universidad —por ejemplo, ayudarles a tomar decisiones de planificación para mejorar la satisfacción de los estudiantes, la reputación universitaria y las tasas de inscripción y retención—. En este estudio se da gran valor a los líderes de la universidad, ya que ofrece un resumen de rendimiento eficiente y económico de la universidad y sus competidores desde la perspectiva del estudiante.

Santoso y Yulia (2019) presentan un estudio que pretende modelar el desempeño de los estudiantes matriculados en la universidad. Para esto, los autores utilizaron dos algoritmos de extracción de datos. En primer lugar, se utilizó una tarea descriptiva basada en el algoritmo *K-Means* para seleccionar varios grupos de estudiantes. En segundo lugar, una tarea de clasificación apoyó dos técnicas de clasificación conocidas como árbol de decisión y Naive Bayes, para predecir la deserción debido al bajo rendimiento en los primeros cuatro semestres de un estudiante. La recolección de datos comprende un archivo plano de 55 atributos y 1665 registros de estudiantes admitidos y matriculados en los programas de ingeniería de sistemas y electrónica. Se usaron las siguientes variables introductorias a los modelos: estudiantes admitidos, semestres o matrículas académicas, información socioeconómica,

deserción de estudiantes, relación de los estudiantes universitarios con las pruebas de acceso y los resultados del éxito, resultado de las pruebas de admisión (lenguaje, inglés, matemáticas y lógica), año académico, período de admisión del estudiante, programa en el que está matriculado, situación académica del estudiante (bloqueo académico debido a un bajo rendimiento académico y sin bloqueo académico); y número de créditos académicos registrados, aprobados, perdidos, cancelados y fallidos.

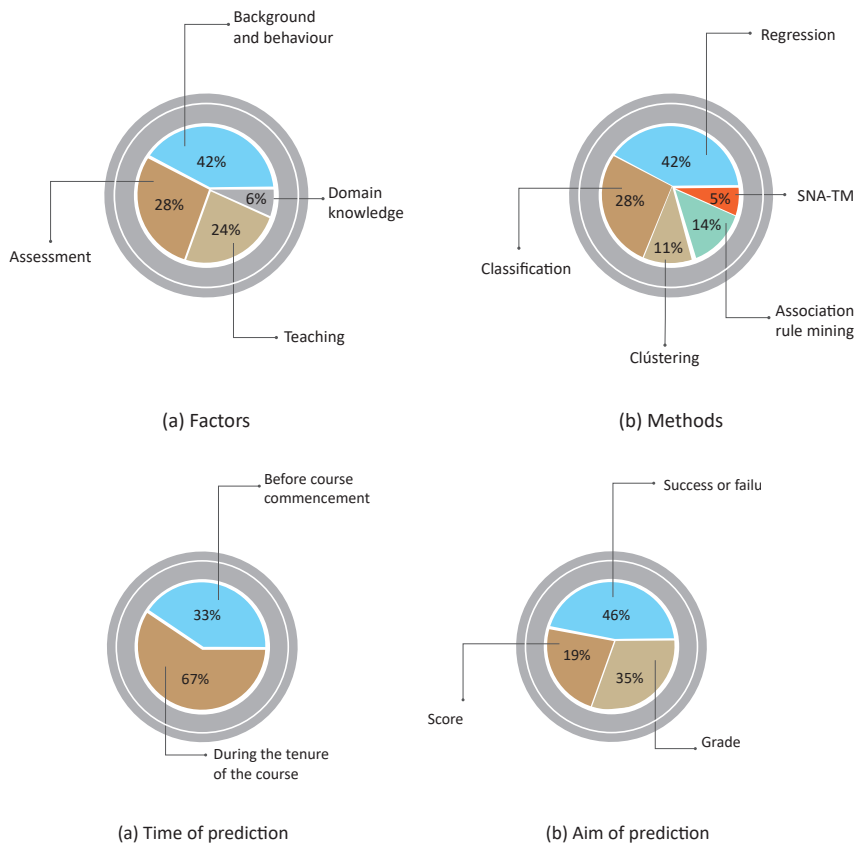
El objetivo de la aplicación del algoritmo *K-Means* fue analizar la población estudiantil de la universidad para identificar características similares entre los grupos. Fue interesante establecer que algunas características socioeconómicas iniciales permitían definir algunos perfiles o grupos: por ejemplo, los grupos de mayor rendimiento académico en los resultados de la prueba de conocimiento se encontraron en las escuelas con bajo estatus socioeconómico. Finalmente, los resultados experimentales muestran que la predicción del abandono escolar mejora, y el rendimiento del estudiante se supervisa cuando se añaden los datos de la inscripción académica anterior. Además, se observó que algoritmo del árbol de decisiones presenta el mayor número de predicciones y un mejor desempeño con la adición del registro académico del semestre anterior; así mismo, al revisar el área bajo la curva, el algoritmo Naive Bayes presenta un buen desempeño con un área mayor a 0,9 en comparación con el algoritmo del árbol de decisiones.

Khan y Ghosh (2020) realizan un análisis del rendimiento de los estudiantes en el aprendizaje en el aula. Se centran en identificar predictores, métodos utilizados para dicha identificación, el tiempo y el objetivo de la predicción. El estado del arte de esta investigación contiene una revisión de ciento cuarenta estudios en esta área cuyo tema es la razón social de la entidad educativa. En cuanto a la metodología, se recolectaron encuestas que contenían información relevante sobre la predicción de grados a corto plazo. Las predicciones asociadas con las encuestas incluyen artículos de investigación publicados solo en inglés y en la base de datos Google Scholar. En ese proceso, se seleccionaron únicamente los estudios relevantes, las variables clave para escoger los artículos fueron: Desempeño en cursos similares, desempeño en prerrequisito y evaluación interna. Se encuentra que la mayoría de artículos que intentan predecir el rendimiento de los estudiantes

se centran en analizar variables que predicen el rendimiento a partir del momento en que ingresa el estudiante a una institución educativa, pero solo el 33 % de estos estudios son capaces de predecir el rendimiento de los estudiantes antes del inicio del curso (figura 29).

Además, la mayoría de los estudios de modelización (46 %) prefieren clasificar el rendimiento a partir de los parámetros *éxito* o *fracaso*. La eficiencia de tal clasificación es generalmente alta debido a un menor número de etiquetas de clase. Algunos de ellos incluso lograron más del 90 % de precisión antes de la dicción. Aun así, muchos de estos estudios ignoran la importante preocupación del desequilibrio de clases.

**Figura 29.** Porcentaje de diversos factores y métodos utilizados en el análisis del rendimiento de los estudiantes y porcentaje de estudios de predicción del desempeño según su objetivo y hora de predicción



**Fuente:** Khan y Ghosh (2020).



Ahmad *et al.* (2020) presentan un sistema interactivo que permite comprender los patrones de comportamiento relacionados con el éxito y el riesgo de los estudiantes. Utilizan datos recopilados de bases de datos institucionales y se propone un modelo temporal innovador con datos de los estudiantes. El modelo se llama FIRST (*Finding Interesting Stories about Students*) y usa un algoritmo de aprendizaje no supervisado *K-Means* para identificar patrones de comportamiento estudiantil y luego mapear ese comportamiento en el rendimiento académico. El modelo se ejecutó con las siguientes variables de entrada: antecedentes con información demográfica (tipo de ciudadanía, género, edad actual); semestre (créditos cursados, créditos aprobados y créditos perdidos); y salida (fecha de graduación, tiempo promedio de graduación y promedio acumulado). Seis profesionales y profesores que ya están familiarizados con múltiples herramientas de análisis de datos y puntajes de riesgo ayudaron a los investigadores. Estos ayudantes seleccionaron un grupo específico de estudiantes con alta eficacia en la presentación opiniones y actitudes, principalmente en debates de grupo y conversaciones dinámicas. Encontraron que para la característica de ingeniería *Progresión del curso a través de semestres*, los estudiantes exitosos son los más abundantes (90,27%), y el nivel medio en términos de rendimiento de los cursos aumenta constantemente.

Rastrollo-Guerrero *et al.* (2020) presentan una revisión del estado del arte del tema de análisis y predicción del rendimiento académico de estudiantes. Los investigadores usan técnicas de *Machine Learning*. Los autores mencionan que predecir el desempeño de los estudiantes es uno de los temas más importantes para contextos de aprendizaje como escuelas y universidades. Estas técnicas ayudan a diseñar mecanismos eficaces que mejoren los resultados académicos y eviten la deserción. La revisión del estado del arte se realizó con aproximadamente 70 trabajos. En ella se muestran diferentes técnicas modernas ampliamente aplicadas para predecir el desempeño de los estudiantes, junto con los objetivos que deben alcanzar en este campo. Estas técnicas y métodos, que pertenecen al área de inteligencia artificial, son principalmente *Machine Learning*, filtrado colaborativo, sistemas de recomendación y redes neuronales artificiales, entre otros.



Crivei *et al.* (2020) investigaron la efectividad de aplicar dos modelos de aprendizaje no supervisados, el análisis de componentes principales (PCA) y la minería de reglas de asociación relacional (RAR) como herramientas inteligentes para el análisis de conjuntos de datos académicos. El estudio realizado en este trabajo destacó el potencial de los modelos de aprendizaje no supervisados para detectar patrones en el análisis del rendimiento académico de los estudiantes. El desempeño de la clasificación no supervisada está influenciado por posibles anomalías del conjunto de datos académicos identificadas a través de las variables evaluación del seminario, nota del proyecto, puntaje del estado del proyecto, nota de prueba escrita y nota del examen final. Se obtiene como resultado que el modelo PCA presentó un rendimiento de 0,722 al someter todas las variables. Sin embargo, al retirar el atributo 3 (puntaje del estado del proyecto), se obtuvo una mejor precisión, de 0,786. Por otro lado, el modelo RAR presentó una precisión general del 0,714 y un error general de 0,422. Sin embargo, al retirar el atributo 3, el modelo RAR proporciona una mayor precisión (0,723) y un error menor (0,421). Una posible causa de las instancias anómalas de la visualización en PCA y RAR, podría ser que el atributo 3 presentara una menor correlación con respecto a la calificación del examen final. Así mismo, se evidencia que el desempeño de clasificación no supervisada está influenciado por posibles anomalías del conjunto de datos académicos y por el pequeño número de atributos utilizados para caracterizar las instancias.

Moubayed *et al.* (2020) utilizan algoritmos de aprendizaje no supervisados para agrupar a los estudiantes en grupos con base en su actividad e interacción en línea. Las métricas consideradas incluyen tanto unas relacionadas con la interacción —las describen como la frecuencia con la que el estudiante interactuó con el material del curso en la plataforma de aprendizaje—, como unas relacionadas con el esfuerzo para tener una representación más completa del compromiso de los estudiantes para completar las tareas del curso. Además, se realizan pruebas de significación estadística para comprobar si los grupos resultantes son estadísticamente diferentes o no. Este estudio se centra en 486 estudiantes que asistieron a un curso de segundo año de pregrado en la Universidad de Ontario Occidental en Canadá. Sin embargo, los datos no incluyen la demografía de los estudiantes, pero

sí variables como fecha del evento, tipo de evento, lugar del evento, inicio del evento, fin del evento e ID del estudiante. Se estudiaron tres modelos de agrupación diferentes procesados a través del algoritmo de aprendizaje no supervisado *K-Means*: el modelo de dos niveles, el modelo de tres niveles y el modelo de cinco niveles. Los resultados experimentales mostraron que el número de inicios de sesión, así como la duración media para enviar las tareas, pueden ser indicativos del nivel de compromiso de los estudiantes, mientras que el número de contenido lee/accesos no lo es.

Kumar *et al.* (2020) combinan y aplican técnicas de *Machine Learning* como aprendizaje no supervisado (*K-Means Clustering* y *Hierarchical Clustering*) y aprendizaje supervisado (*Decision Trees* y Naive Bayes) a un conjunto de datos de estudiantes de K L University para evaluar el rendimiento del estudiante y predecir si un estudiante aprobará o fallará en un examen técnico que se llevó a cabo como proceso de reclutamiento. Hacen uso de variables introductorias como calificaciones intermedias y el porcentaje de avance en B.Tech de estudiantes de segundo y tercer semestre. Tres clústeres se forman con *K-Means Clustering* y *Hierarchical Clustering*: bajo (notas intermedias menores a 85), bueno (notas intermedias de 85 a 90) y excelente (de 90 en adelante). Al comparar *K-Means Clustering* y *Hierarchical Clustering*, el rendimiento de *K-Means* es alto, mientras que *Hierarchical* es moderado. *K-Means* es bueno cuando se trata de grandes conjuntos de datos, y *Hierarchical* es bueno con pequeños conjuntos de datos. Al comparar las técnicas de aprendizaje supervisado se obtienen predicciones del 72%. Al comparar ambos algoritmos, la predicción con más precisión es la del algoritmo de Bayes para este conjunto de datos. En el futuro, los resultados se pueden ampliar con más número de temas y un conjunto de datos más grande.

Almasri *et al.* (2020) hacen uso de técnicas de minería de datos educativos (EDM) supervisadas y no supervisadas para construir un modelo comprensible y eficaz para predecir el rendimiento académico de los estudiantes. Proponen un marco unificado para construir un nuevo modelo de clasificación basado en clústeres supervisados (CB). El marco unificado utiliza la técnica para agrupar los registros históricos de los estudiantes en un conjunto de grupos homogéneos. Luego, se construye el modelo clasificador para cada clúster y los clasificadores unificados finales junto con los centroides. En cada clúster se

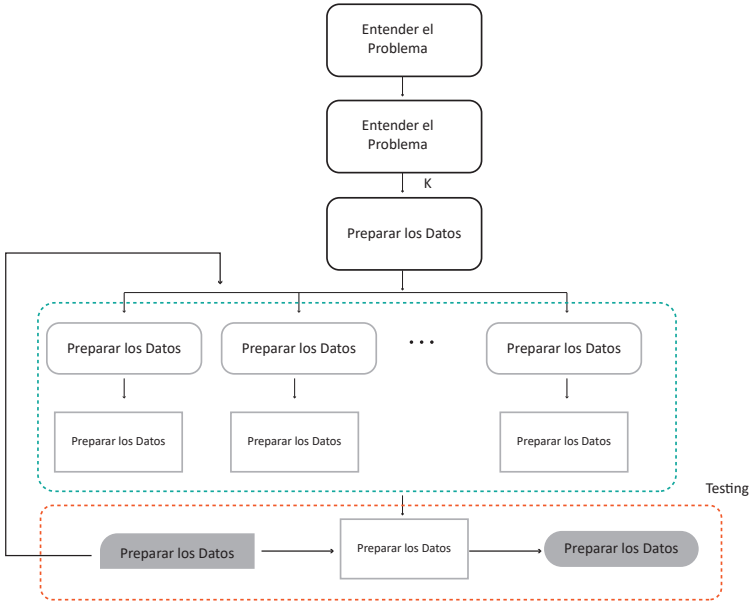
utiliza, como modelo clasificador, CB. Sin embargo, previamente se utilizan técnicas no supervisadas como *canopy clustering* phase, el cual se utiliza en los enfoques de BC propuestos, ya que utilizan dicha fase como un proceso previo para determinar el número de clústeres para el algoritmo *K-Means*.

También se usó *K-Means clustering* algorithm phase, dado que el enfoque propuesto utiliza el valor K de salida de la fase de clúster del canopy que representa un parámetro del algoritmo de clúster de *K-Means*. Para aplicar el modelo propuesto se utilizan los registros históricos de los estudiantes del departamento del MIS de la Universidad Aplicada Balqa (BAU) en Jordania, en 13 años académicos, del 2006 al 2018. Este conjunto de datos contiene registros de 1062 graduados y se usan las variables calificaciones de los estudiantes de 22 cursos MIS durante los años académicos e información demográfica como género, edad, dirección y año de graduación. Los resultados experimentales muestran que el modelo CB gana un rendimiento de alta precisión, el 96,25 %. Además, los autores utilizaron técnicas de selección de características para seleccionar las características relevantes de un espacio de características. Finalmente, el modelo obtiene un rendimiento de alta precisión y utiliza características relevantes alcanzadas al 96,96 %. El porcentaje de características relevantes en promedio es el 57,4 % de las características generales (figura 30).

Iatrellis *et al.* (2020) intentan predecir los resultados de los estudiantes de programas de estudios de educación superior. El enfoque se ha aplicado a un plan de estudios de informática de grado ofrecido por la Universidad de Tesalia en Grecia. Usando el algoritmo *K-Means*, se reveló la presencia de tres grupos coherentes de estudiantes que ayudaron a entrenar al algoritmo para realizar un análisis supervisado más profundo de cada uno de los clústeres. Los dos enfoques del artículo buscan predecir el tiempo de finalización de la titulación y la inscripción de los estudiantes en los programas educativos ofrecidos con el fin de proporcionar información para la mejora educativa, la reducción de costes o la identificación de los estudiantes que están en riesgo académicamente. Las variables utilizadas fueron promedio de calificaciones, campo de especialización, calificación del proyecto final, tiempo para completar el ciclo básico, desempeño en el primer año en materias básicas, escuela secundaria, rango en la clase, programa de movilidad estudiantil,

pasantía, premios, calificaciones en la escuela secundaria y calificaciones en los exámenes de ingreso. Estas variables permitieron identificar clústeres que fueron utilizados en la fase dos para entrenar los modelos de predicción para los estudiantes que comparten un conjunto común de características o lograron resultados similares (como la personalidad del estudiante, el tipo de escuela secundaria, entre otros). Estos tres clústeres generados obtuvieron una precisión de 85,20 %, 71,31 % y 81,25 %.

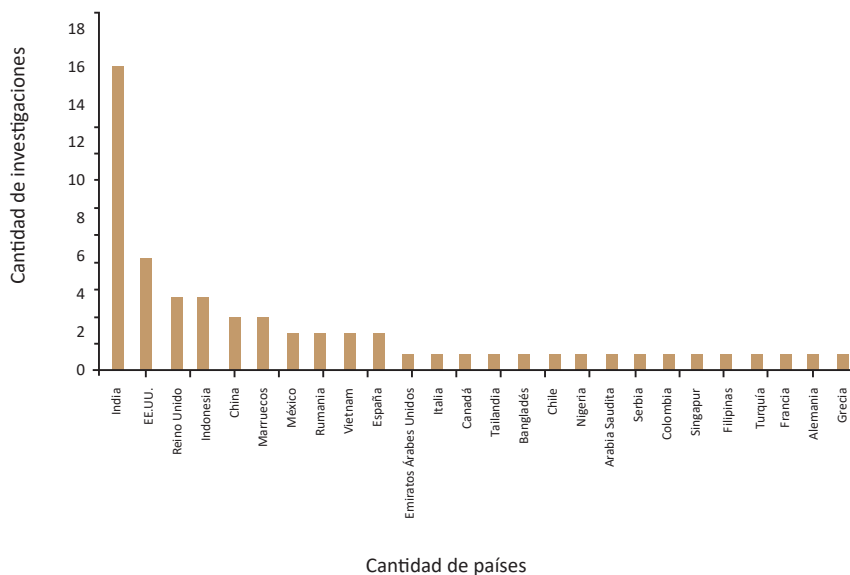
Figura 30. La arquitectura de clasificación CB



Fuente: Almasri *et al.* (2020).

## 5.4. Análisis de las producciones académicas relacionadas con aprendizaje no supervisado

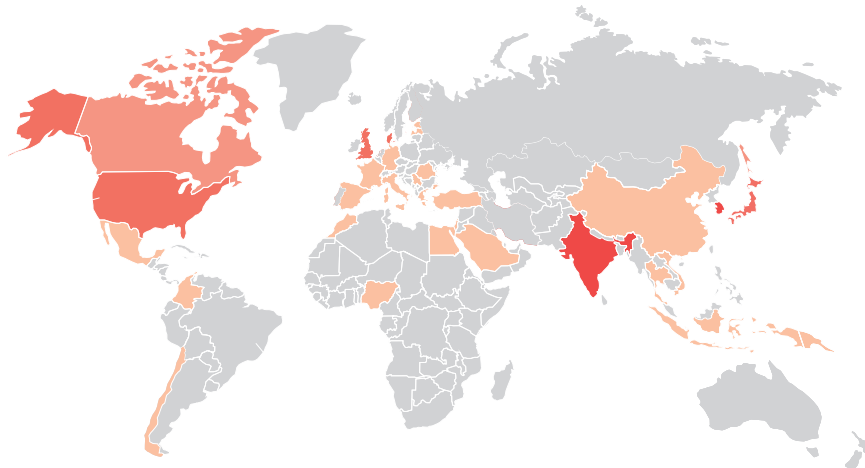
Un aspecto que vale la pena tener en cuenta luego de la breve descripción y el análisis de cada uno de los artículos investigados durante el periodo entre 2010 y 2020 es la nacionalidad de las investigaciones que estudian esta parte de la temática por medio de sus publicaciones en revistas científicas de alto impacto. A continuación, en la figura 31 se presenta el diagrama de barras de las investigaciones del sector educación y su país de publicación.

**Figura 31.** Investigación referencial y su representación en el mundo**Fuente:** elaboración propia.

En la figura 32 es evidente que, durante la última década, las investigaciones de aprendizaje no supervisado guardan fuerte relación con el continente asiático, India, Indonesia y China son los países de mayor representación pues su cultura y su estructura gubernamental se enfocan en fortalecer todos los niveles educativos de los estudiantes. Además, técnicas de este tipo permiten evidenciar variables y conceptos que no se encuentran normalmente en métodos tradicionales, lo cual les permite identificar fallas en el sistema educativo para mejorar el rendimiento y alcanzar la excelencia académica. Finalmente, en el continente americano se destaca Estados Unidos. Se trata de una potencia mundial, y el gobierno invierte recursos en la formación, la permanencia y la finalización exitosa de sus estudiantes en los diferentes niveles de educación.

Leonardo Emiro Contreras Bravo, Giovanni Mauricio Tarazona Bermúdez, Angie Paola Alemán Cardona, Jhonatan Estiven Ruiz Vanegas, Brandon Herney Hernández Arcila, Daniela Garzón Trujillo, Óscar Eduardo Sánchez Gómez, Marisol Ángel Cifuentes

**Figura 32.** Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado



**Fuente:** elaboración propia.

# Capítulo 6

## Aprendizaje no supervisado: algoritmos implementados

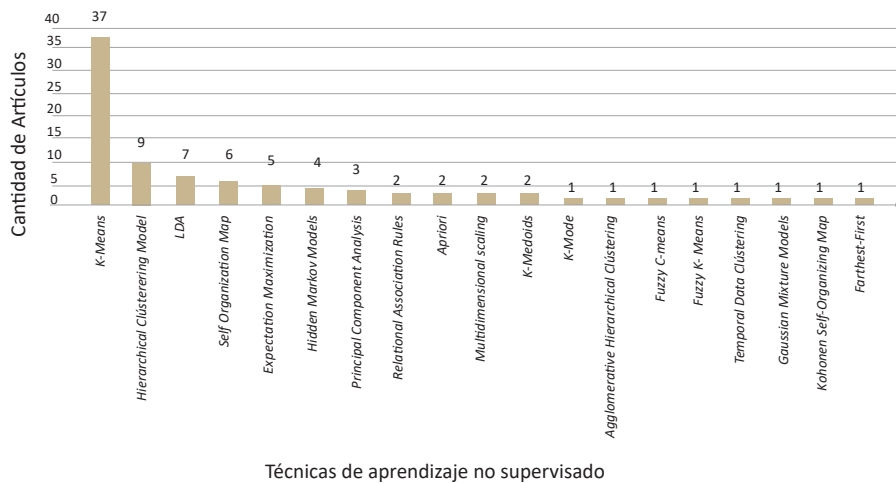
Existen múltiples algoritmos para el aprendizaje no supervisado, así como múltiples autores que combinan modelos o técnicas de aprendizaje supervisado, no supervisado y semisupervisado. En estos casos, las técnicas se ejecutan de forma paralela para medir el rendimiento de cada una y poder analizar su óptima utilización en trabajos posteriores o similares. Los algoritmos usados son *K-Means*, *K-Medoids*, *Fuzzy C-Means*, *C-Means*, *Hierarchical clustering*, *Latent Dirichlet Allocation*, *Principal Component Analysis*, Gaussian mixture, *Hidden Markov Model*, *Apriori*, etc. A continuación, en la figura 33 se muestran cada una de las técnicas de aprendizaje no supervisado, así como el uso de estas por cada una de las investigaciones consultadas en la última década.

### 6.1. *K-Means*

*K-Means* es un algoritmo no supervisado de clustering que se utiliza cuando se tiene una gran cantidad de datos sin etiquetar. Se basa en particiones sistemáticas de datos. El objetivo de este algoritmo es el de encontrar “K” grupos (clústeres) entre los datos sin ningún condicional. Es una de las técnicas de agrupamiento no jerárquico más ampliamente utilizadas. Se usa para minimizar la suma de la distancia euclidiana cuadrada entre patrones y centros de clúster a través de un algoritmo iterativo de escalada en colina (Jin y Han, 2017) . El algoritmo *K-Means* requiere tres parámetros específicos: número de clústeres K, inicialización de clúster, y métrica de distancia:

- Número de clústeres K: El algoritmo *K-Means* generalmente se ejecuta de forma independiente para diferentes valores de K. El valor de K se selecciona entonces utilizando la partición que parece la más significativa para el experto en dominios.
- Inicialización del clúster: Para superar los mínimos locales del clúster *K-Means*, el algoritmo *K-Means* selecciona los centros de clúster iniciales K aleatoriamente a partir de los puntos de datos.
- Métrica de distancia: Existe una variable en el algoritmo que representa la distancia entre los datos y los clústeres. K-medias se utiliza típicamente con la distancia euclidiana o Mahalanob (métrica de distancia para calcular la distancia entre los puntos y centros de clúster).

**Figura 33.** Técnicas de UML utilizadas por las investigaciones consultadas

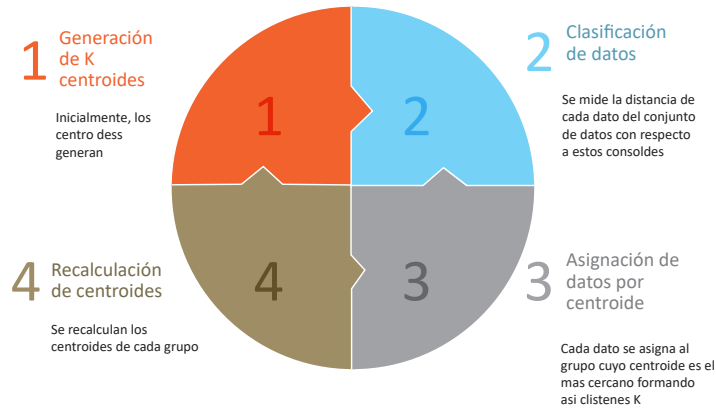


**Fuente:** elaboración propia.

El algoritmo *K-Means* considera los pasos mostrados en la figura 34. Este proceso se repite hasta que los grupos sean estables, es decir, hasta que todos los grupos de una iteración a la siguiente no cambien, o hasta que se haya alcanzado un número predefinido de iteraciones. De lo contrario, el proceso se repite desde el paso 2.



**Figura 34.** Algoritmo *K-Means*



**Fuente:** adaptado de Reyes y Villegas (2018).

Debido a su facilidad de uso, se encontró una alta cantidad de trabajos que utilizaron este algoritmo en la revisión del estado del arte, como se muestra en la tabla 3.

**Tabla 3.** Autores consultados que han usado el algoritmo *K-Means* durante la última década

| Algoritmo      | Referencias bibliográficas       |
|----------------|----------------------------------|
| <i>K-Means</i> | (Ahmad <i>et al.</i> , 2020)     |
|                | (Alonso y Marcela, 2016)         |
|                | (Balakrishnan, 2013)             |
|                | (Bapu <i>et al.</i> , 2015)      |
|                | (Battaglia <i>et al.</i> , 2016) |
|                | (Chau <i>et al.</i> , 2019)      |
|                | (Sobnath <i>et al.</i> , 2019)   |
|                | (Durairaj y Vijitha, 2014)       |
|                | (Hidayah <i>et al.</i> , 2016)   |
|                | (Iam-On y Boongoen, 2017)        |
|                | (Imtiyaz y Singhal, 2016)        |
|                | (Islam y Haque, 2012)            |
|                | (Joshika y Rajeshwari, 2019)     |

| Algoritmo | Referencias bibliográficas                |
|-----------|-------------------------------------------|
|           | (Karnad <i>et al.</i> , 2017)             |
|           | (Khan y Ghosh, 2020)                      |
|           | (Kumar <i>et al.</i> , 2020)              |
|           | (Li y Shi, 2015)                          |
|           | (Liu y D'Aquin, 2017)                     |
|           | (Martinez-Rodriguez <i>et al.</i> , 2019) |
|           | (Moubayed <i>et al.</i> , 2020)           |
|           | (Zhang <i>et al.</i> , 2017)              |
|           | (Najdi y Brahim, 2016)                    |
|           | (Najdi y Er-Raha, 2016)                   |
|           | (Nájera <i>et al.</i> , 2017)             |
|           | (Nguyen y Vo, 2019)                       |
|           | (Oyelade <i>et al.</i> , 2010)            |
|           | (Rana y Garg, 2016)                       |
|           | (Rana y Garg, 2017)                       |
|           | (Sabitha y Mehrotra, 2012)                |
|           | (Sajjadi <i>et al.</i> , 2018)            |
|           | (Santoso y Yulia, 2019)                   |
|           | (Shidhaganti y Patil, 2019)               |
|           | (Shukla y Malviya, 2019)                  |
|           | (Aher y Lobo, 2013)                       |
|           | (Tibay <i>et al.</i> , 2019)              |
|           | (Vital <i>et al.</i> , 2019)              |
|           | (Yamasari <i>et al.</i> , 2017)           |

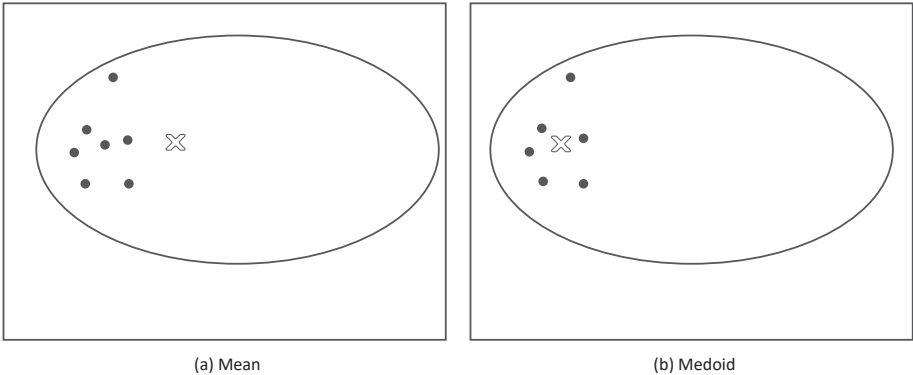
Fuente: elaboración propia.

## 6.2. *K-Medoides*

Es una variante de *K-Means* más robusta a los ruidos y a los valores atípicos. En lugar de usar el punto medio como centro de un cúmulo, *K-Medoides* usa un punto real en el cúmulo para representarlo (Jin y Han, 2017). El medoide es el objeto más céntrico del cúmulo, con una suma mínima de distancias a

otros puntos. El grupo de puntos de la derecha forman un grupo, mientras que el punto más a la derecha es un valor atípico. La media es grandemente influenciada por el valor atípico, y por tanto no puede representar el centro del cúmulo correcto, mientras que el medoide es robusto al valor atípico y representa correctamente el centro del cúmulo (figura 35).

**Figura 35.** Diferencia entre *Mean* y *Medoid*



**Fuente:** Jin y Han (2017).

La idea básica es la siguiente: seleccione puntos  $K$  representativos para formar grupos iniciales, y luego muévase repetidamente a mejores representantes de grupos. Se analizan todas las combinaciones posibles de puntos representativos y no representativos, y se calcula la calidad de la agrupación resultante para cada par. Un punto representativo original se sustituye por el nuevo punto de mayor reducción de la función de distorsión. En cada iteración, el conjunto de mejores puntos para cada clúster forma los nuevos medoides respectivos. La tabla 4 muestra los trabajos relacionados.

**Tabla 4.** Autores consultados que han usado el algoritmo *K-Medoids* durante la última década

| Algoritmo        | Referencias bibliográficas       |
|------------------|----------------------------------|
| <i>K-Medoids</i> | (Cahaya <i>et al.</i> , 2017)    |
|                  | (De La Hoz <i>et al.</i> , 2019) |

**Fuente:** elaboración propia.

### 6.3. Fuzzy C-Means

El algoritmo *Fuzzy C-Means* fue presentado por Bezdek y Dunn, por primera vez, en 1973, como una técnica que combina los métodos de una función objetivo con los de la lógica *Fuzzy*, término presentado por Lofty Zadeh en 1960 como medio para modelar la incertidumbre a través de las etapas de fuzzificación, reglas de evaluación y defuzzificación. El algoritmo *Fuzzy C-Means* realiza la formación de clúster a través de una partición suave de los datos, es decir, para realizar el reconocimiento de patrones a través del hallazgo de los grados de pertenencia de cada individuo a los diferentes clústeres. Un individuo no tendría pertenencia exclusiva a un solo grupo, sino que podría tener grados de pertenencia a distintos grupos, a diferencia de otros métodos que realizan la formación de los clústeres con base en la lógica binaria o la partición dura (Universidad Nacional Agraria La Molina, 2017). El criterio de agrupamiento utilizado para agrupar subconjuntos es una función objetiva de mínimos cuadrados generalizada. Entre las características de este programa está la selección de tres normas (euclidiana, diagonal o mahalonobis), un factor de ponderación ajustable que controla esencialmente la sensibilidad al ruido, la aceptación de números variables de clústeres y salidas con varias medidas de validez de clúster (Tilson *et al.*, 1988)

**Tabla 5.** Autores consultados que han usado el algoritmo *Fuzzy C-Means* durante la última década

| Algoritmo            | Referencias bibliográficas      |
|----------------------|---------------------------------|
| <i>Fuzzy C-Means</i> | (Yamasari <i>et al.</i> , 2017) |

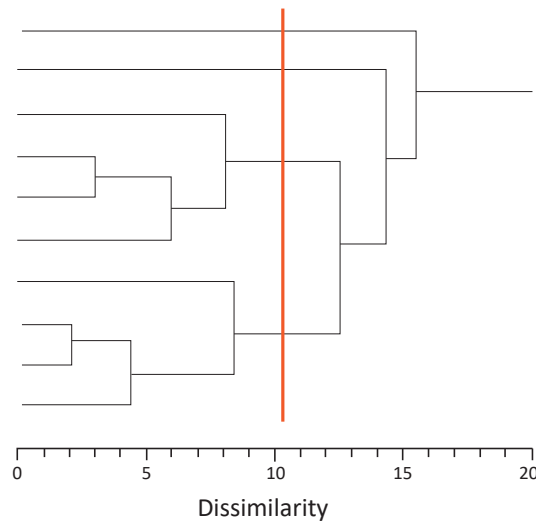
**Fuente:** elaboración propia.

### 6.4. Hierarchical Clustering (HCM/HCA)

El HCM consiste en segmentar o agrupar una muestra de objetos en clústeres, de modo que los que están dentro de cada grupo están más estrechamente relacionados unos con otros que los objetos asignados a diferentes grupos. Así, es posible que se tenga desde un clúster que agrupa todos los objetos hasta  $n$  clústeres que agrupen a  $m$  subconjuntos de datos finitos. Como se trata de un algoritmo de aprendizaje no supervisado, es indispensable

manejar una base de datos no etiquetados que se manipula para determinar grupos. El análisis jerárquico de clústeres forma clústeres iterativamente al unir o dividir sucesivamente grupos. Hay dos tipos: divisivo, que comienza con todo el conjunto de datos en un grupo grande y luego sucesivamente es dividido en grupos más pequeños hasta que cada observación es su propio grupo; y aglomerante, en el que cada observación comienza en su propio grupo, y los grupos se emparejan sucesivamente hasta que al final cada observación está en el mismo grupo grande (Schultz *et al.*, 2013). Por ejemplo, una baraja de cartas de póker en principio tiene un total de 52 cartas —es decir, 52 clústeres de un dato cada uno—; entonces, el algoritmo intenta agrupar datos similares de acuerdo con las variables no etiquetadas, por lo que en la segunda iteración habrá cuatro clústeres, y cada uno de ellos se compondría de 14 datos. En esta iteración, la agrupación se realiza a partir del palo de la carta (corazones, diamantes, picas y tréboles). Sin embargo, el algoritmo no se detiene ahí: seguirá iterando hasta no encontrar nuevas formas de agrupar, por lo que posiblemente se desglosaran clústeres con menos cantidad de datos a partir del número, el color, la letra, el símbolo, etc. Este proceso continúa hasta que hay un número  $n$  definido de clústeres, como se muestra en la figura 36.

**Figura 36.** Modelo de agrupación jerárquica



**Fuente:** Schultz *et al.* (2013).

En la agrupación no supervisada basada en medidas de similitud, si el número de objetos es grande —ya que hay un valor de similitud para cada par de objetos—, la matriz resultante de medidas de similitud puede ser tan enorme que el patrón o estructura subyacente no sea evidente. Si las semejanzas se extienden entre sí en función de sus “valores” o “fuerza”, tenemos una serie de medidas de similitud, y por tanto construimos un sistema jerárquico de representaciones de agrupamiento, que van desde el agrupamiento first (fila superior) hasta el último agrupamiento (línea inferior). El agrupamiento first es el agrupamiento “débil” (es decir que cada uno de los objetos  $n$  se representa como un clúster separado), y el último agrupamiento es el agrupamiento “fuerte” (es decir, todos los objetos  $n$  se agrupan como un solo clúster) (Zhang, 2017). Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 6.

**Tabla 6.** Autores consultados que han usado el algoritmo *Hierarchical Clustering* durante la última década

| Algoritmo                      | Referencias bibliográficas       |
|--------------------------------|----------------------------------|
| <i>Hierarchical Clustering</i> | (Bapu <i>et al.</i> , 2015)      |
|                                | (Battaglia <i>et al.</i> , 2016) |
|                                | (Cheng y Shwe, 2019)             |
|                                | (Miller, 2018)                   |
|                                | (Rana y Garg, 2016)              |
|                                | (Rana y Garg, 2017)              |
|                                | (Shukla y Malviya, 2019)         |

Fuente: elaboración propia.

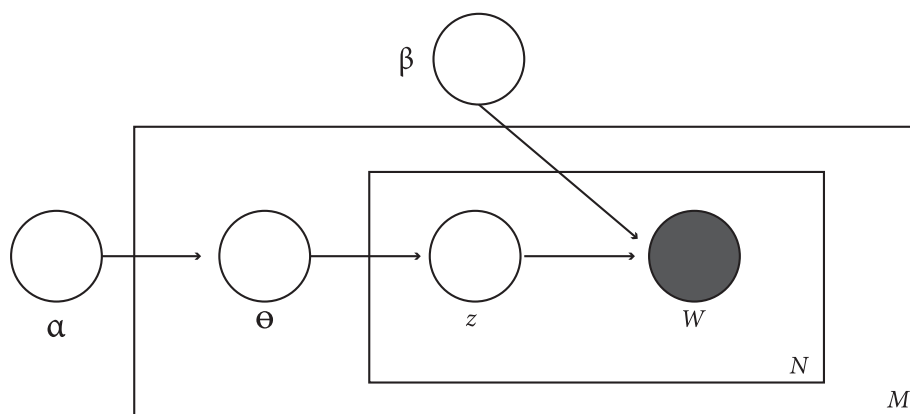
### 6.5. Latent Dirichlet Allocation (LDA)

LDA es un modelo probabilístico generativo. La idea básica es que los documentos se representan como mezclas aleatorias sobre temas latentes, y cada tema se caracteriza por una distribución sobre palabras. El modelo LDA no está necesariamente vinculado con el texto, y tiene aplicaciones a otros problemas relacionados con las colecciones de datos, incluidos los datos de dominios como el filtrado colaborativo, la recuperación de imágenes basadas en contenido y la bioinformática. Se trata de un modelo generativo que

permite que conjuntos de observaciones puedan ser explicados por grupos no observados que explican por qué algunas partes de los datos son similares (Johnson y Robinson, 2003).

LDA implica tres niveles en los que un Dirichlet es muestreado una vez para un corpus, y notablemente el nodo de tema es muestreado repetidamente dentro del documento. Bajo este modelo, los documentos pueden asociarse a múltiples temas. Esta forma de agrupar la información permite que el modelo encuentre asociaciones inversas entre variables, debido a que se genera una recolección independiente del orden, por lo que para el clúster generado es intrascendente que A se relacione con B o que B se relacione con A: ambos entrarán a formar parte de la misma selección, y así se pueden obtener resultados matemáticos de caracterización. En la figura 37 se muestra un ejemplo de asociación en una iteración intermedia: un clúster ya establecido intenta asociar nodos dentro y fuera de sí mismo, de acuerdo con la relación entre la información.

**Figura 37.** Representación gráfica del modelo de LDA. Las cajas son “placas” que representan réplicas. La placa exterior representa los documentos, mientras que la placa interior representa la selección repetida de temas y palabras dentro de un documento



**Fuente:** Johnson y Robinson (2003).

A continuación, se muestra a los autores que utilizaron la técnica de agrupación *Latent Dirichlet Allocation* (tabla 7).

**Tabla 7.** Autores consultados que han usado el algoritmo *Latent Dirichlet Allocation* durante la última década

| Algoritmo                          | Referencias bibliográficas       |
|------------------------------------|----------------------------------|
| <i>Latent Dirichlet Allocation</i> | (Chau <i>et al.</i> , 2019)      |
|                                    | (De La Hoz <i>et al.</i> , 2019) |
|                                    | (Guleria y Sood, 2018)           |
|                                    | (Nguyen y Vo, 2019)              |
|                                    | (Shetty <i>et al.</i> , 2019)    |
|                                    | (Srinivas y Rajendran, 2019)     |

**Fuente:** elaboración propia.

## 6.6. Análisis de componentes principales (PCA)

Según Zhang (2017), el análisis de componentes principales (PCA) es una potente técnica de procesamiento de datos y reducción de dimensiones para extraer la estructura de un conjunto de datos posiblemente de alta dimensión. El PCA tiene varias variantes o generalizaciones importantes: análisis robusto de componentes principales, análisis de componentes principales del núcleo, análisis de componentes principales dispersos, etc. PCA es ampliamente aplicado en ingeniería, biología y ciencias sociales, así como en reconocimiento facial, clasificación de un código postal, análisis de datos de expresión génica, etc.

Su propósito principal es ser un medio para reducir la dimensionalidad de un conjunto de datos multivariado y, también, iluminar su interpretación mediante la identificación de un número menor de variables que, en cierto sentido, resumen el conjunto más grande. El punto de partida del PCA es la matriz de coeficientes de correlación derivados del conjunto de datos original. La lógica detrás del método requiere que las correlaciones se obtengan a partir de variables medidas a una escala continua. En la práctica, estas variables pueden ser precisamente continuas, como con magnitudes como la longitud o el peso, o pueden ser discretas como cuando la variable es algo que hemos contado. El origen del análisis de componentes principales (PCA) está en un artículo de Hotelling (1933) publicado en el *Journal of Educational Psychology* (Syms, 2018). Los autores que utilizaron este algoritmo se encuentran referenciados en la tabla 8.



**Tabla 8.** Autores consultados que han usado el algoritmo *Principal Component Analysis* durante la última década

| Algoritmo                           | Referencias bibliográficas    |
|-------------------------------------|-------------------------------|
| <i>Principal Component Analysis</i> | (Ahmad <i>et al.</i> , 2019)  |
|                                     | (Crivei <i>et al.</i> , 2020) |
|                                     | (Shetty <i>et al.</i> , 2019) |

Fuente: elaboración propia.

## 6.7. Gaussian Mixture Model (GMM)

Según Based (2009), un modelo de mezcla gaussiano (GMM) es una función de densidad de probabilidad paramétrica representada como una suma ponderada de densidades de componentes gaussianos. Los GMM se utilizan comúnmente como modelos paramétricos de la distribución de probabilidad de mediciones continuas. Los parámetros de GMM se estiman a partir de datos de entrenamiento con el algoritmo iterativo de maximización de expectativas (EM) o la estimación a posteriori máxima (MAP) y un modelo previo bien entrenado.

Adicionalmente, los parámetros pueden ser compartidos, o empatados. Por ejemplo, es posible tener una matriz de covarianza común para todos los componentes. La elección de la configuración del modelo (número de componentes, matrices de covarianza plena o diagonal, y la vinculación de parámetros) se determina a menudo por la cantidad de datos disponibles para estimar los parámetros de GMM y cómo se utiliza el GMM en una aplicación biométrica particular. También es importante señalar que, dado que el componente gaussiano está actuando junto para modelar la densidad general de la característica, las matrices de covarianza completa no son necesarias incluso si las características no son estadísticamente independientes (tabla 9).

**Tabla 9.** Autores consultados que han usado el algoritmo *Gaussian Mixture Models* durante la última década

| Algoritmo                      | Referencias bibliográficas |
|--------------------------------|----------------------------|
| <i>Gaussian Mixture Models</i> | (Bey <i>et al.</i> , 2019) |

**Fuente:** elaboración propia.

## 6.8. *Hidden Markov Model* (HMM)

Un modelo oculto de Márkov (HMM) es un modelo estadístico en el que se asume que el sistema a modelar es un proceso de Márkov de parámetros desconocidos. El objetivo es determinar los parámetros desconocidos (u ocultos) de dicha cadena a partir de los parámetros observables. Los parámetros extraídos se pueden emplear para llevar a cabo sucesivos análisis como, por ejemplo, en aplicaciones de reconocimiento de patrones.

Fue desarrollado a partir de la teoría de la estadística pertinente a principios de la década de los setenta y, hoy en día, se consideran como una forma específica de redes bayesianas dinámicas que se basan en la teoría de Bayes. Los MMH son modelos estadísticos para capturar información oculta de símbolos secuenciales observables (por ejemplo, una secuencia nucleotídica). En un HMM, se asume que el sistema que se está modelando es un proceso de Markov con parámetros desconocidos, y el desafío es determinar los parámetros ocultos de los parámetros observables. Un buen HMM modela con precisión la fuente del mundo real de los datos reales observados y tiene la capacidad de simular la fuente. (Franzese y Iuliano, 2018). Según Lan *et al.* (2017), HMM puede ajustar de forma automática y flexible las tendencias, elementos estacionales, covariantes y de distribución. Los trabajos que utilizaron este algoritmo en sus estudios se muestran en la tabla 10.

**Tabla 10.** Autores consultados que han usado el algoritmo *Hidden Markov Models* durante la última década

| Algoritmo                   | Referencias bibliográficas   |
|-----------------------------|------------------------------|
| <i>Hidden Markov Models</i> | (Balakrishnan, 2013)         |
|                             | (Leong <i>et al.</i> , 2016) |
|                             | (Shih <i>et al.</i> , 2010)  |
|                             | (Vital <i>et al.</i> , 2019) |

**Fuente:** elaboración propia.

## 6.9. Apriori

Es un algoritmo utilizado en minería de datos que permite encontrar de forma eficiente “conjuntos de ítems frecuentes” que sirven de base para generar reglas de asociación a partir de la identificación de los ítems individuales más frecuentes en la base y extenderlos a conjuntos de mayor tamaño siempre y cuando esos conjuntos de datos aparezcan suficientemente seguidos en dicha base de datos. *Apriori* utiliza un enfoque “bottom up”: los subconjuntos frecuentes son ampliados por un ítem a la vez (este paso se conoce como generación de candidatos) y grupos de candidatos son validados contra los datos. El algoritmo termina cuando no se pueden encontrar más ampliaciones exitosas de los conjuntos previos. *Apriori* utiliza una búsqueda en anchura y un árbol de Merkle para almacenar el conjunto de ítems candidatos eficientemente (Jin *et al.*, 2011).

**Tabla 11.** Autores consultados que han usado el algoritmo *Apriori* durante la última década

| Algoritmo      | Referencias bibliográficas        |
|----------------|-----------------------------------|
| <i>Apriori</i> | (Aher y Lobo, 2013)               |
|                | (Trandafili <i>et al.</i> , 2012) |

Fuente: elaboración propia.

## 6.10. Escalamiento multidimensional (MDS)

Según Jolliffe *et al.* (2010), MDS representa un conjunto de objetos a través de puntos en un espacio multidimensional agrupados de tal manera que siempre se encuentran juntos aquellos con características similares. MDS busca una representación dimensional inferior de una matriz de disimilitud mientras preserva las distancias de pares tanto como sea posible. Cada nueva dimensión se encuentra de tal manera que minimiza la función de estrés (Jolliffe *et al.*, 2010)

$$C(Z_1, Z_2, \dots, Z_n) = \left[ \sum (d_{ij} - \|Z_i - Z_j\|)^2 \right]^{1/2}$$

En este caso,  $d_{ij}$  son las diferencias de concurrencia del conjunto encontradas por  $d_{ij} = 1 - C_{ij}$ .

Los diagramas o puntos proyectados por el algoritmo están dispuestos en un espacio de modo que las distancias entre pares de puntos tienen la relación más fuerte posible con el grado de similitud entre los pares de objetos. Es decir, dos puntos que están juntos representan dos objetos similares, y un par de puntos que están muy separados representan dos objetos diferentes. El espacio es generalmente un espacio euclidiano de dos o tres dimensiones, pero puede ser no euclidiano y puede tener más dimensiones (Gad, 2002).

**Tabla 12.** Autores consultados que han usado el algoritmo *Multidimensional Scaling* durante la última década

| Algoritmo                | Referencias bibliográficas       |
|--------------------------|----------------------------------|
| Multidimensional Scaling | (De La Hoz <i>et al.</i> , 2019) |
|                          | (Miller, 2018)                   |

Fuente: elaboración propia.

### 6.11. Expectation Maximization (EM)

Es un método de optimización iterativo para estimar la probabilidad máxima cuando hay datos faltantes u ocultos. Permite conocer el conjunto de parámetros del modelo que se ajusta a los datos con base en algunos parámetros desconocidos “ $\Theta$ ”, dados los datos de medición “ $U$ ” (Xu, 2010). Sin embargo, en ocasiones no se proporcionan algunas variables molestas “ocultas”  $J$ , que deben integrarse. En particular, se espera maximizar la probabilidad posterior de los parámetros  $\Theta$  dados los datos  $U$ , y EM calcula una distribución sobre el espacio  $J$  (Xu, 2010):

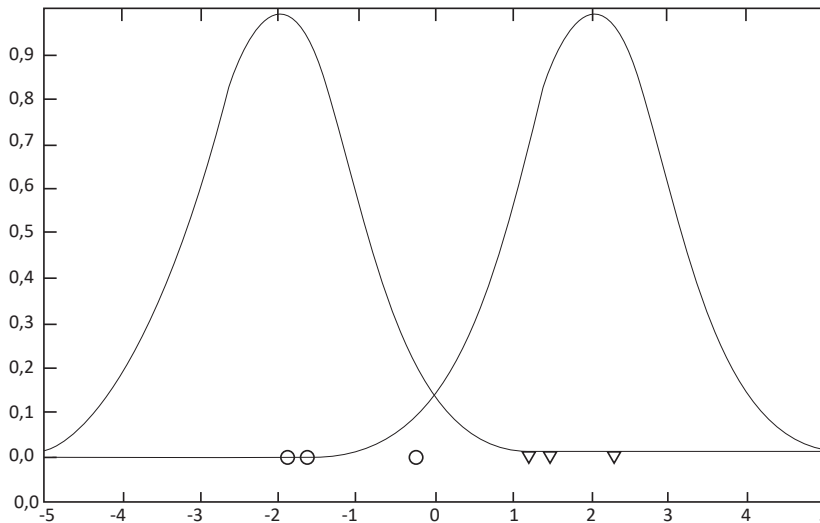
$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} \sum_{J \in J^n} P(\Theta, J | U)$$

El algoritmo EM consta de dos pasos que se repiten hasta que se alcanza un criterio de convergencia; el primer paso se llama *paso de expectativa* (paso E) consiste en una estimación de datos faltantes dados los datos observados y los parámetros estimados actuales; el segundo paso, llamado *paso de*

*maximización* (paso M) consiste en maximizar la función de verosimilitud al asumir el conocimiento de los datos faltantes como los datos estimados encontrados en el paso anterior. La convergencia se lleva a cabo cuando se alcanza la diferencia máxima entre los valores estimados de los parámetros del paso E y el paso M.

Una de las explicaciones más perspicaces de la ME —una que proporciona una comprensión más profunda de su funcionamiento que la intuición de alternar entre variables— es en términos de maximización del límite inferior. En esta derivación, el paso E se puede interpretar como la construcción de un límite inferior local a la distribución posterior, mientras que el paso M optimiza el límite para así mejorar así la estimación de las incógnitas. Esto se demuestra a continuación como un ejemplo simple, en la figura 38.

**Figura 38.** Ejemplo EM: componentes y datos de la mezcla. Los datos constan de tres muestras extraídas de cada componente de la mezcla, que se muestran arriba como círculos y triángulos. Las medias de los componentes de la mezcla son -2 y 2, respectivamente



Fuente: Xu (2010).

A continuación, se detallan las investigaciones consultadas que usaron el algoritmo EM:

**Tabla 13.** Autores consultados que han usado el algoritmo EM durante la última década

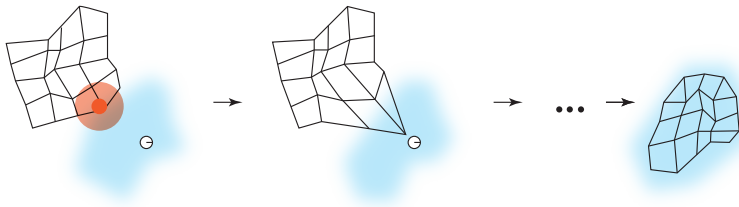
| Algoritmo                       | Referencias bibliográficas        |
|---------------------------------|-----------------------------------|
| <i>Expectation maximization</i> | (Bapu <i>et al.</i> , 2015)       |
|                                 | (Hidayah <i>et al.</i> , 2016)    |
|                                 | (Li y Shi, 2015)                  |
|                                 | (Shukla y Malviya, 2019)          |
|                                 | (Trandafili <i>et al.</i> , 2012) |

Fuente: elaboración propia.

## 6.12. Self-Organization Maps (SOM)

Mapa autoorganizado (SOM) es un algoritmo de reducción de dimensionalidad basado en redes neuronales para el aprendizaje no supervisado que generalmente se utiliza para representar un conjunto de datos de alta dimensión como un patrón discretizado bidimensional. La reducción de la dimensionalidad se realiza mientras se conserva la topología de los datos presentes en el espacio de características original. En SOM, los vectores que están cerca en el espacio de alta dimensión también terminan asignándose a los nodos de SOM que están cerca en el espacio de baja dimensión. *K-Means* puede considerarse un caso simplificado de SOM en el que los nodos (centroides) son independientes entre sí. Los prototipos construidos por SOM son promedios locales de los datos y, por tanto, menos sensible a las variaciones aleatorias que los datos originales (figura 39).

**Figura 39.** Ejemplo del entrenamiento de un mapa auto organizado. La zona azul es la distribución de los datos de entrenamiento, y el disco blanco es el ejemplo de entrenamiento actual para esa distribución



Fuente: Kulcsar *et al.* (2014).

El punto esencial en la aplicabilidad de SOM es el carácter topológico del mapeo: patrones similares se mapean en las ubicaciones cercanas en el mapa. Por tanto, es un método para la visualización de patrones multidimensionales, el número de neuronas puede variar desde unas pocas docenas hasta varios miles (Kulcsar *et al.*, 2014).

La principal ventaja de utilizar un SOM es que los datos se interpretan y comprenden fácilmente. La reducción de la dimensionalidad y la agrupación de cuadrículas facilita la observación de similitudes en los datos. Los SOM tienen en cuenta todos los datos de la entrada para generar estos grupos y pueden modificarse de manera que ciertos datos tengan un efecto mayor o menor en el lugar donde se coloca una entrada (Kohonen, 2013).

Esta organización —una especie de diagrama de similitud de los modelos— permite obtener una idea de las relaciones topográficas de los datos, especialmente de los elementos de datos de alta dimensión. Si los elementos de datos pertenecen a ciertas clases predeterminadas, los modelos (y los nodos) se pueden calibrar de acuerdo con estas clases. Luego, un elemento de entrada desconocido se clasifica de acuerdo con ese nodo, cuyo modelo es más similar con él en alguna métrica utilizada en la construcción del SOM.

Finalmente, los SOM son capaces de manejar varios tipos de problemas de clasificación, a la vez que proporcionan un resumen útil, interactivo e inteligible de los datos. Por ejemplo, un SOM podría utilizarse en lugar de un filtro de spam bayesiano.

Según InteractiveChaos (2013), la reducción de dimensionalidad es otra de las grandes aplicaciones de los algoritmos no supervisados. El objetivo de este tipo de algoritmo es convertir un *dataset* de una cierta dimensión en otro con una menor dimensión. Los motivos para querer realizar esta reducción son variados:

- Puede reducir los recursos necesarios para pasar los datos por un algoritmo supervisado.
- Puede eliminar ruido presente en el *dataset* original.
- Los resultados pueden ser más fácilmente interpretables.
- Hay tres métodos para realizar reducción de dimensionalidad:

- Selección de características, es decir, escoger un subconjunto de las características originales que, según cierto criterio, representen bien el conjunto de datos.
- Derivación de características, método consistente en la creación de nuevas características a partir de las originales. Lógicamente, esto supone una reducción de dimensionalidad solo si creamos características que combinen dos o más de las originales y podemos sustituir las originales por las nuevas sin que se produzca una pérdida de información excesiva.

A continuación, se detallan las investigaciones consultadas que usaron el algoritmo SOM:

**Tabla 14.** Autores consultados que han usado el algoritmo *Self-Organization Map* durante la última década

| Algoritmo                    | Referencias bibliográficas   |
|------------------------------|------------------------------|
| <i>Self-Organization Map</i> | (Ahmad <i>et al.</i> , 2019) |
|                              | (Ciubotariu y Crivei, 2019)  |
|                              | (Miller, 2018)               |
|                              | (Ramadass y Abhraham, 2010)  |
|                              | (Sabitha y Mehrotra, 2012)   |
|                              | (Shidhaganti y Patil, 2019)  |

**Fuente:** elaboración propia.

### 6.13. *Agglomerative Hierarchical Clustering* (AHC)

Es el método más común de agrupamiento jerárquico utilizado para agrupar objetos en grupos según su similitud. También se conoce como AGNES (anidamiento aglomerativo). El algoritmo comienza por tratar cada objeto como un clúster único. A continuación, los pares de grupos se fusionan sucesivamente hasta que todos los grupos se fusionan en un grupo grande que contiene todos los objetos. El resultado es una representación basada en árboles de los objetos, denominada dendrograma (Addinsoft, 2021). El algoritmo *agglomerative Hierarchical clustering* (AHC) tiene las siguientes ventajas:



- Funciona a partir de las diferencias entre los objetos a agrupar. Un tipo de disimilitud puede adaptarse al tema estudiado y la naturaleza de los datos.
- Uno de los resultados es el dendrograma que muestra la agrupación progresiva de los datos. Entonces, es posible hacerse una idea de un número adecuado de clases en las que se pueden agrupar los datos.

*Agglomerative Hierarchical Clustering* (AHC) es un método de clasificación iterativo cuyo principio es simple.

- El proceso comienza calculando la diferencia entre los  $N$  objetos.
- Luego, dos objetos que cuando se agrupan minimizan un criterio de aglomeración dado y se agrupan creando así una clase que comprende estos dos objetos.
- Luego, la diferencia entre esta clase y los  $N-2$  otros objetos se calculan utilizando el criterio de aglomeración. A continuación, se agrupan los dos objetos o clases de objetos cuya agrupación minimiza el criterio de aglomeración.

Este proceso continúa hasta que todos los objetos se hayan agrupado. Estas sucesivas operaciones de agrupamiento producen un árbol de agrupamiento binario (dendrograma) cuya raíz es la clase que contiene todas las observaciones. Este dendrograma representa una jerarquía de particiones. Así, es posible elegir una partición truncando el árbol en un nivel dado dependiendo el nivel de las restricciones definidas por el usuario (el usuario sabe cuántas clases se obtendrán) o de criterios más objetivos, como se muestra en la figura 40.

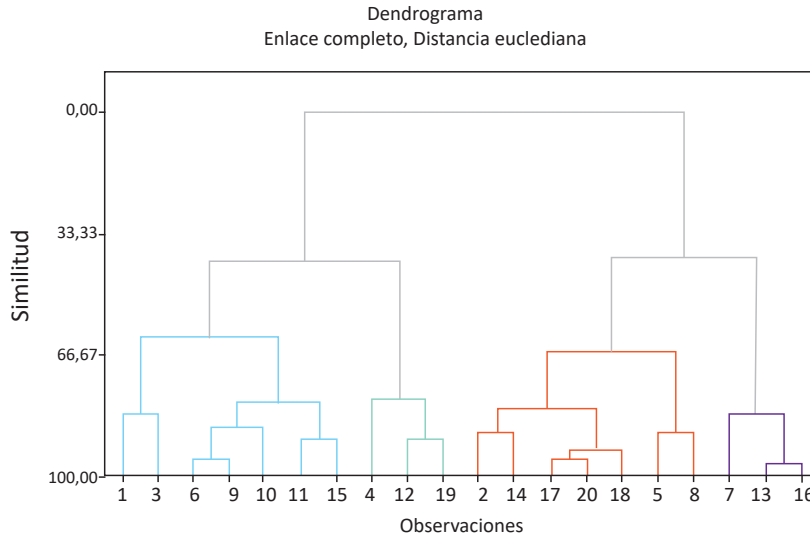
A continuación, se detallan las investigaciones consultadas que usaron el Algoritmo AHC:

**Tabla 15.** Autores consultados que han usado el algoritmo *Agglomerative Hierarchical Clustering* durante la última década

| Algoritmo                                    | Referencias bibliográficas |
|----------------------------------------------|----------------------------|
| <i>Agglomerative Hierarchical Clustering</i> | (Khan y Ghosh, 2020)       |

**Fuente:** elaboración propia.

**Figura 40.** Ejemplo de agrupación jerárquica aglomerativa (AHC) usando el *software* estadístico XLSTAT en Excel



Fuente: Addinsoft (2021).

## 6.14. Temporal Data Clustering

La agrupación de datos temporales proporciona técnicas fundamentales para descubrir la estructura intrínseca y condensar información sobre datos temporales. Este agrupamiento es una estructura inherente general de los datos temporales: se utiliza para que los datos unifiquen agrupaciones temporales a lo largo del tiempo (Nguyen y Vo, 2019). Para obtener agrupaciones de conjuntos a partir de agrupaciones temporales, los datos de conjuntos deben generarse como una entrada del proceso de agrupación de conjuntos. Todos los conglomerados temporales se juntan para formar un espacio conjunto para el proceso. Cada objeto ahora se representa como un vector de asociación con todos los clústeres temporales a lo largo del tiempo. A través de asociaciones de grupo de objetos, la similitud y la disimilitud entre los objetos en cada intervalo de tiempo se conservan para que puedan procesarse en el proceso de agrupamiento de conjuntos. En consecuencia, los grupos de conjuntos pueden tener más compacidad y separación que los únicos (Yang y Chen, 2011).

Los datos temporales son omnipresentes en el mundo real y existen muchas áreas de aplicación: desde el procesamiento de información multimedia hasta la minería de datos temporal. A diferencia de los datos estáticos, existe una gran dependencia entre los datos temporales y el tratamiento adecuado de la dependencia o correlación de los datos. Esta se vuelve fundamental en el procesamiento de datos temporales. El análisis de agrupamiento temporal es una manera eficaz de descubrir la estructura intrínseca y condensar información sobre datos temporales mediante la exploración de regularidades dinámicas subyacentes a los datos temporales en el aprendizaje no supervisado. El objetivo final es dividir un conjunto de datos temporales sin etiquetar en grupos para que las secuencias agrupadas en el mismo grupo sean coherentes.

El análisis de agrupamiento es una tarea de aprendizaje no supervisado extremadamente difícil. Es un problema inherentemente mal planteado y su solución a menudo viola algunos supuestos comunes. A continuación, se detallan las investigaciones consultadas que usaron el algoritmo *Temporal Data Clustering*.

**Tabla 16.** Autores consultados que han usado el algoritmo *Temporal Data Clustering* durante la última década

| Algoritmo                       | Referencias bibliográficas |
|---------------------------------|----------------------------|
| <i>Temporal Data Clustering</i> | (Nguyen y Vo, 2019)        |

Fuente: elaboración propia.

### 6.15. *Relational Association Rules (RAR)*

Las reglas de asociación relacional (RAR) son conceptos de análisis y minería de datos que amplían las reglas de asociación clásicas para capturar diferentes relaciones entre los atributos que caracterizan los datos. Las reglas de asociación relacional (RAR) se introdujeron en la literatura sobre minería de datos como una extensión de las reglas de asociación clásicas, con el objetivo de descubrir varios tipos de relaciones, tanto ordinales como no ordinales, entre atributos de datos (Bart y Van den, 2002).

Para obtener agrupaciones de conjuntos a partir de agrupaciones temporales, los datos de conjuntos deben generarse como una entrada del proceso de agrupación de conjuntos. Todos los conglomerados temporales se juntan para formar un espacio conjunto para el proceso. Cada objeto ahora se representa como un vector de asociación con todos los clústeres temporales a lo largo del tiempo. A través de asociaciones de grupo de objetos, la similitud y la disimilitud entre los objetos en cada intervalo de tiempo se conservan para que puedan procesarse en el agrupamiento de conjuntos. En consecuencia, los grupos de conjuntos pueden tener más compacidad y separación que los únicos (Nguyen y Vo, 2019). A continuación, se detallan las investigaciones consultadas que usaron el algoritmo relational association rules.

**Tabla 17.** Autores consultados que han usado el algoritmo *Relational Association Rules* durante la última década

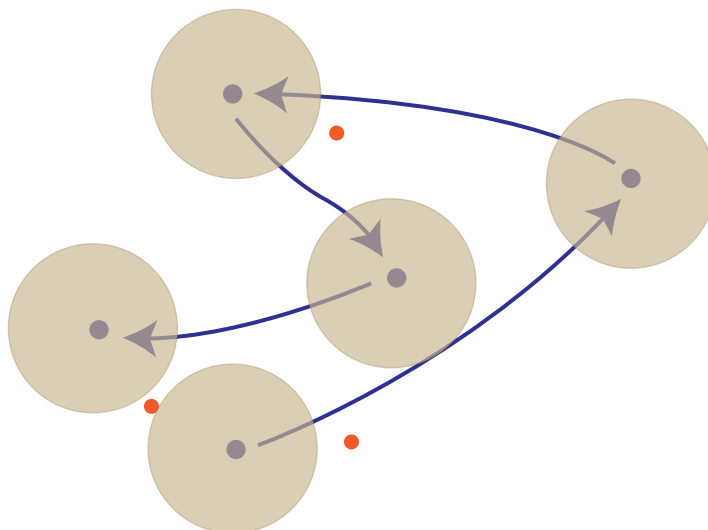
| Algoritmo                           | Referencias bibliográficas    |
|-------------------------------------|-------------------------------|
| <i>Relational Association Rules</i> | (Ciubotariu y Crivei, 2019)   |
|                                     | (Crivei <i>et al.</i> , 2020) |

**Fuente:** elaboración propia.

### 6.16. *Farthest-First*

*Farthest-First* es un algoritmo propuesto por Hochbaum y Shmoys en 1985; es rápido y codicioso. En este algoritmo, los puntos  $K$  se seleccionan primero como centros de grupos. El primer centro se selecciona al azar. El segundo centro se selecciona con avidez como el punto más alejado del primero (figura 41). Cada centro restante se determina seleccionando con avidez el punto más alejado del conjunto de centros ya elegidos, y los puntos restantes se agregan al grupo cuyo centro es el más cercano. Este algoritmo es una variante del algoritmo *K-Means*. El *Farthest-First* comienza con la selección aleatoria de una instancia que se convierte en el centroide del clúster, calcula la distancia entre cada una de las instancias y el centro. La distancia que se encuentre más alejada del centro más cercano es seleccionada como el nuevo centro del clúster. Este proceso es cíclico hasta alcanzar el número de clústeres buscado (Ponce y Margain, 2016).

**Figura 41.** Ejemplo *Farthest-First*. Los primeros cinco pasos del primer recorrido más lejano de un conjunto de puntos planos



Fuente: Ocando *et al.* (2017).

En la tabla 18 se detallan las investigaciones que usaron el algoritmo *Farthest-First* durante la última década.

**Tabla 18.** Autores consultados que han usado el algoritmo *Farthest-First* durante la última década

| Algoritmo             | Referencias bibliográficas |
|-----------------------|----------------------------|
| <i>Farthest-First</i> | (Shukla y Malviya, 2019)   |

Fuente: elaboración propia.

## 6.17. Fuzzy K-Means

En los casos en los que se usan estos algoritmos de agrupamiento, cada elemento tiene un grado de pertenencia difuso a los grupos. Este tipo de algoritmos surge de la necesidad de resolver una deficiencia del agrupamiento exclusivo, que considera que cada elemento se puede agrupar inequívocamente con los elementos de su clúster y que, por tanto, no se asemeja al resto de los elementos (Universidad de Princeton, 2015).

Cada objeto en el *Fuzzy Clustering* tiene algún grado de pertenencia con el grupo. Por tanto, los objetos que están presentes en el borde del grupo son diferentes de los objetos que están presentes en el centroide; es decir, los objetos en el borde tienen un grado menor que los objetos en el centro. Cualquier objeto  $p$  ha asignado un conjunto de coeficientes que están presentes en el  $k$ -1th cluster  $w_{k-1}(p)$  (Khalid *et al.*, 2014). En las  $C$ -medias difusas, el centroide de un grupo es la media de todos los objetos presentes en el grupo medido por el grado de pertenencia de los puntos al grupo:

$$C_{k-1} = \frac{\sum_p W_{k-1}(p)^m p}{\sum_p W_{k-1}(p)^m}$$

El procedimiento *Fuzzy-K-Means* permite que cada vector de características  $x$  tenga un grado de pertenencia al clúster  $K$ . A continuación, se detallan las investigaciones consultadas que usaron el Algoritmo *Fuzzy K-Means* durante la última década.

**Tabla 19.** Autores consultados que han usado el algoritmo *Fuzzy K-Means* durante la última década

| Algoritmo            | Referencias bibliográficas      |
|----------------------|---------------------------------|
| <i>Fuzzy K-Means</i> | (Yamasari <i>et al.</i> , 2017) |

**Fuente:** elaboración propia.

## 6.18. *K-Modes*

Según Sangam y Om (2015), el algoritmo *K-Modes* —bien conocido por su simplicidad— es un popular algoritmo de partición para agrupar datos categóricos. En este trabajo, discutimos las limitaciones de la función de distancia utilizada en este algoritmo con un ejemplo ilustrativo y luego proponemos un coeficiente de similitud basado en la entropía de información. Analizamos la complejidad temporal del algoritmo *K-Modes* con el coeficiente de similitud propuesto. La principal ventaja de este coeficiente es que mejora la precisión de agrupamiento y mantiene la escalabilidad del algoritmo *K-Modes*. Realizamos las pruebas de escalabilidad en conjuntos de datos sintéticos. El algoritmo de agrupamiento *K-Modes* fue propuesto por primera

vez públicamente por Huang. Se trata de un algoritmo que admite los objetos de datos categóricos.

Inicialmente, se realiza una selección de los centroides al azar de los objetos de datos dados. Debido a la aleatoriedad en la selección de centroides iniciales, se trata de la solución local óptima. Dado que el algoritmo *K-Modes* proviene del algoritmo *K-Means*, también puede tratarse como un problema de optimización. El objetivo del algoritmo *K-Modes* es dividir el conjunto de datos  $S$  en  $K$  grupos aislados (llamados clústeres), para así minimizando la función de costo dada a continuación (Sangam y Om, 2015):

$$\text{Cost}(W, M) = \sum_{l=1}^k \sum_{i=1}^N w_{i,l} \text{dis}(x_i, m_{-l})$$

$$\text{Subject to } 0 \leq w_{i,l} \leq 1, \text{ for } 1 \leq i \leq N, \quad 1 \leq l \leq k$$

Este modelo, al igual que los demás, es descrito de forma secuencial como un proceso iterativo que se detiene al encontrar, en este caso, un valor óptimo local que agrupe datos con alta similitud. Algunos de los autores referenciados anteriormente que utilizaron esta técnica, se muestran en la tabla 20.

**Tabla 20.** Autores consultados que han usado el algoritmo *K-Modes* durante la última década

| Algoritmo      | Referencias bibliográficas      |
|----------------|---------------------------------|
| <i>K-Modes</i> | (Aissaoui <i>et al.</i> , 2019) |

Fuente: elaboración propia.

## 6.19. Kohonen Self-Organizing Map

El método está estrechamente relacionado con el análisis de conglomerados (partición) y otros métodos de análisis de datos. Si bien las redes SOM de Kohonen se han aplicado con éxito como herramienta de clasificación en varios dominios —incluido el reconocimiento de voz, la compresión de datos de imágenes, el reconocimiento de imágenes o caracteres, el control de robots y el diagnóstico médico—, su potencial como sustituto robusto para el análisis de conglomerados permanece relativamente sin investigar.

La red Kohonen fue desarrollada por el profesor Teuvo Kohonen, quien representó este tipo de mapa de características. La autoorganización es un caso de autoformación. Estos tipos de formación y aprendizaje se basan en las leyes de aprendizaje no supervisado por motivos biológicos y dependen de la información local. En la red neuronal, el mecanismo por el cual el cerebro almacena información funciona como un almacenamiento distribuido de información, dividido en muchas unidades. No se trata, como se pudiera pensar, de un almacenamiento centralizado en una unidad en particular. El diseño de un almacenamiento tal debe basarse en el concepto del almacenamiento de información a largo plazo en el cerebro y, en consecuencia, en la adaptación de las conexiones sinápticas entre neuronas. Las actividades de las neuronas presentan los patrones almacenados. Las redes con comportamientos de aprendizaje competitivo aprenden y organizan información sin recibir respuestas correctas para los patrones de entrada. Por tanto, los problemas encontrados en las redes neuronales supervisadas se eliminan.

*Kohonen Self-Organizing Map* (KSOM) consta de nodos dispuestos en una matriz 1D o 2D. Dado un vector de entrada  $x$ , el algoritmo SOM procede a identificar la neurona ganadora en el espacio de salida. El mapa de características calculado por el algoritmo denota una transformación no lineal. Las neuronas en el campo se someten a una competencia normal con base en una métrica de distancia  $d$ . En el ejemplo,  $d = 1$  (Ramadass y Abraham, 2012). La figura 42 muestra los diagramas generales para entender cómo funciona el algoritmo Kohonen SOM.

Algunos de los autores referenciados anteriormente que utilizaron esta técnica se muestran en la tabla 21.

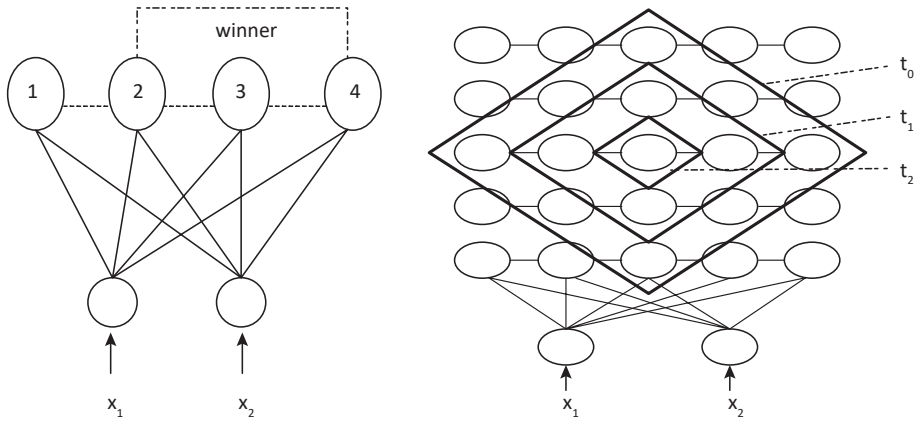
**Tabla 21.** Autores consultados que han usado el algoritmo *Kohonen Self-Organizing Map* durante la última década

| Algoritmo                          | Referencias bibliográficas |
|------------------------------------|----------------------------|
| <i>Kohonen Self-Organizing Map</i> | (Ramadass y Abraham, 2012) |

**Fuente:** elaboración propia.



**Figura 42.** (Izq.) Ejemplificación Kohonen SOM en 1D.  
(Der.) Ejemplificación Kohonen SOM en 2D



**Fuente:** Ramadass y Abraham (2012).



## Capítulo 7

# Aprendizaje supervisado: Factores y variables que influyen en el rendimiento académico

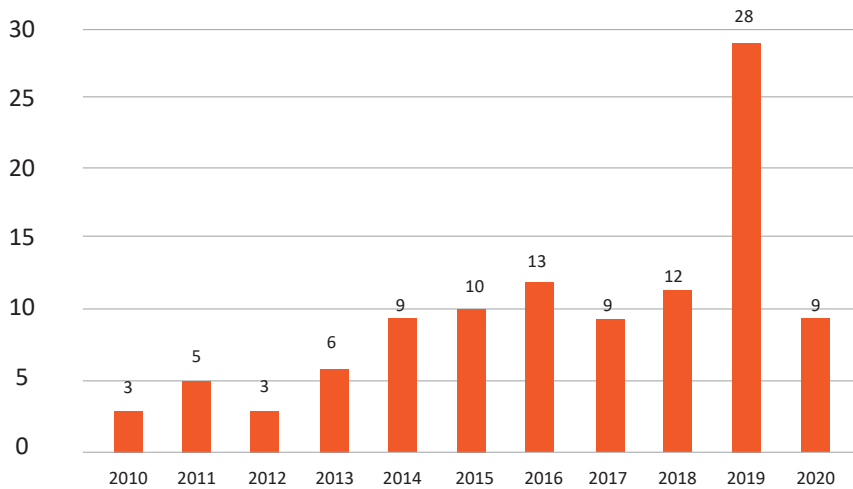
Antes de proceder a establecer las variables y los factores que se han tenido en cuenta en los que analizan el rendimiento académico con el uso de algoritmos de *Machine Learning* supervisados, daremos un vistazo a la información recopilada.

A partir de la información recopilada y filtrada por medio de los criterios mencionados anteriormente para el periodo entre 2010 y 2020 —información concerniente al uso del aprendizaje no supervisado para determinar el rendimiento académico de los estudiantes en educación superior—, se pudo generar, como se muestra en la figura 43, un panorama general de esta técnica de análisis de datos. Así mismo, la cantidad de artículos encontrados al aplicar las ecuaciones de búsqueda es de 104. En ese sentido, hay una creciente anual de la producción de artículos desde el 2010: este tipo de investigaciones se desarrollan debido a los resultados eficientes presentados en años anteriores.

En este capítulo se presenta de forma detallada un conglomerado de variables usadas por diferentes autores de múltiples referencias bibliográficas, encontradas en las bases de datos mencionadas anteriormente. Estas variables se trabajan con el propósito de predecir el rendimiento de estudiantes en educación superior mediante la construcción de modelos de aprendizaje automático supervisado. Las variables se agrupan en seis grupos principales, tal como se muestra en la figura 44, como una primera aproximación al

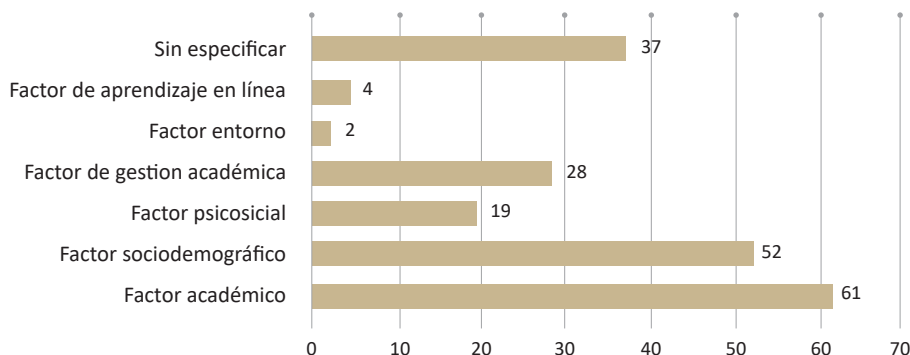
trabajo realizado de revisión bibliográfica en la última década. Se presenta el número de artículos agrupados por cada uno de los factores en los que se tuvieron en cuenta sus variables para determinar el rendimiento académico a partir de modelos de *Machine Learning*. Cada uno de los factores definidos se describirá en los numerales siguientes.

**Figura 43.** Agrupación de referencias bibliográficas por año



Fuente: elaboración propia.

**Figura 44.** Número de artículos que tienen en cuenta factores que influyen en el rendimiento académico



Fuente: elaboración propia.

## 7.1. Factor académico

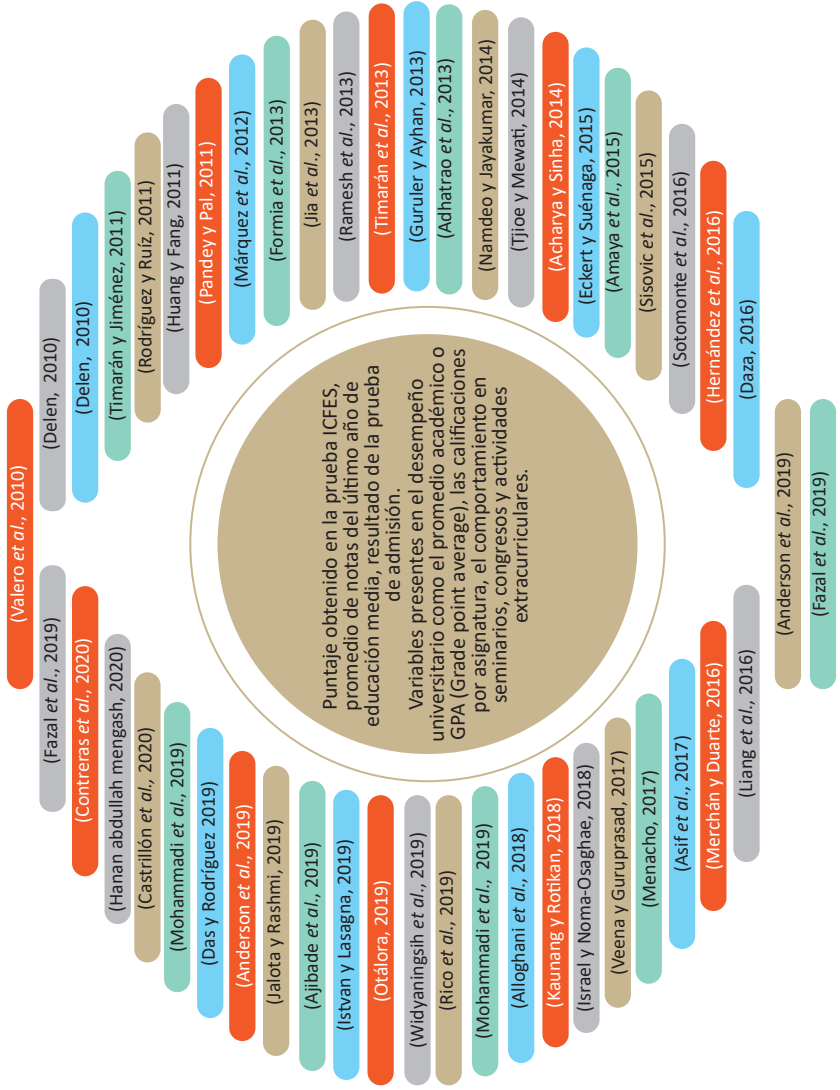
El factor académico tiene en cuenta un grupo de métricas cualitativas y cuantitativas que describen el proceso de aprendizaje de un estudiante. Estas métricas permiten a una determinada institución de educación llevar un registro antes y después del ingreso. Este factor es uno de los más importantes dado a que es utilizado constantemente para entrenar modelos de diferentes investigaciones, y de esta forma predecir el rendimiento académico a partir de un porcentaje de precisión. Este factor comprende mediciones de variables del estudiante antes de ingresar a la universidad. Es decir que, para Colombia, se tiene en cuenta el puntaje obtenido en la prueba ICFES, el promedio de notas del último año de educación media, el resultado de la prueba de admisión, entre otras. De igual forma, se analizan variables relacionadas con el desempeño universitario, como el promedio académico o GPA (*grade point average*), las calificaciones por asignatura, el comportamiento en seminarios, congresos y actividades extracurriculares.

Es importante resaltar que, en la mayoría de los artículos consultados, el factor con mayor aparición es el académico; este permite a los investigadores analizar un panorama inicial del rendimiento estudiantil, como se ve en la figura 45.

## 7.2. Factor sociodemográfico

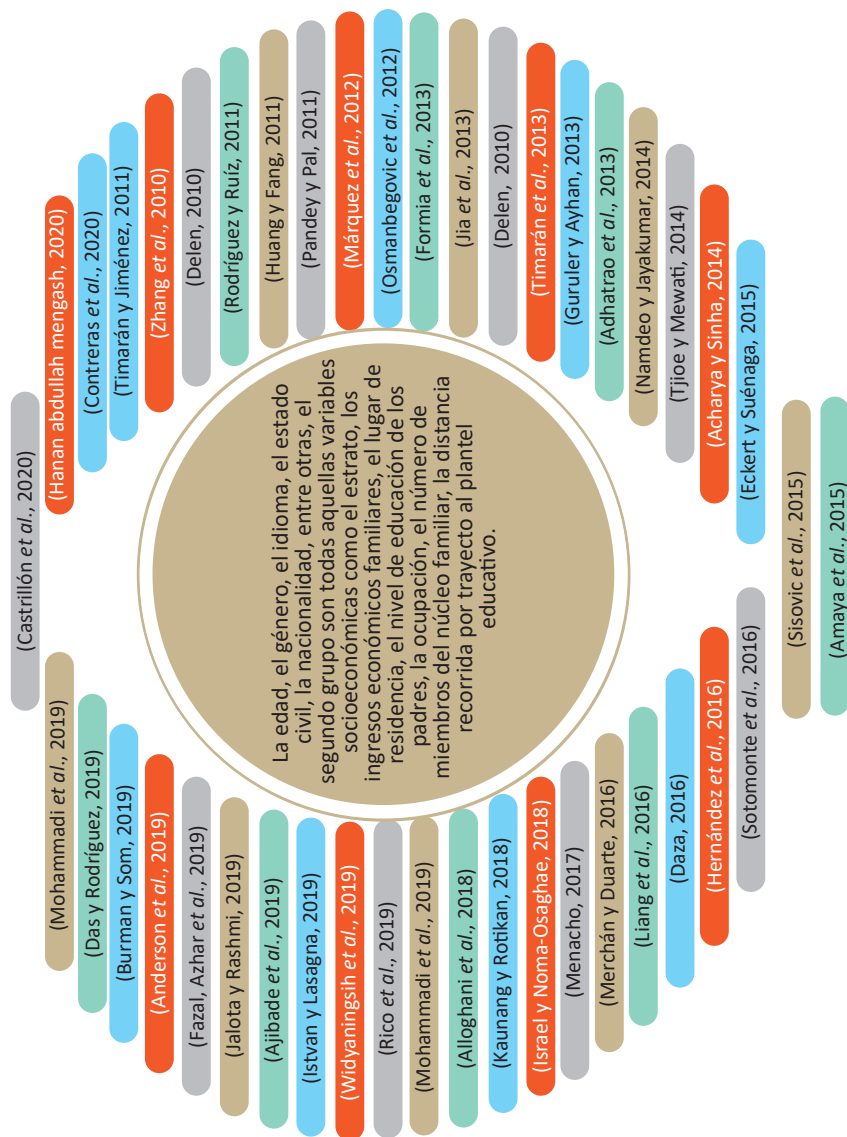
En diversas investigaciones y estudios especializados que tienen como tema el desempeño estudiantil, se han identificado variables sociodemográficas que intervienen en el rendimiento académico. Estas variables se organizan en dos grandes grupos: el primero tiene que ver con variables inherentes al estudiante —la edad, el género, el idioma, el estado civil, la nacionalidad, entre otras—; el segundo grupo comprende aquellas variables socioeconómicas como el estrato, los ingresos económicos familiares, el lugar de residencia, el nivel de educación de los padres, la ocupación, el número de miembros del núcleo familiar, la distancia recorrida por trayecto al plantel educativo (figura 46).

Figura 45. Investigaciones que consideran variables del factor académico



Fuente: elaboración propia.

**Figura 46.** Investigaciones que consideran variables del factor sociodemográfico

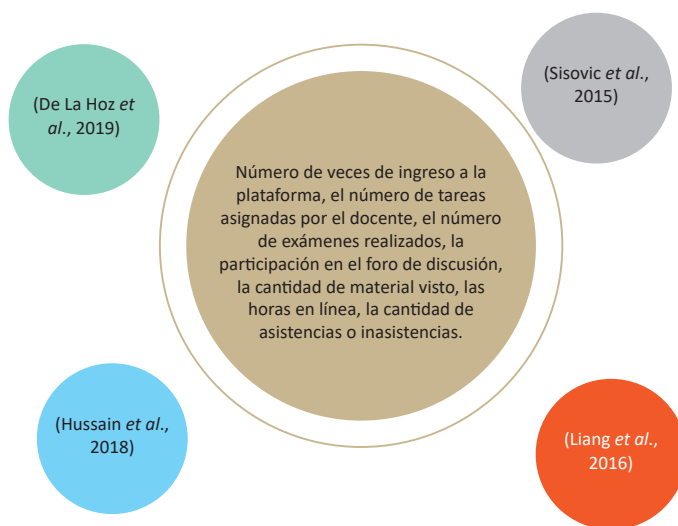


**Fuente:** elaboración propia.

### 7.3. Factor de aprendizaje en línea

La educación virtual ha tomado fuerza por las condiciones actuales del entorno dinámico y las facilidades que brinda esta alternativa de aprendizaje en la formación académica. Diferentes investigaciones se centraron en la interacción del estudiante con las plataformas de educación virtual disponibles. Es importante resaltar que el uso de este tipo de aplicaciones mejora el nivel de enseñanza a partir de una comparación con el modelo tradicional de enseñanza. Las variables que se tuvieron en cuenta fueron el número de veces de ingreso a la plataforma, el número de tareas asignadas por el docente, el número de exámenes realizados, la participación en el foro de discusión, la cantidad de material visto, las horas en línea, la cantidad de asistencias o inasistencias (figura 47).

**Figura 47.** Investigaciones que consideran variables del factor de aprendizaje en línea



**Fuente:** elaboración propia.

### 7.4. Factor de gestión académica

Una de las aspiraciones personales más importantes de los estudiantes al ingresar a la educación superior es el destacar académicamente con el fin de mejorar la calidad de vida personal y colectiva. Por tal motivo, este factor describe cómo ha sido la evolución del estudiante a lo largo de su vida



universitaria. Conocer este proceso le permite a la institución identificar fortalezas y debilidades para generar planes de acción y acompañamiento. En la figura 48 aparecen las variables que se tuvieron en cuenta como el año de ingreso a la universidad, el número de créditos, las becas obtenidas, el número de créditos cursados, el número de créditos aprobados, el número de créditos perdidos, la nota final de cada asignatura, el número de asignaturas cursadas, el número de asignaturas aprobadas, el número de asignaturas perdidas, el número de asignaturas repetidas, el número de veces que el estudiante ha perdido una asignatura.

**Figura 48.** Investigaciones que consideran variables del factor de gestión académica

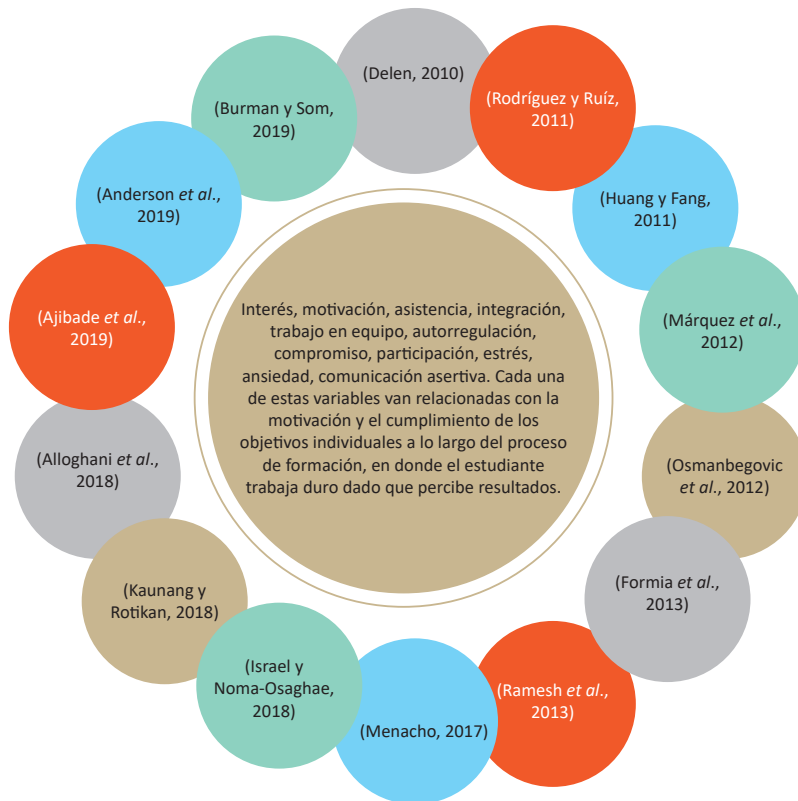


**Fuente:** elaboración propia.

## 7.5. Factor psicosocial

Este factor en particular se debe tener en cuenta porque abarca la relación del individuo con la sociedad. El acercamiento influencia al estudiante a comportarse según su círculo social lo exija; es decir, la persona podría desarrollar interés, motivación, asistencia, integración, trabajo en equipo, autorregulación, compromiso, participación, estrés, ansiedad, comunicación asertiva. Cada una de estas variables van relacionadas con la motivación y el cumplimiento de los objetivos individuales a lo largo del proceso de formación: el estudiante trabaja duro dado que percibe resultados (figura 49).

**Figura 49.** Investigaciones que consideran variables del factor psicosocial

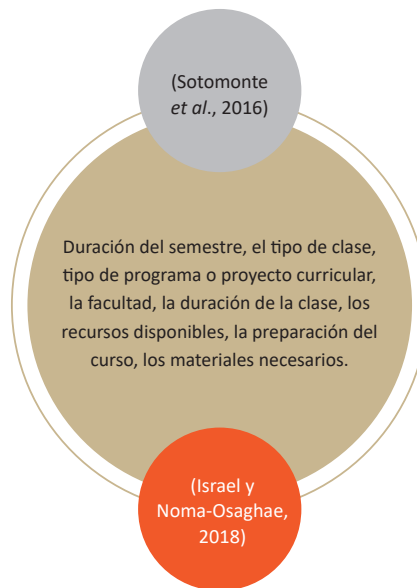


**Fuente:** elaboración propia.

## 7.6. Factor de entorno académico

El entorno académico engloba todo aquello relacionado con el sistema educativo, el contexto en el que el estudiante convive con la comunidad universitaria dentro de instalaciones físicas. Es por esto que se identificaron variables tales como la duración del semestre, el tipo de clase, el tipo de programa o proyecto curricular, la facultad, la duración de la clase, los recursos disponibles, la preparación del curso, los materiales necesarios (figura 50).

**Figura 50.** Investigaciones que consideran variables del factor de entorno académico



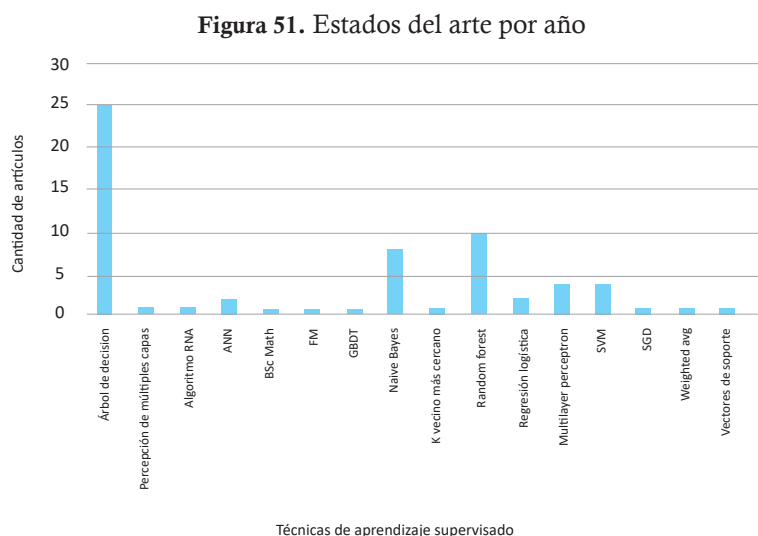
**Fuente:** elaboración propia.



## Capítulo 8

# Aprendizaje supervisado: Investigación referencial

A continuación, se realiza una descripción y un análisis detallado de cada uno de los artículos investigados durante el periodo 2010-2020, con el fin de evidenciar y hacer énfasis en los algoritmos de aprendizaje automático no supervisado aplicados, las variables utilizadas en los modelos, la eficiencia de las técnicas o métodos aplicados, el lenguaje de programación o *software* utilizado y los resultados finales de cada estudio. Estos artículos se agruparon por rangos de acuerdo con la cantidad de artículos por año, como se muestra en la figura 51.



**Fuente:** elaboración propia, basado en el estado del arte.

## 8.1. Periodo 2010-2013

En este periodo se utilizaron los algoritmos de *Machine Learning* redes neuronales, árboles de decisión (C4.5), máquinas de vectores de soporte (SVM), K-vecinos más cercanos (KNN), Naive Bayes (NB) y regresión logística. En la tabla 22 se encuentra información de las investigaciones realizadas periodo.

**Tabla 22.** Investigación documental periodo 2010-2013

| Año  | Autor                         | Nombre del artículo                                                                                                                            | Algoritmos usados                                                                                                                         | Tamaño de la muestra                                                        | Mejor resultado                                         | Software utilizado | Lugar    |
|------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|---------------------------------------------------------|--------------------|----------|
| 2010 | (Delen, 2010)                 | A Comparative Analysis of Machine Learning Techniques for Student Retention Management                                                         | Redes neuronales, árboles de decisión C4.5, máquinas de vectores de soporte, regresión logística, bosque aleatorio, árboles potenciados y | 16066                                                                       | Máquina de vectores de soporte y fusión de información. | -                  | EE.UU.   |
| 2010 | (Valero <i>et al.</i> , 2010) | Minería de datos: predicción de la deserción escolar mediante el algoritmo de árboles de decisión y el algoritmo de los k vecinos más cercanos | Árboles de decisión C4.5, k vecinos más cercanos                                                                                          | 723                                                                         | K vecinos más cercanos                                  | WEKA               | México   |
| 2010 | (Zhang <i>et al.</i> , 2010)  | Using Data Mining to improve student retention in HE: a case study.                                                                            | Árboles de decisión J48, máquinas de vectores de soporte, Naive Bayes                                                                     | 5458                                                                        | Naive Bayes                                             | Oracle 11g         | Portugal |
| 2011 | (Timaran y Jiménez, 2011)     | Detección de Patrones de Deserción Estudiantil en Programas de Pregrado de Instituciones de Educación Superior con CRISP-DM                    | Árboles de decisión C4.5, algoritmo <i>Apriori</i> y K-Means                                                                              | Universidad de Nariño 15805 y para la Institución Universitaria CESMAG 5010 | El conjunto de los algoritmos arrojó las variables      | WEKA               | Colombia |

| Año  | Autor                          | Nombre del artículo                                                                                   | Algoritmos usados                                  | Tamaño de la muestra | Mejor resultado           | Software utilizado | Lugar                 |
|------|--------------------------------|-------------------------------------------------------------------------------------------------------|----------------------------------------------------|----------------------|---------------------------|--------------------|-----------------------|
| 2011 | (Rodríguez y Ruíz, 2011)       | Indicadores de rendimiento de estudiantes universitarios: calificaciones versus créditos acumulados   | Regresión lineal                                   | 587                  | -                         | -                  | Uruguay               |
| 2011 | (Huang y Fang, 2011)           | Work in Progress - Prediction of Students' Academic Performance in an Introductory Engineering Course | MLR, MLP, RBF, SVM                                 | 239                  | SVM 62,3 %-69,0 %         | -                  | EE. UU.               |
| 2011 | (Pandey, 2011)                 | Data Mining: A prediction of performer or underperformer using classification                         | Naive Bayes                                        | 600                  | BSc Math 81,02 %          | MATLAB             | India                 |
| 2012 | (Márquez <i>et al.</i> , 2012) | Predicción del fracaso y el abandono escolar mediante técnicas de minería de datos                    | Árboles de decisión J48                            | 610                  | -                         | WEKA               | México                |
| 2012 | (Osmanbegovic y Suljic, 2012)  | Data Mining Approach for Predicting Student Performance                                               | Árboles de decisión J48, red neuronal, Naive Bayes | 257                  | Naive Bayes 76,65 %       | WEKA               | Bosnia                |
| 2013 | (Formia <i>et al.</i> , 2013)  | Caracterización de la deserción universitaria en la UNRN utilizando minería de datos                  | Árbol de decisión C4.5                             | 5 Grupos             | -                         | -                  | Argentina             |
| 2013 | (Jia y Mareboyana, 2013)       | Machine Learning Algorithms and Predictive Models for Undergraduate Student Retention                 | Árbol de decisión j48, vector máquina de soporte   | 771                  | Árbol de decisión 94,42 % | WEKA               | EE. UU. San Francisco |

| Año  | Autor                           | Nombre del artículo                                                                  | Algoritmos usados                                                                                        | Tamaño de la muestra | Mejor resultado              | Software utilizado      | Lugar    |
|------|---------------------------------|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|----------------------|------------------------------|-------------------------|----------|
| 2013 | (Ramesh <i>et al.</i> , 2013)   | Predicting Student Performance: A Statistical and Data Mining Approach               | Naive Bayes, Perceptrón Multicapa, Optimización mínima secuencial (SMO), árbol de decisión J48 y RERTree | 500                  | Perceptrón multicapa 72,38 % | WEKA                    | India    |
| 2013 | (Timarán <i>et al.</i> , 2013)  | Descubrimiento de perfiles de deserción estudiantil con técnicas de minería de datos | Árbol de decisión C4.5, <i>K-Means</i> , clustering                                                      | 6870                 | -                            | WEKA                    | Colombia |
| 2013 | (Guruler y Istanbulu, 2014)     | Modeling Student Performance in Higher Education Using Data Mining                   | Árbol de decisión                                                                                        | 6470                 | Modelo 1 87 %                | MSQL SERVER             | Turquía  |
| 2013 | (Adhatrao <i>et al.</i> , 2013) | Predicting Students' Performance Using ID3 And C4.5 Classification Algorithms        | ID3 y C4.5                                                                                               | 182                  | 75,275 %                     | RapidMiner, PHP y Excel | India    |

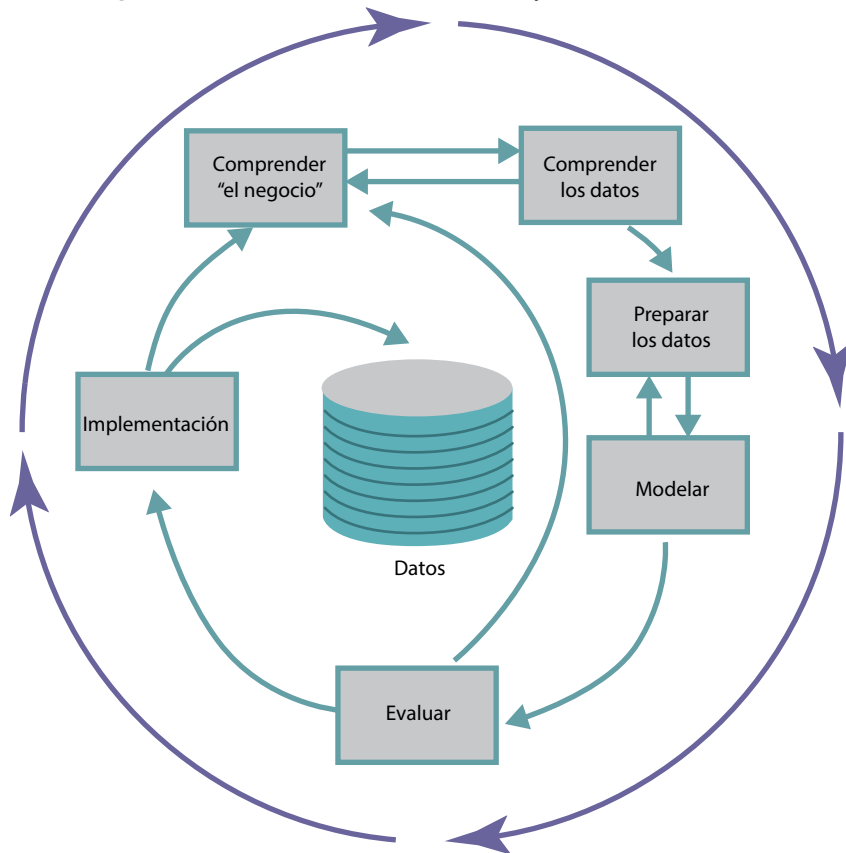
Fuente: elaboración propia.

Delen (2010) toma como base la comprensión del abandono para predecir la deserción de estudiantes de primer año. Se tuvieron en cuenta 16 066 datos históricos recolectados en cinco años consecutivos para la utilización de técnicas de minería de datos tanto individuales como en conjunto. Además, por medio de un análisis de sensibilidad, se determinó que las variables más significativas para el análisis son las educativas y socioeconómicas. La metodología utilizada fue el proceso transversal para la minería de datos (CRISP-DM por sus siglas en inglés), un proceso de seis pasos descritos en la figura 52. En el estudio se aplicaron cuatro métodos de clasificación: redes neuronales, árboles de decisión, máquinas de vectores de soporte y regresión logística; además de tres técnicas de conjunto (bosque aleatorio, árboles



potenciados y fusión de información). Al utilizar la totalidad de los datos, fue claro que la máquina de vectores de soporte es el mejor modelo, con una tasa de predicción del 87,23 %; en segundo lugar, está el árbol de decisión, con un 87,16 %. Luego de la limpieza de datos con una muestra equilibrada, se realizaron experimentos que dieron como mejor resultado a la máquina de vectores de soporte, con un 81,18 %. El modelo que produjo los mejores resultados fue el de fusión de información con un 82,10 %.

**Figura 52.** Proceso de minería de datos y validación cruzada



**Fuente:** Dursun *et al.* (2010).

Valero *et al.* (2010) aplicaron técnicas de minería de datos para predecir la deserción escolar en la Universidad Tecnológica de Izúcar de Matamoros. Utilizaron el análisis de variables socioeconómicas brindadas por estudios que realiza la institución. Para la investigación, se utilizaron los algoritmos

de árboles de decisión C4.5 y K-vecinos más cercanos. La investigación comprende las fases de integración, selección, limpieza y transformación, minería de datos, pruebas y verificación de resultados. Para la construcción de los modelos se usaron 726 datos en el *software* WEKA. Para árboles de decisión se obtuvo un 67,07 % de precisión, y para el caso de K-vecinos más cercanos se utilizó el método de validación cruzada, y se estableció un valor K de 5, y una precisión del 67,77 %, mayor que el del C4.5.

En su investigación, Zhang *et al.* (2010) intentaron mejorar la retención de estudiantes de educación superior con tecnologías de procesamiento de lenguaje natural y minería de datos para realizar un seguimiento, analizar el comportamiento académico y brindar una base para la intervención que evite el abandono escolar. Para el proyecto, se utilizó la metodología de gestión de cursos de sistemas de minería (MCMS), el cual crea una gestión del conocimiento basado en minería de datos y permite la integración de datos basada en modelos que los extraen de diferentes sistemas para informar y analizar.

Los datos pertenecen a la Universidad Thames Valley —corresponde a un total de 4223 estudiantes—. Se aplicaron los algoritmos Naive Bayes, máquina de vectores de soporte y árbol de decisión. En la tabla 23 se presenta la precisión de la predicción de cada uno de los algoritmos mencionados anteriormente; Naive Bayes dio el mejor resultado.

**Tabla 23.** Resultados de la predicción  
(Naive Bayes, Support Vector Machine, Decision Tree)

| Accuracy          | Naive Bayes (%) | Support Vector Machine (%) | Decision Tree (%) |
|-------------------|-----------------|----------------------------|-------------------|
| Negative Accuracy | 85,90           | 78,70                      | 71,20             |
| Positive Accuracy | 93,10           | 88,30                      | 91,40             |
| Average Accuracy  | 89,50           | 83,50                      | 81,30             |

Fuente: Zhang *et al.* (2010).

Timaran y Jiménez (2011) presentaron un proyecto de investigación cuyo objetivo es detectar patrones de deserción estudiantil por medio del análisis de datos socioeconómicos y académicos de programas de pregrado de la Universidad de Nariño y la Institución Universitaria CESMAG. La metodología utilizada es la CRISP-DM (*Cross-Industry Standard Process for*

*Data Mining*), que comprende seis fases: análisis del problema, análisis de los datos, preparación de los datos, modelado, evaluación y explotación. El tiempo de recolección de datos se da entre el 2004 y el 2011. Se obtuvo, de la Universidad de Nariño, información de 15 805 estudiantes, y de la Institución Universitaria CESMAG, datos de 5010 estudiantes. Se utilizó minería de datos y el algoritmo de clasificación de árboles de decisión con el método de validación cruzada de  $n$  pliegues; para la asociación se hizo uso del algoritmo *Apriori* y para la agrupación se utilizó la técnica particional con el algoritmo *K-Means*. Los resultados fueron los siguientes: para árboles de decisión, el porcentaje de precisión supera el 75 %; en la asociación se definieron reglas con una confianza del 100 % así como en la agrupación con *K-Means*, al elegir un  $k$  de 2. Estos fueron los patrones más representativos para caracterizar a los estudiantes que desertan. A partir del desarrollo del modelo, se concluyó que los factores que en mayor medida afectan la deserción en ambas instituciones son los académicos y los socioeconómicos, así como que el estudiante sea soltero, tenga un promedio bajo, haya perdido materias en los primeros semestres, provenga de un colegio público, sea menor de edad, pertenezca a la facultad de ingeniería y haya perdido entre una y dos materias en el área de ciencias básicas.

Rodríguez y Ruíz (2011) estudiaron el comportamiento de diferentes indicadores de rendimiento académico por medio de métodos de regresión lineal multivariante con factores como el promedio de calificaciones y el avance en la carrera; es decir, la razón entre los créditos acumulados y los créditos que teóricamente debió cursar el estudiante a lo largo del proceso educativo. Se analizaron 587 datos de estudiantes matriculados en Química de la Universidad de la República de Uruguay. Se llegó a la conclusión de que ambos indicadores se comportan de manera similar para predecir el rendimiento futuro. El promedio puede considerarse como una medida más amplia que el avance de la carrera, con una varianza en común del 37,8 % y los modelos predictivos de regresión solo difieren en un 8,3 % entre los indicadores. Se concluye que el rendimiento académico se ve afectado directamente por el rendimiento previo. Igualmente, la orientación hacia las metas de aprendizaje y la percepción de la capacidad influyen positivamente

en el indicador promedio, por tal motivo es importante medir el rendimiento académico con ambos indicadores.

Huang y Fang (2011) desarrollaron un modelo para predecir el rendimiento académico de estudiantes de un curso de Ingeniería Dinámica; para ello, recopilaron datos de 239 estudiantes de pregrado de tres semestres. Utilizaron los modelos de regresión lineal multivariante (MLR), redes neuronales de perceptrón multicapa (MLP), base radial de redes neuronales de función (RBF) y vector de máquinas de soporte (SVM). La metodología inició con la recopilación y el preprocesamiento de datos, luego se hizo el modelamiento y finalmente se realizó validación y comparación. Los resultados de exactitud de cada modelo son presentados en la tabla 24. Los resultados no tienen una diferencia significativa en cuanto al porcentaje de precisión de la predicción, pero en la mayoría de los casos el modelo con el porcentaje más alto es el correspondiente al vector de máquinas de soporte (SVM).

**Tabla 24.** Resultados de la predicción de los modelos MLR, MLP, RBF y SVM

| Model | Average prediction accuracy (%) |                               |                               | Percentage (%) of good predictions |                               |                               |
|-------|---------------------------------|-------------------------------|-------------------------------|------------------------------------|-------------------------------|-------------------------------|
|       | Sem. A,<br>n= 128               | Valida.<br>(Sem. B,<br>n= 58) | Valida.<br>(Sem. C,<br>n= 53) | Sem. A,<br>n= 128                  | Valida.<br>(Sem. B,<br>n= 58) | Valida.<br>(Sem. C,<br>n= 53) |
| MLR   | 90,3                            | 90,5                          | 88,2                          | 65,6                               | 56,9                          | 58,5                          |
| MLP   | 90,3                            | 90,6                          | 88,0                          | 66,4                               | 56,9                          | 52,8                          |
| RBF   | 89,1                            | 91,0                          | 89,0                          | 57,0                               | 72,4                          | 56,6                          |
| SVM   | 90,2                            | 90,9                          | 89,0                          | 64,1                               | 69,0                          | 62,3                          |

**Fuente:** Huang y Fang (2011).

En su trabajo, Pandey y Pal (2011) tuvieron como objetivo ayudar a una institución a reducir el bajo rendimiento académico y mejorar el desempeño estudiantil por medio del uso de la técnica de clasificación Naive Bayes. Los datos se recopilaron de distintos grados asociados con la universidad Dr. R. M. L. Awadh, en Faizabad, India. Se utilizó una muestra de seiscientos datos. Se usó Microsoft SQL Server 2005 para almacenarlos y Matlab como entorno de programación. La investigación comprende un proceso de minería de datos que incluye la preparación, la selección y la transformación

de los datos. En la tabla 25 se presenta la probabilidad de división en las categorías definidas. Se concluye que, según la categoría y la probabilidad, se puede predecir el comportamiento del estudiante de un periodo a otro.

**Tabla 25.** Resultados de la predicción

| Medium  | Category | Division | Class     | Probability |
|---------|----------|----------|-----------|-------------|
| English | Gen      | II       | BA(NC)    | 0,549218    |
|         |          | I        | BA(CA)    | 0,527331    |
|         |          | I        | BSc(Bio.) | 0,712938    |
|         |          | I        | BSc(Math) | 0,805456    |
|         |          | II       | Bcom      | 0,685979    |
|         | OBC      | II       | BA(NC)    | 0,596068    |
|         |          | I        | BA(CA)    | 0,520268    |
|         |          | I        | BSc(Bio.) | 0,717771    |
|         |          | I        | BSc(Math) | 0,810201    |
|         |          | II       | Bcom      | 0,758005    |
|         | SC/ST    | II       | BA(NC)    | 0,471241    |
|         |          | II       | BA(CA)    | 0,507795    |
|         |          | I        | BSc(Bio.) | 0,601875    |
|         |          | I        | BSc(Math) | 0,715802    |
|         |          | II       | Bcom      | 0,594066    |
| Hindi   | Gen      | II       | BA(NC)    | 0,467961    |
|         |          | II       | BA(CA)    | 0,585719    |
|         |          | I        | BSc(Bio.) | 0,376518    |
|         |          | I        | BSc(Math) | 0,504112    |
|         |          | II       | Bcom      | 0,484983    |

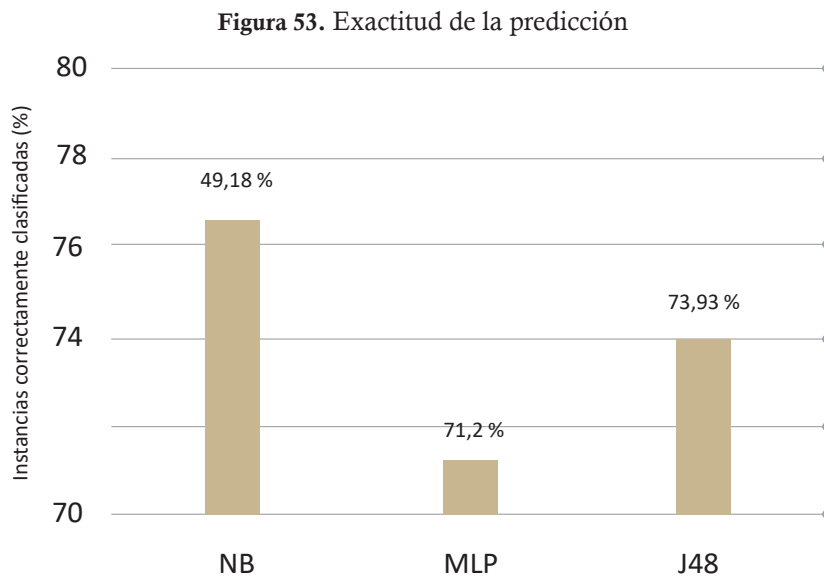
| Medium | Category | Division | Class     | Probability |
|--------|----------|----------|-----------|-------------|
| Hindi  | OBC      | II       | BA(NC)    | 0,568261    |
|        |          | II       | BA(CA)    | 0,652269    |
|        |          | I        | BSc(Bio.) | 0,428287    |
|        |          | I        | BSc(Math) | 0,553675    |
|        |          | II       | Bcom      | 0,634676    |
|        | SC/ST    | III      | BA(NC)    | 0,384808    |
|        |          | II       | BA(CA)    | 0,600667    |
|        |          | II       | BSc(Bio.) | 0,335702    |
|        |          | I        | BSc(Math) | 0,373642    |
|        |          | FAIL     | Bcom      | 0,456478    |

**Fuente:** Pandey y Pal (2011).

Márquez *et al.* 2012 aplican técnicas de minería de datos para predecir el fracaso escolar y la deserción. Usaron datos de 670 estudiantes de la Unidad Académica Preparatoria de la Universidad Autónoma de Zacatecas (UAPUAZ), de México. El *software* elegido para la realización de las pruebas fue WEKA. Se aplicaron diez algoritmos de caja blanca o árboles de decisión. Los cinco algoritmos de inducción de reglas de clasificación que se usaron son JRip, NNge, OneR, Prism y Ridor; y los cinco algoritmos de árboles de decisión que se utilizaron son J48, SimpleCart, ADTree, RandomTree y REPTree. Los resultados se obtuvieron al variar las condiciones de la siguiente manera:

- 77 atributos disponibles: ADTree 97,6 %
- 15 mejores atributos: ADTree 97,3 %
- 15 mejores atributos y considerando costo de clasificación: ADTree y SimpleCart 96,6 %
- 15 mejores atributos y previamente balanceando los datos de entrenamiento con el algoritmo SMOTE: ADTree 97,2 %
- El algoritmo de árboles de decisión ADTree arrojó el mejor porcentaje de precisión en los cuatro casos de estudio.

Osmanbegovic *et al.* (2012), a través de la aplicación de diferentes métodos y técnicas de datos, predijeron el éxito de los estudiantes de la Universidad de Tuzla, en la Facultad de Ingeniería, en un curso impartido en el periodo 2010-2011. Para su investigación tuvieron en cuenta diferentes factores y variables, recopilados por encuestas. El fin era aumentar la calidad del sistema de educación superior. La muestra fue de 257 estudiantes del curso Informática Empresarial y se utilizó el *software* WEKA. El proceso inició con un análisis de entrada con cuatro pruebas (chi-cuadrado, una prueba R, prueba de ganancia de información y prueba de relación de ganancia) para obtener la importancia de cada atributo. Los algoritmos de clasificación utilizados son perceptrón multicapa (MLP), Naive Bayes y árbol de decisión J48. En la figura 53 se puede observar que Naive Bayes es el mejor algoritmo para predecir el éxito del curso: si se aprobó o no, y mejorar de manera considerable el aprendizaje.



Fuente: Osmanbegovic *et al.* (2012).

Formia *et al.* (2013), con una creciente preocupación por la deserción presentada en la Universidad Nacional de Río Negro, utilizaron técnicas de minería de datos para predecir y entender el fenómeno. Al analizar, y realizar procesamiento de datos y proyecciones de los atributos en las clases, fue

posible sacar conclusiones que permitieron reducir el porcentaje de alumnos que abandonan la carrera. Se realizaron pruebas con el algoritmo de árboles de decisión C4.5 que es capaz de predecir con exactitud el 71,65 %.

Jia y Mareboyana (2013), por medio de la utilización de algoritmos de aprendizaje automático como árbol de decisión J48, máquina de vectores de soporte (SVM) y redes neuronales generados en el *software* WEKA, revelaron que los factores que más influyen en el abandono son el promedio acumulado de calificaciones y el tiempo en cursar la totalidad de créditos. Igualmente, se realizaron algoritmos conjuntos para crear una red neuronal de dos capas y una regresión lineal. Finalmente, y después de realizar diferentes pruebas, el algoritmo que arrojó la mejor precisión fue el SVM transformado, con un 94,29 %.

Ramesh *et al.* (2013) buscaron identificar los factores que influyen en el desempeño de los estudiantes en los exámenes finales. Usaron algoritmos de minería de datos que permiten predecir la calificación final y de esta forma identificar aquellos que están en riesgo de bajo rendimiento. Se recopilieron datos de nueve escuelas del distrito de Kancheepuram y se obtuvo una muestra de 500 estudiantes a los cuales se les aplicó un cuestionario que incluía preguntas con variables económicas, psicológicas y escolares. Se utilizó el *software* WEKA para implementar los algoritmos Naive Bayes, perceptrón multicapa, optimización mínima secuencial (SMO), árbol de clasificación J48 y REPTree. En la tabla 26 se presenta la precisión general del rendimiento de los clasificadores en el conjunto de datos; el perceptrón multicapa tiene la mejor precisión, con un 72,38 %. Se concluye que unas de las variables más significativas es la ocupación de los padres (tabla 26).

**Tabla 26.** Exactitud de la predicción

|           | Naive Bayes | MLP     | SMO     | J48     | REPTree |
|-----------|-------------|---------|---------|---------|---------|
| Exactitud | 49,5 %      | 72,38 % | 57,25 % | 64,88 % | 60,13 % |

**Fuente:** Ramesh *et al.* (2013).

Timarán *et al.* (2013) tuvieron por objetivo del proyecto detectar patrones de deserción estudiantil teniendo en cuenta datos socioeconómicos, académicos, disciplinarios e institucionales de estudiantes de pregrado de la



Universidad de Nariño y la institución universitaria IUCESMAG en Pasto Colombia. Se utilizaron técnicas de clasificación y *clustering* para conocer perfiles de estudiantes. Iniciaron su investigación procesando y transformando datos, para posteriormente aplicar técnicas de minería como árbol de decisión y *K-Means*. A partir de la generación en conjunto de ambos modelos, se obtuvo como resultado una confiabilidad mayor al 75 %, y que el patrón más común de deserción está determinado por un promedio bajo, tener materias perdidas en los primeros semestres, pagar una matrícula al y tener un bajo puntaje en las pruebas ICFES.

Guruler y Istanbulu (2014) muestran un caso de estudio que pretende revelar características individuales asociadas con el éxito académico de los estudiantes. Se utilizaron técnicas de clasificación para categorizar a los alumnos según su GPA o promedio acumulado. Los datos provienen de la Facultad de Economía y Ciencias Sociales de la Universidad Mugla Sitki Kocma; se obtuvieron 6470 registros. Se usaron árboles de decisión de Microsoft (MDT) del servicio de Microsoft SQL Server. Se realizaron, además, dos modelos que dieron como resultado que los factores más relevantes son los tipos de matrícula de la universidad y los ingresos familiares mensuales. Al revisar la elevación obtenida, el modelo 1 tuvo un valor de 1,74 y el modelo 2, de 1,36, lo que indica que tienen capacidad de predicción.

Adhatrao *et al.* (2013) desarrollaron un sistema capaz de predecir el desempeño de los estudiantes con técnicas de minería de datos bajo clasificación. Se tuvieron en cuenta datos como el género, las calificaciones, los exámenes de ingreso y los resultados del primer año de formación. Además, se aplicaron los algoritmos ID3 y C4.5. La implementación comprendió cinco etapas: recolección de información, limpieza de datos, aplicación de los algoritmos para obtener árboles de decisión, utilización de los datos de prueba en los árboles de decisión y finalmente el desarrollo de la interfaz en forma de aplicación web. Los datos fueron proporcionados en una hoja de cálculo Microsoft Excel 2003, y para tener mayor facilidad se introdujeron en una base de datos MySQL. Se alimentó esta hoja de forma tal que pudiera utilizarse como entrada en RapidMiner. Se generaron dos árboles de decisión con cada uno de los atributos, y se encontró información oculta útil para predecir el desempeño de los estudiantes y que fue plasmada en una

aplicación web que refleja una evaluación de desempeño. El resultado es una precisión del 75,145% para los algoritmos ID3 y C4.5. Por tanto, para un total de 182 estudiantes, el porcentaje promedio de precisión es de aproximadamente 75,275%.

## 8.2. Periodo 2014-2017

Para este periodo, luego de la investigación referencial realizada, los algoritmos utilizados fueron árboles de decisión C4.5 y regresión lineal. La información recopilada se encuentra en la tabla 27.

**Tabla 27.** Investigación documental periodo 2014-2017

| Año  | Autor                          | Nombre del artículo                                                                                                     | Algoritmos usados                                                                                                                        | Tamaño de la muestra | Mejor resultado                | Software utilizado | Lugar     |
|------|--------------------------------|-------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|----------------------|--------------------------------|--------------------|-----------|
| 2014 | (Namdeo y Jayakumar, 2014)     | Predicting Students Performance Using Data Mining Technique with Rough Set Theory Concepts                              | Naïve Bayes, MLP, J48 y <i>Random Forest</i>                                                                                             | 53                   | Naive Bayes 66,66%             | WEKA               | India     |
| 2014 | (Christian y Ayub, 2014)       | Exploration of Classification Using NBTtree for Predicting Students' Performance                                        | Árbol de decisión y Naive Bayes combinados                                                                                               | 2687                 | 81,46%                         | WEKA               | Indonesia |
| 2014 | (Acharya y Sinha, 2014)        | Early Prediction of Students Performance using <i>Machine Learning</i> Techniques                                       | Árbol de decisión C4.5, Perceptrón Multicapa (MLP), Naive Bayes (NB), K vecino más cercano (1-NN) y Optimización mínima secuencial (SMO) | 309                  | Árbol de decisión C4.5, 79,48% | WEKA               | India     |
| 2015 | (Eckert y Suénaga, 2015)       | Análisis de Deserción-Permanencia de Estudiantes Universitarios Utilizando Técnica de Clasificación en Minería de Datos | Árbol de decisión C4.5, Naive Bayes Aumentado a Árbol (TAN) y Clasificador Reglas OneR                                                   | 855                  | Árbol de decisión C4.5 80,234% | WEKA               | Argentina |
| 2015 | (Heredia <i>et al.</i> , 2015) | Student Dropout Predictive Model Using Data Mining Techniques                                                           | Árboles de decisión IDE3, J48                                                                                                            | 201                  | Árbol de decisión ID3          | WEKA               | Colombia  |

| Año  | Autor                            | Nombre del artículo                                                                                             | Algoritmos usados                                                                   | Tamaño de la muestra | Mejor resultado             | Software utilizado                     | Lugar          |
|------|----------------------------------|-----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|----------------------|-----------------------------|----------------------------------------|----------------|
| 2015 | (Sisovic <i>et al.</i> , 2015)   | Mining Student Data to Assess the Impact of Moodle Activities and Prior Knowledge on Programming Course Success | Árbol de decisión C4.5, JRip y PART                                                 | 153                  | C4.5 83,01 %                | WEKA                                   | Croacia        |
| 2016 | (Sotomonte <i>et al.</i> , 2016) | Hacia la construcción de un modelo predictivo de deserción académica basado en técnicas de minería de datos     | Árboles de decisión J48                                                             | 174                  | Árboles de decisión J48     | WEKA                                   | Colombia       |
| 2016 | (Hernandez <i>et al.</i> , 2016) | Comparative Study of Algorithms to Predict the Desertion in the Students at the ITSM-Mexico                     | Regresión logística, algoritmo de clústeres, árboles de decisión y redes neuronales | 134                  | Regresión logística 100 %   | Microsoft SQL Server Analysis Services | México         |
| 2016 | (Daza, 2016)                     | Un modelo basado en árboles de decisión para predecir la deserción estudiantil en la Educación Superior Privada | Árboles de decisión (C5.0, CRT y CHAID)                                             | 1761                 | Árboles de decisión 90,52 % | SPSS                                   | Perú           |
| 2016 | (Dhanalakshmi y Saravanan, 2016) | Opinion mining from student feedback data using supervised learning algorithms                                  | SVM, Naive Bayes, K vecino más cercano y clasificador de red neuronal               | 6433                 | K vecino más cercano 100 %  | Rapid Miner                            | Omán           |
| 2016 | (Wei <i>et al.</i> , 2016)       | Prediction of the scholarship using comprehensive development                                                   | Naive Bayes y árboles de decisión                                                   | -                    | Naive Bayes                 | -                                      | China          |
| 2016 | (Hasbun <i>et al.</i> , 2016)    | Extracurricular activities as dropout prediction factors in higher education using decision trees               | Árbol de decisión CART                                                              | 4840                 | Precisión 79,23 %           | R                                      | Chile          |
| 2016 | (Al-Barrak y Al-Razgan, 2016)    | Predicting Students Final GPA Using Decision Trees: A Case Study                                                | Árboles de decisión J48                                                             | 236                  | -                           | WEKA                                   | Arabia Saudita |
| 2016 | (Liang <i>et al.</i> , 2016)     | Big Data Application in Education: Dropout Prediction in Edx MOOCs                                              | SVM, <i>Random forest</i> , regresión logística y GDBT (árbol de decisión)          | 39                   | GB DT 88 %                  | -                                      | China          |

| Año  | Autor                        | Nombre del artículo                                                                                                              | Algoritmos usados                                                                              | Tamaño de la muestra | Mejor resultado                         | Software utilizado | Lugar             |
|------|------------------------------|----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|----------------------|-----------------------------------------|--------------------|-------------------|
| 2016 | (Merchán y Duarte, 2016)     | Analysis of Data Mining Techniques for Constructing a Predictive Model for Academic Performance                                  | J48, PART y Ridor                                                                              | 932                  | J48 78 %                                | WEKA               | Colombia          |
| 2017 | (Katara y Dubey, 2017)       | A Comparative Study of Classification Algorithms in EDM using 2 Level Classification for Predicting Student's Performance        | K vecino más cercano (KNN), Máquina de vectores de soporte (SVM) y Árbol de decisión J48       | 649                  | KNN+ SVM 98,98                          | MATLAB             | India             |
| 2017 | (Cuji <i>et al.</i> , 2017)  | Modelo predictivo de deserción estudiantil basado en árboles de decisión                                                         | Árbol de decisión CART                                                                         | 485                  | Árbol de decisión CART 94%              | R                  | Ecuador           |
| 2017 | (Yasir <i>et al.</i> , 2017) | Determining Single Tuition Fee of Higher Education in Indonesia: A Comparative Analysis of Data Mining Classification Algorithms | Naive Bayes, Multinomial Naive Bayes, Bernoulli Naive Bayes, Árbol de decisión y SVM           | 3258                 | Árbol de decisión 81,4 %                | Python             | Indonesia         |
| 2017 | (Abubakar y Hj, 2017)        | Prediction of Students' Performance in ELearning Environment Using Random Forest                                                 | Naive Bayes, K vecino más cercano, Árbol de decisión CART y Random Forest                      | 26                   | Random Forest 76,9% (mayor información) | MATLAB             | Nigeria y Malasia |
| 2017 | (Asif <i>et al.</i> , 2017)  | Analyzing undergraduate students' performance using educational Data Mining                                                      | Árbol de decisión, Naive Bayes, K vecino más cercano, Red neuronal y <i>Random Forest Tree</i> | 85                   | Naive Bayes 83,65 %                     | RapidMiner         | Pakistan          |
| 2017 | (Menacho Chio, 2017)         | Predicción del rendimiento académico aplicando técnicas de minería de datos                                                      | Regresión logística, árboles de decisión, Naive Bayes y Redes neuronales                       | 914                  | Naive Bayes 71 %                        | WEKA               | Perú              |
| 2017 | (Veena y Guruprasad, 2017)   | Comparative Analysis of Classification Algorithms for Student Performance                                                        | Naive Bayes, SMO y K vecino más cercano                                                        | 200                  | K-vecino más cercano 72,61 798 %        | WEKA               | India             |

| Año  | Autor                         | Nombre del artículo                                                                               | Algoritmos usados                                     | Tamaño de la muestra | Mejor resultado             | Software utilizado | Lugar  |
|------|-------------------------------|---------------------------------------------------------------------------------------------------|-------------------------------------------------------|----------------------|-----------------------------|--------------------|--------|
| 2017 | (Burgos <i>et al.</i> , 2018) | Data Mining of learning students' performance: A tutoring action plan to prevent academic dropout | Regresión logística, Red neuronal, SVM, PESFAM y SEDM | 100                  | Regresión logística 97,13 % | -                  | España |

Fuente: elaboración propia.

Namdeo y Jayakumar (2014) recopilaron datos educativos de 53 estudiantes del Masters in Computer Applications (MCA) en el 2007. Se generaron veinte atributos condicionales y de clase, y se usaron en la aplicación de algoritmos de minería de datos. Estos fueron Naive Bayes, perceptrón multicapa, árbol de decisión J4.8 y bosque aleatorio. Se utilizó el *software* WEKA y se obtuvo el resultado reflejado en la tabla 28: el algoritmo Naive Bayes da una mejor clasificación con respecto a los demás, considerándose como el mejor modelo.

**Tabla 28.** Comparación de algoritmos

| Algorithm             | Training (%) | Cross validation (%) | Percentage split (%) | Test on 2008 data (%) |
|-----------------------|--------------|----------------------|----------------------|-----------------------|
| Naive Bayes           | 84,31        | 66,66                | 38,29                | 31,57                 |
| Multilayer perceptron | 96,07        | 58,82                | 58,85                | 26,31                 |
| J48                   | 78,43        | 56,86                | 58,82                | 17,54                 |
| Random forest         | 100          | 56,86                | 58,82                | 24,56                 |

Fuente: Namdeo y Jayakumar (2014).

Christian y Ayub (2014) identificaron que el rendimiento de los estudiantes depende de diversos factores al tiempo. En su estudio, tuvieron en cuenta los datos personales, educativos, de admisión y académicos. Utilizaron la técnica de clasificación NBTree, un algoritmo híbrido de Naive Bayes y árboles de decisión combinados, útil para predecir el desempeño de los alumnos. Se realizó una clasificación de dos niveles: nivel universitario y nivel de facultad. La metodología de investigación del estudio consiste en la preparación,

la selección y la transformación de datos; y la clasificación del NBTree. Se utilizó el *software* WEKA con 2687 instancias y una exactitud del 81,46 %. Se concluyó que el género tenía influencia en la culminación de los estudios; en el caso de los graduados, el atributo crédito y el test score tienen efectos significativos en la duración y el promedio ponderado.

Marvin y Ayub (2014) identificaron que el rendimiento de los estudiantes depende de muchos factores, los cuales pueden afectar considerablemente. En el estudio tuvieron en cuenta los datos personales, educativos, de admisión y académicos. Utilizaron la técnica de clasificación NBTree el cual es un algoritmo híbrido de Naive Bayes y árboles de decisión combinados, útil para predecir el desempeño de los alumnos. Se realizó una clasificación de dos niveles, es decir el nivel universitario y el nivel de facultad. La metodología de investigación del estudio consiste en la preparación, la selección y transformación de datos; y la clasificación del NBTree. Se utilizó el *software* WEKA con 2687 instancias y una exactitud del 81,46 %. Se concluyó que el género tenía influencia para culminar los estudios, en el caso de los graduados el atributo crédito y el test score tienen efectos significativos en la duración y promedio ponderado que presenta un obstáculo en la finalización exitosa de los estudios.

Acharya y Sinha (2014) identificaron un conjunto de atributos de un grupo de estudiantes de pregrado con especialización en Ciencias de la Computación de universidades de Kolkata para realizar una predicción temprana del desempeño académico. Se realizó una encuesta a una muestra total de 309 estudiantes y se aplicaron los algoritmos árbol de decisión C4.5, perceptrón multicapa (MLP), Naive Bayes (NB), K-vecino más cercano (1-NN) y optimización mínima secuencial (SMO) por medio del *software* WEKA. Al realizar diferentes pruebas y comparar cada uno de los algoritmos, fue evidente que los mejores resultados se obtuvieron del árbol de decisión C4.5, con una exactitud del 79,8 %. Se determinó además que el atributo más importante para conocer la calificación del estudiante es el de ingresos familiares.

Eckert y Suénaga (2015) identifican factores que influyen en la deserción de estudiantes de Ingeniería en Informática de la Universidad Gestión Dachary en Argentina. Los datos recopilados comprenden información de

ingreso y la generada en el transcurso del periodo académico. Para el análisis se aplicaron algoritmos de clasificación como árboles de decisión, redes bayesianas y reglas. Los datos corresponden a los estudiantes ingresados en el periodo de tiempo comprendido entre el 2000 y el 2009; con un total de 855 casos. La metodología utilizada fue el proceso KDD que consiste en la integración y recopilación de datos, minería de datos y evaluación e interpretación de resultados. Las variables influyentes identificadas son las asignaturas aprobadas, la cantidad y el resultado de asignaturas cursadas, la procedencia y la edad de ingreso del estudiante. El algoritmo con mayor porcentaje de instancias clasificadas correctamente es el árbol de decisión C4.5, como se muestra en la tabla 29.

**Tabla 29.** Resultados de algoritmos de clasificación

| Algoritmos     | ICC (%) | Deserción | VP (%) | FP (%) | Precisión (%) |
|----------------|---------|-----------|--------|--------|---------------|
| J48 (C4.5)     | 80,234  | Des       | 79,1   | 18,7   | 79,7          |
|                |         | NoDes     | 81,3   | 20,9   | 80,8          |
| BayesNet (TAN) | 78,129  | Des       | 81,0   | 24,5   | 75,3          |
|                |         | NoDes     | 75,5   | 19,0   | 81,1          |
| OneR           | 76,608  | Des       | 83,7   | 30,0   | 72,1          |
|                |         | NoDes     | 70,0   | 16,3   | 82,3          |

**Fuente:** Eckert y Suénaga (2015).

(Amaya *et al.*, 2014) presentan un modelo predictivo de deserción estudiantil, aplicado a estudiantes de la Universidad Simón Bolívar, utilizando la herramienta WEKA. Para el estudio se desarrollaron fases, tales como la recopilación de información acerca de técnicas de minería de datos para elegir la adecuada, la preparación, preprocesamiento y la transformación de datos, la construcción y prueba del modelo y validación del mismo. Optaron por la utilización de árboles de decisión dado a que en la investigación realizada representan una ventaja con respecto a las demás técnicas de clasificación. Se construyó un árbol de decisión J4.8 y un ID3 con 201 instancias con un factor de confianza del 0,75 y 1. Teniendo como mejor resultado el algoritmo ID3 porque presenta mayor precisión, dado que tiene el mayor número de reglas y probabilidad de clasificación.



Sisovic *et al.* (2015), centran su investigación en el comportamiento de aprobación de la asignatura Programación 1, en la que los factores determinantes son la importancia del conocimiento previo y el trabajo continuo. En ese sentido, el trabajo analiza el aumento de estudiantes matriculados en el Departamento de Informática con un nivel superior en matemáticas en el examen nacional y la asistencia al seminario preparatorio. Los datos se extrajeron de la plataforma universitaria Moodle durante dos semestres de invierno (2013-2014 y 2014-2015) de la Universidad de Rijeka. Se utiliza el *software* WEKA y se realizan las fases de definición de la pregunta de investigación, los atributos que incluyen información de los estudiantes y el procesamiento previo de los datos. Se obtienen un conjunto de 153 instancias. Los clasificadores utilizados en la investigación fueron árboles de decisión J48 (C4.5), JRip y PART. En la tabla 30 se presenta la precisión de los clasificadores con costo de aprendizaje sensible o sin él: el algoritmo C4.5 arrojó mejor resultado y se llegó a la conclusión que la asistencia al seminario preparatorio y el trabajo continuo son las variables influyentes.

**Tabla 30.** Precisión de los clasificadores

| Cost-sensitive learning | C4.5 (%) | JRip (%) | PART (%) |
|-------------------------|----------|----------|----------|
| NO                      | 83,01    | 83,01    | 83,66    |
| YES                     | 83,01    | 78,43    | 81,05    |

**Fuente:** Sisovic *et al.* (2015).

Sotomonte *et al.* (2016) hicieron uso de la minería de datos con el fin de generar un modelo de árbol de decisión utilizando el algoritmo de clasificación J48 por medio de la herramienta WEKA. Este método les permitiría predecir la deserción académica. La metodología usada es la CRISP-DM, que consiste en un ciclo de vida iterativo. Para el estudio se recopilieron datos desde el 2009 de las fuentes de información de estadísticas de deserción de los últimos años, zonificación y discriminación por programa, área del conocimiento, condiciones sociales, puntaje ICFES, pensum y acuerdo académico. Al aplicar el algoritmo, se obtuvo que las variables que influyen son la cantidad de materias vistas, el género y la distancia del lugar de residencia al plantel educativo.



Hernandez *et al.* (2016) presentan un estudio de la predicción de los posibles riesgos de desertar de los estudiantes del Instituto Tecnológico Superior de Misantla-México. Utilizaron información personal y académica de los educandos. Los algoritmos aplicados fueron regresión logística, clústeres, árboles de decisión y redes neuronales. En la tabla 31 se puede observar la precisión y la sensibilidad de los algoritmos a partir de la muestra de 134 estudiantes; la regresión logística es el algoritmo con mejores resultados de exactitud y precisión.

**Tabla 31.** Medidas de precisión y sensibilidad

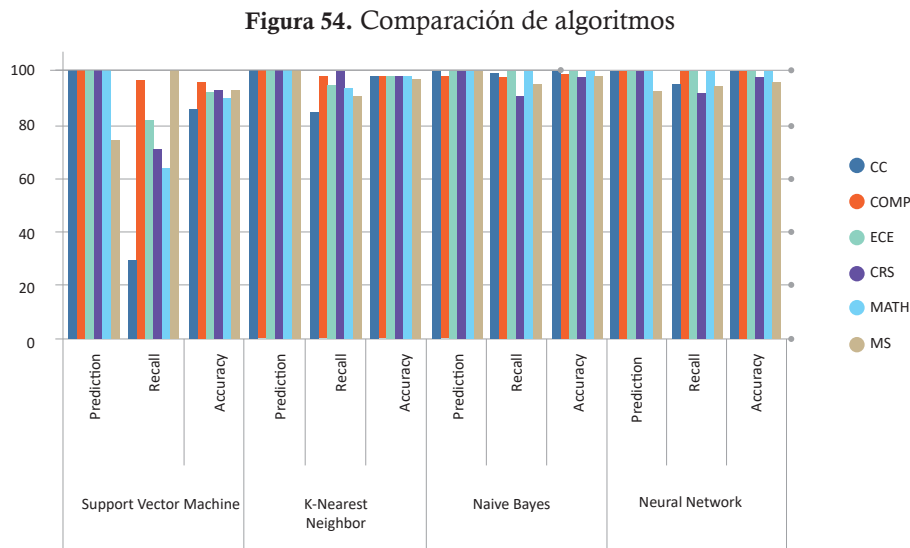
|              | Clúster (%) | Árbil de Decision (%) | Regresión Logística (%) | Red Neuronal (%) |
|--------------|-------------|-----------------------|-------------------------|------------------|
| Precisión    | 100         | 0                     | 100                     | 100              |
| Sensibilidad | 75          | 0                     | 100                     | 88               |
| Exactitud    | 95          | 80                    | 100                     | 98               |
| Medida-F     | 86          | 0                     | 100                     | 93               |
| Elevación    | 5           | 0                     | 5                       | 5                |

**Fuente:** Hernández *et al.* (2016).

Daza (2016) hace uso de las técnicas de minería de datos para predecir el abandono en la educación superior privada. La metodología utilizada es la CRIPs-DM. Se identificaron todas las posibles variables de entrada, se realizó un análisis de la base de datos, se construyó una propuesta de nuevas variables, se desarrolló una correlación y se creó el algoritmo. Los datos se obtuvieron de la Universidad Privada César Vallejo, de la escuela profesional de Ingeniería de Sistemas. Se extrajeron datos de 1761 estudiantes. El modelo por utilizar son los árboles de decisión —algoritmos C5.0, CRT y CHAID—. El mejor resultado, con un 90,52 % de precisión, fue el del C5.0 que permitió determinar las variables que aportan a la investigación, es decir el código universitario, el rendimiento académico de la universidad y del colegio, la edad, el sexo, el currículo, el ciclo y la cantidad de créditos aprobados. Además, se logró distinguir que en los primeros tres semestres está el índice de deserción más alto.

Dhanalakshmi y Saravanan (2016) hicieron uso de algoritmos de aprendizaje supervisado para explorar la minería de opiniones a partir de los

resultados de la encuesta de evaluación del módulo Middle East College de Omán. Se utilizó la herramienta de análisis de datos de código abierto rapidminer y se compararon los algoritmos máquina de vectores de soporte (SVM), Naive Bayes (NB), K-vecino más cercano (KNN) y redes neuronales (NN). La metodología se basa en realizar la evaluación del módulo y una encuesta relámpago, procesar los datos de la encuesta, extraer entidades y características para hallar la polaridad individual, entrenar el modelo con algoritmos SVM, KNN, NB y NN para clasificar las clases positivas y negativas; y finalmente comparar los resultados del rendimiento de los algoritmos aplicados. En la figura 54 se puede visualizar el comportamiento de cada algoritmo: el resultado del KNN muestra una precisión del 100 % y se considera el mejor de todos.



Fuente: Dhanalaksmi y Saravanan (2016).

Wei *et al.* (2016) se enfocaron en predecir la cantidad de estudiantes que adquieren una beca a partir de la prueba de calidad integral con el algoritmo Naive Bayes (NB) y el árbol de decisión para probar el modelo de clasificación en el experimento. Los datos fueron proporcionados por una Universidad de China, en donde se realizó un procesamiento de datos eliminando los faltantes y reduciendo la dimensión de la base para realizar posteriormente una estadística. La variable utilizada es el puntaje global

promedio (GPA), el cual determina las fallas o éxitos en el examen que otorga la beca. Para la validación del modelo, se hizo uso de una validación cruzada diez veces, una matriz de confusión y AUC. El método con mejor precisión es el NB, que tiende a ser del 100 % de precisión.

Hasbun *et al.* (2016) estudian la importancia de las actividades extra-curriculares para predecir la deserción de los estudiantes de Licenciatura en Ciencias (Ingeniería y Negocios). Se recopilieron 4840 datos de la Universidad Adolfo Ibáñez de cuatro cohortes, entre los años 2008 a 2011. Entre las variables analizadas para un primer conjunto de datos estaba la información agregada del estudiante y la media de calificaciones acumuladas (GPA); para el segundo grupo de datos se utilizó información detallada de las matrículas de los estudiantes en los cursos, y sus calificaciones finales. El tercer conjunto comprendía información del curso; el cuarto conjunto incluyó los deportes practicados por los estudiantes; y, finalmente, el último conjunto contenía información de la escuela secundaria del estudiante. Se entrenaron dos modelos: uno que incluye todos los datos y el otro en el que se eliminan las calificaciones y actividades de valor de créditos obligatorios. Estos modelos se verificaron mediante la validación cruzada del árbol de decisión CART y el *software* R con dieciocho variables independientes, con una precisión de predicción del 79,29 %. Se llegó a la conclusión de que los estudiantes con una eficiencia deportiva del 68,33 % o menos son más propensos a la deserción.

Al-Barrak y Al-Razgan (2016) utilizaron minería de datos educativos para predecir el GPA final y las calificaciones en dos cursos. Se revisaron datos de los expedientes de las mujeres que se graduaron de la facultad de ciencias de la Universidad King, y se recopilaron un total de 236 datos de estudiantes, organizados en una hoja de Microsoft Excel. Dentro los expedientes se encontraban atributos de nombre, ID, GPA final, semestre de graduación, especialidad, nacionalidad, campus y todos los cursos tomados por el estudiante, además de la calificación. Para la aplicación del algoritmo se usó el *software* WEKA. El algoritmo arrojó que, por semestre, los cursos más importantes son los plasmados en la tabla 32.

**Tabla 32.** Cursos determinantes por semestre

| Semester | Course Name                                 |
|----------|---------------------------------------------|
| 3        | <i>Java 1 (the only specialized course)</i> |
| 4        | <i>Database Principles</i>                  |
| 5        | <i>Software Engineering 1</i>               |
| 6        | <i>Information security</i>                 |
| 7        | <i>Computer Ethics</i>                      |
| 8        | <i>Project 2</i>                            |

**Fuente:** Al-Barrak y Al-Rasgan (2016).

Liang *et al.* (2016) estudiaron predicciones en cursos en línea masivamente abiertos. La variable objetivo a analizar es el registro del comportamiento de aprendizaje del estudiante. Se utilizaron los datos de un mes para predecir si los estudiantes abandonarían el curso en los próximos diez días. Se recolectaron 39 datos de los cursos de la plataforma XuetnagX. La metodología se basa en la extracción de los datos de la plataforma Edv, el procesamiento de los datos, la ingeniería de funciones y la prueba de rendimientos de los diferentes modelos de clasificación de aprendizaje supervisado, como máquina de vectores de soporte (SVM), regresión logística, bosque aleatorio y potenciación del gradiente en árbol de decisión (GBDT). En la tabla 33 se presentan los resultados de la precisión de la predicción, y se concluye que GBDT tiene el mejor resultado, de 88%.

**Tabla 33.** Resultados de los modelos aplicados

| Modelo | Precisión |
|--------|-----------|
| LR     | 0,65      |
| SVM    | 0,65      |
| RF     | 0,86      |
| GBDT   | 0,88      |

**Fuente:** Liang *et al.* (2016).

Merchán y Duarte (2016) construyeron un modelo predictivo para analizar el rendimiento académico de los estudiantes. Aplicaron métodos y técnicas

de minería de datos en 932 estudiantes de Ingeniería de Sistemas de la Universidad El Bosque.

Estos investigadores realizaron minería de datos con algoritmos J48, PART y RIDOR, y utilizaron el *software* WEKA que arrojó resultados de precisión de clasificación de las iteraciones definidas para el desempeño académico. El árbol de decisión J48 logró clasificar correctamente alrededor de 78 % de las instancias evaluadas, como se muestra en la tabla 34.

**Tabla 34.** Tasas de precisión

|               | J48 (%)                       | PART (%)                      | RIDOR (%)                     |
|---------------|-------------------------------|-------------------------------|-------------------------------|
| Iteración # 1 | R: 81<br>P: 62,70<br>S: 83,30 | R: 64,7<br>P: 57,1<br>S: 78,8 | R: 72,2<br>P: 71,8<br>S: 69,6 |
| Iteración # 2 | R: 27,8<br>P: 42,3<br>S: 0    | R: 16,7<br>P: 42,6<br>S: 0    | R: 47,9<br>P: 40,2<br>S: 22,7 |
| Iteración # 3 | R: 43,2<br>P: 42,7<br>S: 18,2 | R: 34,8<br>P: 40,3<br>S: 25,4 | R: 44,5<br>P: 37,5<br>S: 22,1 |
| Iteración # 4 | R: 86,8<br>P: 85,3<br>S: 76,9 | R: 85,4<br>P: 83,3<br>S: 66,7 | R: 64,9<br>P: 55,3<br>S: 25   |

**Fuente:** Merchán y Duarte (2016).

Katare y Dubey (2017) investigaron el proceso de clasificación de 649 datos de estudiantes en grupos predeterminados de clases. Utilizaron algoritmos de dos niveles: máquina de vectores de soporte (SVM), K-vecino más cercanos (KNN) y el árbol de decisión para calcular la medida de ganancia de la información. Todos los experimentos se realizan en Java con la herramienta MATLAB. Dadas las comparaciones realizadas, el modelo de segundo nivel —es decir, la dupla de algoritmos SVM Y KNN— proporciona una tasa de precisión superior al 98 %, mayor que la arrojada por el árbol de decisión J48. Se concluye que el rendimiento académico de los estudiantes es importante para lograr decisión, así como para brindar mejores servicios educativos.

Cuji *et al.* (2017) construyen un modelo predictivo de deserción estudiantil con el fin de pronosticar la probabilidad que un alumno abandone un

programa académico. La metodología utilizada es la KDD (*knowledge discovery in database*). Esta metodología dicta etapas de selección, procesamiento, transformación, minería de datos, y evaluación. Se utilizó el algoritmo CART en *software* R a una muestra total de 485 estudiantes. Los resultados del estudio se obtuvieron a partir del análisis de once variables. Dado que el árbol de decisión se formó a partir de una estructura de cuatro niveles, se concluyó que, según el orden, la variable nivel es la que más incidía en la deserción. El modelo tuvo una certeza del 94 %.

En su estudio, Yasir *et al.* (2017) tienen la necesidad de calcular la tarifa de matrícula única del estudiante (UKT). En ese sentido, se procedió a realizar un estudio que comparara cinco algoritmos de clasificación de minería de datos: Gaussian Naive Bayes, multinomial Naive Bayes, Bernoulli Naive Bayes, árbol de decisión y máquina de vectores de soporte (SVM). El propósito era encontrar el algoritmo ideal para determinar las clases de UKT. La metodología diseñada proponía tres etapas principales: la definición de variables, la normalización de datos y, por último, la prueba de entrenamiento. Se usaron 230 datos y una evaluación cruzada de diez veces. El algoritmo con precisión promedio más alta el árbol de decisión, con un 81,4 %. Se utiliza el árbol de decisión para definir las clases de 3258 estudiantes.

Abubakar y Hj (2017) tienen como objetivo predecir el desempeño de los estudiantes teniendo en cuenta la interacción de estos en entornos virtuales, al analizar las calificaciones obtenidas y los conocimientos previos. Se hizo uso de algoritmos de aprendizaje supervisado: bosque aleatorio, K-vecino más cercano y árbol de decisión. La metodología consiste en la preparación y el procesamiento de registros de 26 estudiantes y una selección de once atributos según características definidas. El *software* utilizado fue MATLAB y se realizó una validación cruzada de diez veces para entrenar el modelo. Se hizo una comparación de cada una de las técnicas definidas. Se las describe en términos de exactitud, precisión, recuperación y medida F. Es notable que el algoritmo Naive Bayes superó a los demás con un 92,3 % de exactitud frente al 76,9 %, 69,2 % y 61,5 % de los algoritmos K-vecino más cercano, bosque aleatorio y árbol de decisión.

Asif *et al.* (2017) tienen por objetivo estudiar el desempeño de los estudiantes universitarios, predecir la academia según el resultado final de un

programa de cuatro años y estudiar las progresiones típicas para combinar el sistema con los resultados de la predicción. Se dividió la muestra en dos grupos: los estudiantes de nivel bajo y de nivel alto. Los datos corresponden a estudiantes de Licenciatura en Tecnología de la Información, y se tienen en cuenta las cohortes académicas del 2007-2008 y 2008-2009. Las variables relacionan las calificaciones de preadmisión y las calificaciones de todos los cursos a lo largo de la formación académica. Debido al desequilibrio de los datos, se optó por utilizar algoritmos de clasificación para aplicarlos a los operadores disponibles en el *software RapidMiner*. Concluye que se puede predecir la graduación de los estudiantes por medio de variables académicas, socioeconómicas y demográficas.

Menacho Chiok (2017) aplica técnicas de minería de datos con uso de algoritmos como regresión logística, árboles de decisión, redes bayesianas y redes neuronales. Los datos se obtuvieron de 914 estudiantes matriculados en el curso de Estadística General de la Universidad Nacional Agraria La Molina (UNALM), de los semestres 2013-II y 2014-I. El fin es predecir la clasificación final categorizada en *desaprobado* o *aprobado*. Para la comparación se utiliza la matriz de confusión, una matriz de contingencia que muestra la distribución de la clasificación real y la predicha. Se usó el programa WEKA, compuesto por un conjunto de algoritmos. La red Naive Bayes arroja una mayor precisión, y el 71 % de las instancias se están clasificando correctamente. Las variables que influyen en el resultado del curso son el promedio ponderado, la situación del curso, la nota en diferencial y el número de veces que llevó diferencial.

El grupo *Comparative Analysis of Classification Algorithms for Student Performance* (2017) presenta un modelo predictivo con base en el desempeño de los estudiantes de segundo semestre. Se aplicaron técnicas de clasificación de minería de datos y se desarrollaron modelos de clasificación como Naive Bayes, máquina de vectores de soporte (SMO) y K-vecino más cercano. Además, también se busca conocer las calificaciones de los estudiantes más influyentes a lo largo del proceso educativo. La metodología trabajada consiste comprender el alcance del problema a resolver, recolectar y seleccionar los datos de doscientos estudiantes de Ciencias de la Información que han completado el título y finalmente la preparación de datos: extracción y

transformación para generar un conjunto de calificaciones de los estudiantes. Al realizar el respectivo análisis con técnicas de clasificación, se procedió a identificar el mejor algoritmo. El algoritmo K-NN tiene el rendimiento más alto con 72,61 % de precisión, seguido por SMO y Naive Bayes, con 71,22 % y 66,94 % respectivamente.

Debido a la gran cantidad de datos generados en la educación virtual, Burgos *et al.* (2018) utilizaron herramientas de regresión logística con fines de clasificación para predecir el abandono académico. Para la aplicación, se usaron datos históricos de alumnos de cinco materias diferentes del programa de Licenciatura en Ingeniería Informática de la Universidad Abierta de Madrid. Los resultados de los modelos se integraron en un plan de tutoría, y se redujo así la tasa de deserción en un 14 % con respecto a años anteriores.

### 8.3. Periodo 2018-2020

Algunos de los más importantes algoritmos que se usan en este periodo son árbol de decisión, bosque aleatorio, Naive Bayes, máquina de vectores de soporte, entre otros. En esta sección se describe el trabajo realizado por cada uno de los autores, ejercicio que permite reconocer diferentes propuestas metodológicas y los resultados de los modelos propuestos. La información más importante se encuentra en la tabla 35.

**Tabla 35.** Investigación documental periodo 2018-2020

| Año  | Autor                        | Nombre del artículo                                                                                 | Algoritmos usados                           | Tamaño de la muestra | Mejor resultado                      | Software utilizado | Lugar  |
|------|------------------------------|-----------------------------------------------------------------------------------------------------|---------------------------------------------|----------------------|--------------------------------------|--------------------|--------|
| 2018 | (Patil <i>et al.</i> , 2018) | Prediction system for student performance using Data Mining classification                          | Árbol de decisión (ID3, Improve ID3 y C4.5) | 50                   | Árbol de decisión (Improve ID3) 74 % | -                  | India  |
| 2018 | (Rico y Sánchez, 2018)       | Diseño de un modelo para automatizar la predicción del rendimiento académico en estudiantes del IPN | Naive Bayes                                 | 94                   | Naive Bayes 63,83 %                  | HTML5 y PHP        | México |



| Año  | Autor                            | Nombre del artículo                                                                                                       | Algoritmos usados                                                          | Tamaño de la muestra | Mejor resultado            | Software utilizado | Lugar           |
|------|----------------------------------|---------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|----------------------|----------------------------|--------------------|-----------------|
| 2018 | Fernandes <i>et al.</i> , 2019)  | Educational Data Mining: predictive analysis of academic performance of public-school students in the capital of Brazil   | Árbol de decisión (CM y DS)                                                | 834                  | Árbol de decisión (CM) 94% | H2O Flow           | Brasil          |
| 2018 | (Maya <i>et al.</i> , 2018)      | Diseño de un Modelo predictivo aplicando Minería de Datos para identificar causas de Deserción Estudiantil Universitaria  | Árbol de decisión J48 y REPtree                                            | 170                  | J48 60%                    | WEKA               | Valle de Toluca |
| 2018 | Adekitan y Noma-Osaghae, 2018)   | Data Mining approach to predicting the performance of first year student in a university using the admission requirements | Random forest, árbol de decisión, Naive Bayes, regresión logística y Rprop | 1445                 | Regresión logística 50,23% | KNIME y Orange     | Nigeria         |
| 2018 | (Kaunang y Rotikan, 2018)        | Students' academic performance prediction using Data Mining                                                               | Árbol de decisión y random forest                                          | 300                  | Árbol de decisión 63,51%   | WEKA               | Indonesia       |
| 2018 | (Alloghani <i>et al.</i> , 2019) | Application of Machine Learning on Student Data for the Appraisal of Academic Performance                                 | Árbol de decisión, red neuronal y Naive Bayes                              | 147                  | Redes neuronales 93,1%     | WEKA               | Rusia           |

| Año  | Autor                               | Nombre del artículo                                                                                                                               | Algoritmos usados                                                                          | Tamaño de la muestra | Mejor resultado                                    | Software utilizado                   | Lugar     |
|------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|----------------------|----------------------------------------------------|--------------------------------------|-----------|
| 2018 | (M. Hussain <i>et al.</i> , 2019)   | Using <i>Machine Learning</i> to predict student difficulties from learning session data                                                          | Redes neuronales artificiales, SVM, regresión logística, Naive Bayes y árboles de decisión | 100                  | SVM 80%                                            | MATLAB                               | China     |
| 2019 | (González <i>et al.</i> , 2019)     | Nodes: Plataforma para la predicción de deserción escolar utilizando técnicas de inteligencia artificial                                          | Árbol de decisión                                                                          | -                    | Árbol de decisión 25,9%                            | Azure <i>Machine Learning</i> Studio | México    |
| 2019 | (Mohammadi <i>et al.</i> , 2019)    | Comparative study of supervised learning algorithms for student performance prediction                                                            | K vecino más cercano, Naive Bayes y árbol de decisión                                      | 230                  | K vecino más cercano 54,64%                        | WEKA                                 | Japón     |
| 2019 | (Rico <i>et al.</i> , 2019)         | Construcción e implementación de un modelo para predecir el rendimiento académico de estudiantes universitarios mediante el algoritmo Naive Bayes | Naive Bayes                                                                                | 122                  | 70,42%                                             | HTML5 y PHP                          | México    |
| 2019 | (Widyaningsih <i>et al.</i> , 2019) | A semi-supervised learning approach for predicting student's performance: first-year students case study                                          | <i>K-Means</i> clustering y Naive Bayes clasificación                                      | 140                  | K means clustering y Naive Bayes clasificación 96% | -                                    | Indonesia |
| 2019 | (Nieto <i>et al.</i> , 2019)        | Usage of <i>Machine Learning</i> for strategic decision making at higher educational institutions                                                 | Árbol de decisión, random forest y regresión logística                                     | 6100                 | Random forest 91,93%                               | KNIME                                | Colombia  |

| Año  | Autor                          | Nombre del artículo                                                                                                                                             | Algoritmos usados                                                                                                         | Tamaño de la muestra       | Mejor resultado                                                                       | Software utilizado | Lugar     |
|------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|----------------------------|---------------------------------------------------------------------------------------|--------------------|-----------|
| 2019 | (Otálora, 2019)                | Modelo para la identificación de patrones de desempeño académico estudiantil para fortalecer el acompañamiento académico en la Universidad Nacional de Colombia | Árbol de decisión y clasificadores bayesianos.                                                                            | -                          | Árbol de decisión 65,61 %                                                             | -                  | Colombia  |
| 2019 | (Istvan y Lasagna, 2019)       | Sistema Informático para la Detección Temprana de Deserción Estudiantil Universitaria                                                                           | Árbol de decisión                                                                                                         | -                          | -                                                                                     | WEKA y Rapid Miner | Argentina |
| 2019 | (Albán, 2019)                  | Contribuciones a la predicción de la deserción universitaria a través de minería de datos                                                                       | 1)Árbol de decisión y regresión logística<br>2)Árbol de decisión<br>3)Multilayer perceptrón, bosque aleatorio, PART y J48 | 1)1178<br>2)3162<br>3)6000 | 1)Árbol de decisión 91,7%<br>2)Árbol de decisión 98,71 %<br>3)Metaclasificador 99,5 % | WEKA               | Perú      |
| 2019 | (Zarate, 2019)                 | Reconocimiento de patrones de deserción de estudiantes universitarios basados en modelos de clasificación                                                       | Redes neuronales, ID3, C4.5                                                                                               | 970                        | Árbol de decisión C4.4 87,77%                                                         | Aplicativo propio  | Perú      |
| 2019 | (Heredia <i>et al.</i> , 2019) | Extracción y procesamiento de datos para modelado de trayectorias académicas en cursos universitarios                                                           | Árbol de decisión                                                                                                         | 240                        | -                                                                                     | Python             | Uruguay   |

Leonardo Emiro Contreras Bravo, Giovanni Mauricio Tarazona Bermúdez, Angie Paola Alemán Cardona, Jhonatan Estiven Ruiz Vanegas, Brandon Herney Hernández Arcila, Daniela Garzón Trujillo, Óscar Eduardo Sánchez Gómez, Marisol Ángel Cifuentes

| Año  | Autor            | Nombre del artículo                                                                                                                                                                                                                   | Algoritmos usados                                                           | Tamaño de la muestra | Mejor resultado        | Software utilizado   | Lugar   |
|------|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|----------------------|------------------------|----------------------|---------|
| 2019 | (Loja, 2019)     | Aplicación de técnicas de minería de datos en el contexto del rendimiento académico en la Universidad de Cuenca                                                                                                                       | Árbol de decisión, Bosque aleatorio y K vecino más cercano                  | 43313                | Bosque aleatorio 75,6% | Python y WEKA        | Ecuador |
| 2019 | (Zambrano, 2019) | Análisis de retención de estudiantes en la Carrera de Negocios Internacionales de la Facultad de Ciencias Económicas y Administrativas de la UCSG, durante los dos primeros años de la carrera, mediante técnica de Minería de Datos. | Árbol de decisión (J48 y C4.5) y Naive bayes                                | -                    | C4.5 91%               | KNIME                | Ecuador |
| 2019 | (Vega, 2019)     | Modelo de pronóstico de rendimiento académico de alumnos en los cursos del programa de estudios básicos de la Universidad Ricardo Palma usando algoritmos de <i>Machine Learning</i>                                                  | Redes Neuronales Artificiales (RNA), Gradient boosting machine y XGBoosting | 9118                 | XGBoosting 89,73%      | R, Python & MS Excel | Perú    |

| Año  | Autor                          | Nombre del artículo                                                                                                                                                                     | Algoritmos usados                                                                                                                                                              | Tamaño de la muestra | Mejor resultado              | Software utilizado             | Lugar    |
|------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|------------------------------|--------------------------------|----------|
| 2019 | (Forero <i>et al.</i> , 2019)  | <i>Machine Learning</i> for the identification of students at risk of academic desertion                                                                                                | Averaged Perceptrón, Bayes Point Machine, Boosted Decision Tree, Decision Forest, Decision Jungle, Locally Deep Support Vector Machine, Logistic Regression & Neuronal Network | 3201                 | Boosted Decision Tree 93,5%  | Azure Machine Learning Studio. | Colombia |
| 2019 | (Masa-banda y Zapata, 2019)    | Modelos basados en la minería de datos para determinar factores de deserción estudiantil en la Facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi. | J48, random forest y secuencial minimal optimization (SMO)                                                                                                                     | 1457                 | Conjunto 92%                 | SPSS, Eviews & WEKA            | Ecuador  |
| 2019 | (Ajibade <i>et al.</i> , 2019) | Educational Data Mining: enhancement of student performance model using ensemble methods                                                                                                | Máquina de vectores de soporte (SVM), K-vecino más cercano (KNN) y árbol de decisión                                                                                           | 500                  | Árbol de decisión 91,5%      | XAPI                           | Malaysia |
| 2019 | Jalota y Agrawal, 2019)        | Analysis of educational Data Mining using classification                                                                                                                                | J48, máquina de vectores de soporte, Naive Bayes, bosque aleatorio y perceptrón multicapa                                                                                      | 480                  | Multilayer perceptrón 76,06% | WEKA                           | India    |
| 2019 | (Aman <i>et al.</i> , 2019)    | A predictive model for predicting students' academic performance                                                                                                                        | Algoritmo propuesto (árboles de decisión + árboles logísticos), Bosque aleatorio & J48,                                                                                        | 1021                 | Algoritmo propuesto 83,09%   | WEKA                           | Pakistan |

| Año  | Autor                            | Nombre del artículo                                                                                                                        | Algoritmos usados                                           | Tamaño de la muestra | Mejor resultado             | Software utilizado      | Lugar    |
|------|----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|----------------------|-----------------------------|-------------------------|----------|
| 2019 | (Anderson <i>et al.</i> , 2019)  | Predicting Graduation at a Public R1 University                                                                                            | SVM, árbol de decisión, regresión logística y SGD           | 14706                | SGD 82,4%                   | PYTHON                  | EE. UU.  |
| 2019 | (Burman y Som, 2019)             | Predicting students' academic performance using support vector machine                                                                     | SVM (Kernel lineal y Kernel radial)                         | 1000                 | SVM radial 90,97%           | R                       | India    |
| 2019 | Rodriguez-Marek, 2019)           | A predictive analytics system for forecasting student academic performance: insights from a pilot project at Eastern Washington University | K-vecino más cercano y Naive Bayes                          | 279                  | K vecino más cercano 70%    | -                       | EE. UU.  |
| 2019 | (De La Hoz <i>et al.</i> , 2019) | Metodología de Aprendizaje Automático para la Clasificación y Predicción de Usuarios en Ambientes Virtuales de Educación                   | Clustering jerárquico, lógica Fuzzy y K-vecinos más cercano | 403                  | K vecino más cercano 91%    | R                       | Colombia |
| 2019 | (Mohammadi <i>et al.</i> , 2019) | Comparative study of supervised learning algorithms for student performance prediction                                                     | K vecino más cercano, Naive Bayes y árbol de decisión       | 230                  | K vecino más cercano 54,64% | WEKA                    | Pakistan |
| 2020 | (Bhutto <i>et al.</i> , 2020)    | Predicting students' academic performance through supervised machine learning                                                              | Regresión logística y SVM                                   | 480                  | SVM 78%                     | WEKA y JUPITER (PYTHON) | Pakistan |

| Año  | Autor                             | Nombre del artículo                                                                                                    | Algoritmos usados                                                             | Tamaño de la muestra | Mejor resultado           | Software utilizado | Lugar          |
|------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|----------------------|---------------------------|--------------------|----------------|
| 2020 | Viloria <i>et al.</i> , 2020)     | Data Mining applied in school dropout prediction                                                                       | JRip, NNge, OneR, Prism, Ridor, ADTree, J48, RandomTree, REPTree & SimpleCart | -                    | AD tree 98,6 %            | WEKA               | Colombia       |
| 2020 | (Castrillón <i>et al.</i> , 2020) | Predicción del rendimiento académico por medio de técnicas de inteligencia artificial                                  | J48                                                                           | 460                  | 91,67 %                   | WEKA               | Colombia       |
| 2020 | (Mengash, 2020)                   | Using Data Mining techniques to predict student performance to support decision making in university admission systems | Ann, Svm, árbol de decisión, Naive Bayes                                      | 2,039                | ANN 79,22 %               | -                  | Arabia Saudita |
| 2020 | (Contreras <i>et al.</i> , 2020)  | Aplicación del aprendizaje automático como estrategia de predicción de la interrupción académica universitaria         | K-vecino más cercano, SVC, perceptrón y árbol de decisión                     | 2114                 | Árbol de decisión 85,80 % | PYTHON             | Colombia       |

Fuente: elaboración propia.

Patil *et al.* (2018) realizan un desarrollo centrado en un sistema que permita analizar el desempeño de los estudiantes con conceptos de minería de datos y modelos de clasificación. Para lograrlo, se preparó un conjunto de datos con información de los estudiantes, y específicamente de su género, calificaciones, clasificación de los exámenes de ingresos y resultados del tercer año del grupo anterior. Las metodologías se dividen en dos fases: la primera es la fase de aprendizaje, en la que se examinan los datos de entrenamiento y se generan reglas de clasificación. La segunda fase es la de clasificación, en la que los datos de prueba se clasifican en clases según el conjunto de datos de entrenamiento. Se aplicaron los algoritmos de árbol de decisión C4.5, ID3 e ID3 mejorado. Los resultados arrojaron que el clasificador ID3

mejorado da como resultado la mejor precisión con un 74 %, seguido por el clasificador C4.5 con 72 %. El clasificador ID3 original proporciona una precisión del 70 %.

Rico y Sánchez (2018) plantearon un modelo predictivo del rendimiento académico de estudiantes del Instituto Politécnico Nacional. Analizaron calificaciones de actividades académicas y la calificación final de 94 estudiantes matriculados en la carrera de ingeniería del IPN. La metodología de la investigación implaba la integración y la recopilación de datos, el preprocesado de la base obtenida, la organización visual que permitiera la minería de datos, una evaluación de patrones y finalmente la generación del conocimiento. Para la fase de minería de datos, la técnica utilizada fue el algoritmo de Naive Bayes, que se programó en la plataforma HTML5 y PHP para posteriormente publicar los resultados. Los conjuntos de datos se construyeron de manera aleatoria. La exactitud del primer conjunto fue del 59,57 % y del segundo, del 68,09 %. En promedio, la exactitud del modelo predictivo fue del 63,83 %.

Fernandes *et al.* (2019) realizan un análisis predictivo del desempeño académico de los estudiantes de las escuelas públicas del Distrito Federal de Brasil en los periodos escolares del 2015 y 2016. Se hizo un análisis estadístico para conocer los conjuntos de datos establecidos, los cuales tienen variables del inicio del año escolar y variables académicas recopiladas. La metodología utilizada se basa en el proceso estándar transversal tradicional para la minería de datos (CRISP-DM), que comprende seis pasos para lograr la predicción del desempeño académico. Esta metodología implica la comprensión de los objetivos y los requisitos del proyecto, entender los datos, prepararlos, realizar el modelado, la evaluación y el despliegue. Se usaron 485 872 registros para aplicar el algoritmo de potenciación del gradiente (GBM) y se pudo concluir que las variables que influyen de manera significativa en el desempeño deficiente de los estudiantes son el vecindario y la situación de la vivienda del estudiante.

Maya *et al.* (2018) desarrollaron un modelo de predicción con base en técnicas y métodos de clasificación. El fin era identificar patrones de influencia en la deserción estudiantil. El trabajo se inicia con un análisis exploratorio, correlacional y explicativo. Además, se implementó la metodología



de la cebolla y el ciclo DM. Los métodos que se utilizaron son los algoritmos de clasificación JRIP, OneR, ZeroR, J48, y REPTree; y de selección CfsSubsetEval y BestFirst). La muestra fue de 170 estudiantes. Se identificaron cada una de las causas que impactan en la deserción y la reprobación escolar. El mejor algoritmo fue el J48, con un factor de confianza del 0,60, 0,75 y 1. Entonces, un estudiante está en riesgo de abandonar sus estudios si tiene planes de matrimonio, se le dificultan las asignaturas de informática, ha presentado exámenes extraordinarios de materias como programación, base de datos, redes de cómputo, soporte técnico, sistemas operativos o inglés; y si su tiempo libre lo ocupa para otras actividades diferentes a estudiar e ir a bibliotecas.

Adekitan y Noma-Osaghae (2018) examinaron la relación entre los requisitos de ingreso y el rendimiento académico de los alumnos en su primer año, con el fin de predecir el rendimiento de estudiantes de una universidad de Nigeria. Se analizaron 1445 expedientes que datan del periodo comprendido entre 2005 y 2009; se tuvieron en cuenta diferentes características tanto académicas como demográficas. Se hizo uso de la plataforma KNIME para seleccionar, mediante muestreo estratificado, los datos de entrenamiento y se usaron seis algoritmos: random forest, tree ensemble, decision tree, Naive Bayes, logistic regression y resilient *backpropagation* (Rprop) multi-layer perceptron). El algoritmo con mejor comportamiento fue Rprop MLP, con un 49,078% de precisión. El segundo mejor algoritmo predictor fue random forest, con 48,38%, seguido por el árbol de decisión (47,24%) y la regresión logística (46,25%).

Para verificar que el modelo aplicado en la plataforma KNIME fuera el adecuado, se utilizó la plataforma de minería de datos de Orange. Se aplicaron los algoritmos árbol de decisión, bosque aleatorio, red neuronal, Naive Bayes, regresión logística y AdaBoost. El algoritmo con mayor valor de métrica *recall* para la clasificación es la red neuronal, con un 51,9%. En segundo lugar, está el random forest (50,3%). El tercer y cuarto lugar en porcentaje de precisión lo ocupan el árbol de decisión (43,6%) y Adaboost (42,9%).

Kaunang y Rotikan (2018) proponen un modelo para predecir el rendimiento académico de los estudiantes de ciencias de la computación con técnicas de minería de datos. La recopilación de los datos se hizo mediante

cuestionarios que contenían datos demográficos, el GPA anterior y la información de antecedentes familiares. Se aplicaron dos modelos de minería de datos: árbol de decisión y bosque aleatorio. El estudio es realizado en el *software* WEKA para evaluar el modelo de predicción propuesto. Para evaluar la calidad de la clasificación se utilizaron cuatro métricas: exactitud, precisión, recuperación y medida. Al evaluar las técnicas de clasificación se obtienen resultados que se presentan en la tabla 36. El árbol de decisión es el mejor modelo cuando se lo compara con *random forest*: recibir el valor de precisión más alto, del 66,9%.

**Tabla 36.** Resultado del método de clasificación

| Evaluation Metrics | Decision Tree |            | Random Forest |            |
|--------------------|---------------|------------|---------------|------------|
|                    | 5-fold CV     | 10-fold CV | 5-fold CV     | 10-fold CV |
| Accuracy           | 62,85%        | 66,85%     | 61,14%        | 61,14%     |
| Precision          | 0,543         | 0,622      | 0,572         | 0,573      |
| Recall             | 0,629         | 0,669      | 0,611         | 0,611      |
| F-measure          | 0,565         | 0,632      | 0,588         | 0,589      |

**Fuente:** Kaunang y Rotikan (2018).

Alloghani *et al.* (2019) trabajan en su investigación el análisis predictivo y utilizan árboles de decisión, redes neuronales y algoritmos Naive Bayes con el fin de clasificar y agrupar patrones de aprendizaje previamente identificados que ayuden a identificar los posibles tipos de rendimiento académico. Para la aplicación se tuvieron en cuenta estudiantes de catorce países y se obtuvieron doce atributos categóricos y cuatro numéricos. El ejercicio se realizó con el *software* WEKA, y se evaluó a partir de las medidas *verdadero positivo*, *falso positivo*, *precisión*, *recuperación* y *puntuación*. Los resultados muestran que el algoritmo red neuronal se ajusta mejor a los datos, con un 93,1 % de precisión, seguido del algoritmo C4.5 (92,7 %), CART (91,7 %) y Naive Bayes (87,1 %).

En cuanto al trabajo de Hussain *et al.* (2019) support vector machines (SVMs, el objetivo es predecir los posibles obstáculos que los estudiantes puedan enfrentar en una sesión posterior al curso de diseño digital. Los datos analizados fueron extraídos de los registros del sistema de aprendizaje

mejorado por la tecnología. Los investigadores aplicaron algoritmos de aprendizaje automático como redes neuronales artificiales (ANN), máquinas de vectores de soporte (SVM), regresión logística (LR), Naive Bayes (NBC) y árbol de decisión (DT). Se tomaron datos de estudiantes de primer año de licenciatura de la Universidad de Génova. La fuente de los algoritmos es MATLAB y las variables de entrada se relacionan con actividades de resolución de ejercicios y tareas. Para medir la veracidad del modelo, se adoptaron métricas de precisión, puntuación F1, recuperación, RMSE, estadística Kappa y la gráfica ROC. En ese sentido, los mejores modelos son el ANN y SVM, con una precisión del 75 %. Posteriormente se hizo un entrenamiento con características seleccionadas mediante el método de inversión Alpha y se logró una precisión cercana al 80 % con el modelo SVM.

En su proyecto, González *et al.* (2019) hacen énfasis en el estudio y el análisis de la deserción en la zona centro del Estado de Veracruz por medio del uso de Azure *Machine Learning* Studio, plataforma que permite realizar pruebas de comportamiento de los datos. Se eligió un algoritmo para implementar una plataforma tecnológica que pueda predecir el porcentaje de posibilidad de deserción de un estudiante al matricularse en el nivel medio superior. El algoritmo sugerido es un árbol de decisión impulsado de dos clases.

Mohammadi *et al.* (2019) implementan algoritmos de clasificación —K-vecinos más cercano (KNN), Naive Bayes y árbol de decisión— para minar datos de 230 estudiantes de la Facultad de Ciencias de la Computación de la Universidad de Kabul. Se hace un proceso de minería de datos que incluye la preparación, la selección y la transformación de características y la implementación del modelo óptimo —en este caso, el algoritmo KNN (tabla 37)—.

**Tabla 37.** Precisión de modelos

| Algoritmo | Precisión (%) |
|-----------|---------------|
| KNN       | 0,5464        |
| DT        | 0,5325        |
| NB        | 0,4616        |

**Fuente:** Mohammadi *et al.* (2019).

Rico *et al.* (2019) hacen la presentación de un proyecto que comprende la construcción, la evaluación y la aplicación de un modelo que predice el rendimiento académico de estudiantes universitarios por medio de la utilización de la técnica Naive Bayes. Los datos totales adquiridos son de estudiantes de Ingeniería del Instituto Politécnico Nacional. Se recopilieron 122 datos para entrenamiento, y 71 para predicción.

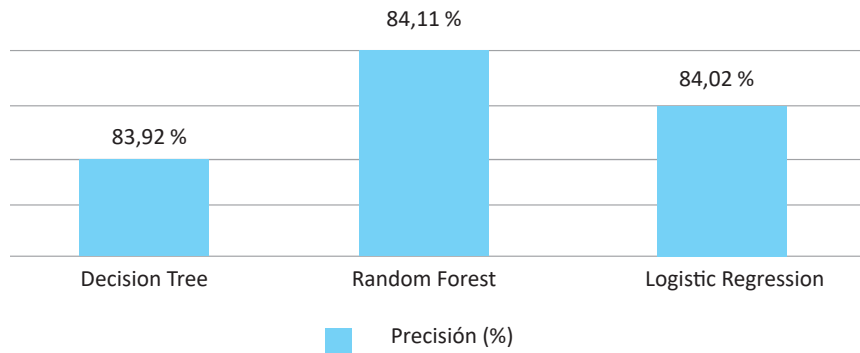
Cabe aclarar que se tuvieron en cuenta variables de aprobación o reprobación, ingreso familiar, escolaridad del padre, entre otros. El algoritmo Naive Bayes se desarrolló en la plataforma de programación HTML5 y PHP. Para probar la exactitud estimada del modelo, se aplicó la predicción del rendimiento académico de setenta y un estudiantes del mismo curso y se lo comparó con los datos reales. La exactitud del modelo es de 70,4225 %.

Widyaningsih *et al.* (2019) se enfocan en la utilización del aprendizaje semisupervisado para clasificar el desempeño de los estudiantes en el primer año de estudios en el Departamento de Matemáticas de la Universidad de Indonesia. La muestra consta de 140 estudiantes con veintisiete registros. La muestra se dividió en dos categorías: medio y alto. Como metodología para el conjunto de datos, se realizó preprocesamiento, agrupamiento de estudiantes, identificación de conglomerados, clasificación del estudiante y análisis de resultados. Se hizo una combinación del algoritmo cluster *K-Means* y el clasificador Naive Bayes para el modelo de estudio. El desempeño del modelo es medido arrojando métricas de precisión (96 %), sensibilidad (92,86 %) y especificidad promedio del 96 %.

Nieto *et al.* (2019) implementan tres algoritmos de clasificación de aprendizaje supervisado a estudiantes de ingeniería de pregrado en América del Sur. El objetivo es predecir las tasas de graduación y el número de graduados y no graduados. Los algoritmos aplicados son árbol de decisión, regresión logística y bosque aleatorio. Se realizó la caracterización, la limpieza y la preparación de datos. Posteriormente se hace contextualización de los algoritmos y finalmente se presentan las herramientas y métricas utilizadas. El periodo analizado está comprendido entre el 2004 al 2014. La muestra es de 6100 estudiantes de Ingeniería Electrónica. Se usaron un 70 % de los datos para el entrenamiento, y un 30 % para la predicción. Para el procesamiento de datos y los algoritmos de *Machine Learning* se utilizó la plataforma

analítica KNIME 3.4.0. En la figura 55 se puede observar el porcentaje de precisión de cada algoritmo: el mejor fue el bosque aleatorio, con un 84,11 %.

**Figura 55.** Comparación de la precisión de los algoritmos



**Fuente:** Nieto *et al.* (2019).

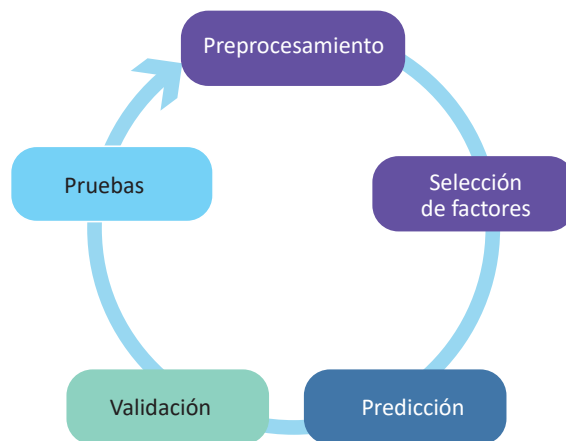
Otálora (2019) emplea métodos de minería de datos y aprendizaje automático para analizar información académica de programas curriculares de pregrado, en un periodo del 2012 al 2017. Se identificaron factores previos al ingreso del estudiante y de la secuencia de inscripción de asignaturas. Además, se construyeron diversos modelos de clasificación y predicción como árboles de decisión y clasificadores bayesianos. Algunas de las variables consideradas son el puntaje de admisión en el componente de matemáticas, el desempeño de la nivelación, la carga de semestre en créditos, la facultad a la que pertenece el estudiante y el desempeño académico de los estudiantes con el docente que imparte la asignatura. Al momento de calcular el modelo del aprendizaje supervisado se utiliza la herramienta matriz de confusión, de la cual se derivan métricas como exactitud, precisión, exhaustividad y el promedio de la precisión. En promedio, los árboles de decisión tienen un 65,61 % de efectividad, y son el mejor clasificador para el caso de estudio en cuestión. Los clasificadores bayesianos tienen, en promedio, una exactitud del 62 %.

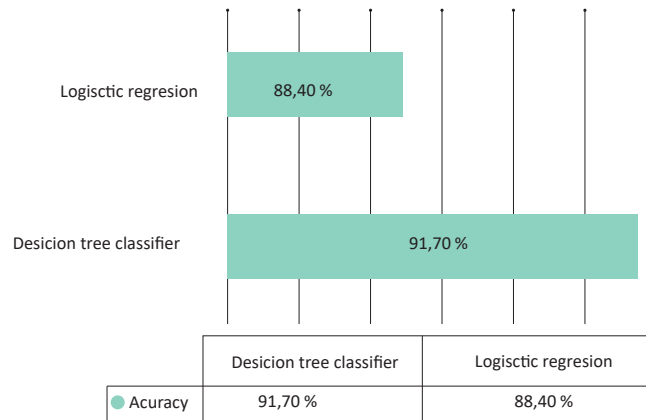
Istvan y Lasagna (2019) desarrollan un modelo de predicción de deserción a partir de una base de datos de gestión académica que contiene atributos personales y académicos: materias cursadas, libres, aprobadas, notas de parciales, finales, año de ingreso, asistencias, encuestas, entre otros. La

metodología de minería de datos para este proyecto es el proceso KDD, que comprende cinco fases: selección de datos, preprocesamiento de datos, transformación de datos, minería de datos e interpretación y evaluación de resultados. Para el estudio en particular, se utiliza la técnica de árboles de decisión en la herramienta de datos Rapidminer y el algoritmo W-J48 en el componente WEKA.

Para poder predecir la deserción estudiantil, Albán (2019) se propuso hacer una revisión integral sistemática de la literatura sobre predicción de deserción estudiantil universitaria, determinar nuevos factores de deserción, diseñar modelos de predicción por medio de técnicas de minería de datos y construir un modelo híbrido. Inicialmente determinó los factores tecnológicos para la predicción de la deserción estudiantil de las universidades de carreras de ingeniería de universidades públicas de Ecuador. Su investigación se basa en el aprendizaje automático y la aplicación de técnicas de regresión logística y árbol de decisión. Además, establece la metodología presentada en la figura 56a. Se definieron variables que influyen en el abandono académico debido al uso inadecuado. Se realizaron diez pruebas a cada algoritmo, y para medir el desempeño se utilizó la precisión. El método que presenta mayor tasa de precisión es el árbol de decisión, con un 91,7%. La precisión se muestra en la figura 56b. Se concluyó que el factor con mayor influencia es el uso del internet sin fines académicos.

**Figuras 56a y 56b.** Proceso de predicción de deserción universitaria. Precisión de modelos de predicción

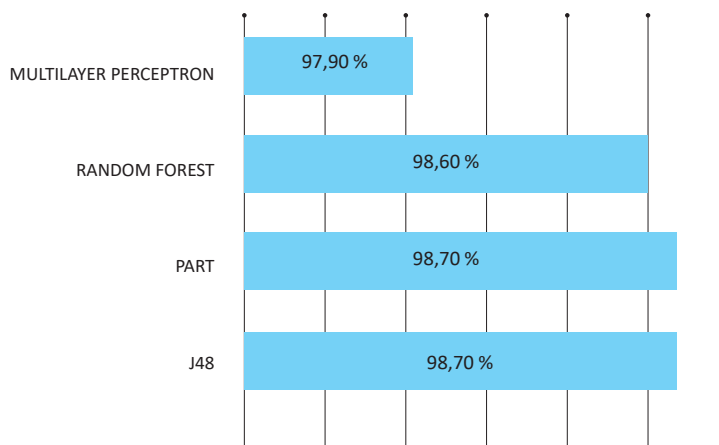




Fuente: Albán (2019).

Uno de los algoritmos usados en esta investigación fue el árbol de decisión. Los resultados de los experimentos arrojaron una tasa de precisión de la predicción del 97,75 % para el entrenamiento, y de 98,71 % para la muestra de prueba. Se identificaron variables muy influyentes: la adicción a las redes sociales, la adicción al apego emocional, el estado civil y la edad como las de mayor influencia para la predicción. También se usaron metaclasificadores para predecir la deserción estudiantil universitaria, como multilayer perceptron, bosque aleatorio, PART y J48. Los resultados muestran que los algoritmos PART tienen la predicción más precisa y medida (figura 57).

**Figura 57.** Precisión de los modelos de predicción



Fuente: Albán (2019).

Zarate (2019) las cuales incluyen características personales (género, edad, por su parte, estudia los factores que afectan el rendimiento académico de estudiantes de la Universidad de San Agustín, y tiene en cuenta para ello características de género, información de preingreso y desempeño en las asignaturas. Se pretende implementar una aplicación que le dé información al cuerpo directivo para identificar estudiantes que puedan estar en riesgo de abandonar la carrera universitaria. Para determinar los factores influyentes, se utilizan modelos de clasificación generados por algoritmos: redes neuronales, ID3 y C4.5. Luego, se compara cada uno de los modelos con métricas de precisión, exactitud y sensibilidad. La muestra consta de 970 estudiantes de la Escuela Profesional de Ingeniería de Sistemas con un intervalo de confianza de 90 %. Se determinó que el modelo con mejores métricas es el árbol de decisión C4.5, con una exactitud del 83,18 %, una precisión del 87,77 % y una sensibilidad del 90,80 %.

Heredia *et al.* (2019) tienen como precedente de su trabajo los altos niveles de deserción en Uruguay. Desarrollaron, a partir del procesamiento de datos, herramientas que monitorean la actividad estudiantil y permiten diagnosticar la situación académica del alumno. El objetivo principal es construir una plataforma denominada *Fin Analytics*, que exporte los datos de Moodle y analice información del entorno virtual de aprendizaje de un curso de la Facultad de Ingeniería. Para la plataforma se utilizó Git, además de la investigación de herramientas como RapidMiner, KNIME, Orange y Python, que sirvieron para recoger funcionalidades que iban a implementarse en la plataforma virtual. Para el análisis de datos, se obtuvieron muestras de bases de datos oficiales y encuestas demográficas de un total de 240 estudiantes. En cuanto a la aplicación del aprendizaje automático, se usó el árbol de decisión en diferentes modelos y se obtuvieron resultados de exactitud por encima del 74 %. Otras métricas como recall y la técnica precision [en inglés] tienen valores cercanos al 70 % en promedio, respectivamente.

Loja (2019) aplicó algoritmos para el desarrollo de modelos de clasificación y predicción con la metodología CRISP-DM. Esta metodología comprende etapas de entendimiento del negocio (Universidad de Cuenca), entendimiento de los datos, preparación y transformación de datos. Los modelos a elección son de clasificación y clusterización; se utilizan para la



construcción de las herramientas de Python y librerías como pandas, sklearn, matplotlib. También se usó la herramienta WEKA. Se hacen dos experimentos: el primero concierne a la investigación realizada, y su objetivo es predecir la deserción a partir de atributos académicos y socioeconómicos. En este experimento se preparan y se transforman los datos, y debe decirse que se obtuvieron en total 43 313 registros de estudiantes. Para la construcción del modelo se emplearon ocho técnicas: K-vecinos más cercanos, clasificación de vectores de soporte, árbol de decisión, bosque aleatorio, perceptrón multicapa, Naive Bayes, ada boost y análisis cuadrático discriminante. Para la validación cruzada del modelo, se usó la métrica de exactitud con diez subconjuntos. Primero, se ejecuta la clasificación de los ocho algoritmos iniciales. Como resultado, los tres algoritmos con mejor desempeño son bosque aleatorio, K-vecinos más cercano y perceptrón multicapa. Posteriormente se ejecuta el proceso de optimización de parámetros para obtener el mejor posible y luego aplicar el algoritmo de redes neuronales y la validación cruzada. Para las variables totales, el mejor algoritmo es bosque aleatorio, con una exactitud media del 73,2%; y para el caso de la evaluación exclusiva de las académicas, el algoritmo escogido también es bosque aleatorio, con una mejoría en la exactitud del 75,6%.

Zambrano (2019) Genera un modelo predictivo de retención-deserción de estudiantes de Negocios Internacionales de la Facultad de Ciencias Económicas y Administrativas, en un periodo de tiempo entre 2016 y 2018. Como metodología de minería de datos se incluyeron las fases de integración y selección, exploración y procesamiento, limpieza, transformación y reducción de datos, minería de datos, evaluación e interpretación de resultados y difusión, y uso de modelos. Los modelos escogidos son árbol de decisión (C4.5 y J48) y Naive Bayes compilados en la herramienta KNIME. El mejor algoritmo es el árbol de decisión C4.5 con un 90% de exactitud aproximadamente.

A través del uso de algoritmos de *Machine Learning* para identificar a los estudiantes de alta probabilidad de tener un bajo rendimiento académico en alguno de los 13 cursos del Programa de Estudios de la Universidad Ricardo Palma en Perú, Vega (2019) implementa, analiza y compara tres algoritmos: redes neuronales artificiales (RNA), gradient boosting machine (GBM) y

XGBoosting. Con estos algoritmos pretende encontrar el rendimiento académico por medio del pronóstico de la cantidad de estudiantes aprobados y desaprobados. Para realizar la aplicación de los algoritmos, se utilizó la metodología CRISP-DM que comprende una secuencia de seis etapas. Los datos son correspondientes a las notas de los alumnos matriculados desde el ciclo 2015-1 hasta el ciclo 2019-0, y en total se obtuvieron 9118 datos de alumnos y 574 283 calificaciones. Los procedimientos de análisis se construyeron en los *softwares* R, Python y MS Excel. En el experimento se tuvo en cuenta el indicador de precisión. Según los resultados, la mejor técnica de modelado fue XGBoosting. Es importante aclarar que, para la predicción del rendimiento académico de cada uno de los cursos, se tiene en cuenta el modelo XGBoosting en once ocasiones, y gradient boosting machine (GBM) y red neuronal artificial (RNA) en una ocasión.

Forero *et al.* (2019) presentan un caso de estudio en el que identifican el riesgo de deserción de estudiantes de Ingeniería Industrial de la Universidad Distrital Francisco José de Caldas, en el periodo entre 2003-1 y 2018-1. Para este estudio, aplican técnicas de aprendizaje automático en el *software* Azure Machine Learning Studio. Según la información proporcionada por la Universidad, se obtuvo una muestra de 3201 datos de variables académicas y sociales. Se aplicaron en total ocho algoritmos y se implementaron diferentes métricas de evaluación. En la tabla 38 se muestran los resultados de cada una de las pruebas realizadas, y se identifica el algoritmo de aprendizaje adecuado para la naturaleza de los datos: boosting decision tree, con una precisión del 90,3 %.

**Tabla 38.** Comparación de algoritmos de aprendizaje automático

| Algorithm                           | TP    | TN    | FP    | FN    | A     | P     | R     |
|-------------------------------------|-------|-------|-------|-------|-------|-------|-------|
| Averaged Perceptron                 | 0,879 | 0,955 | 0,121 | 0,045 | 0,919 | 0,879 | 0,947 |
| Bayes point machine                 | 0,852 | 0,963 | 0,148 | 0,037 | 0,908 | 0,852 | 0,958 |
| Boosted decision tree               | 0,903 | 0,965 | 0,097 | 0,035 | 0,936 | 0,903 | 0,958 |
| Decision forest                     | 0,899 | 0,956 | 0,101 | 0,044 | 0,93  | 0,889 | 0,947 |
| Decision jungle                     | 0,891 | 0,982 | 0,109 | 0,018 | 0,938 | 0,891 | 0,979 |
| Locally Deep support vector machine | 0,891 | 0,982 | 0,109 | 0,018 | 0,938 | 0,891 | 0,979 |

| Algorithm           | TP    | TN    | FP    | FN    | A     | P    | R     |
|---------------------|-------|-------|-------|-------|-------|------|-------|
| Logistic Regression | 0,850 | 0,963 | 0,150 | 0,037 | 0,906 | 0,85 | 0,958 |
| Neural network      | 0,870 | 0,975 | 0,130 | 0,025 | 0,923 | 0,87 | 0,972 |

Fuente: Forero *et al.* (2019).

Masabanda y Zapata (2019) proponen un modelo para determinar factores de deserción estudiantil por medio del uso de técnicas de minería de datos. La muestra se obtuvo al aplicar una encuesta a 1457 estudiantes de la Facultad de Ciencias de la Ingeniería y Aplicadas de la carrera de Ingeniería Eléctrica, Sistemas de Información, Electromecánica e Industrial. La metodología aplicada es *Knowledge Discovery in Databases* (KDD), que comprende etapas de selección y preprocesamiento, transformación, extracción e interpretación, y evaluación. Se aplicaron las técnicas J48, *random forest* y *Secuencial Minimal Optimization* (SMO). Estos algoritmos se usaron en conjunto en los *softwares* SPSS, Eviews y WEKA. Como resultado, se tuvo una predicción de deserción del 92 %. Se encontró que las variables más influyentes en el abandono estudiantil fueron conducta en el aula de clases, *bullying*, motivación del docente-alumno, limitado conocimiento de la asignatura, adicción de las redes sociales, estado emocional, conocimiento adquirido en los cursos de nivelación, formación académica, sistema de ingreso a la universidad, problemas familiares, residencia y expectativas respecto a la carrera seleccionada.

Ajibade *et al.* (2019) Educational Data Mining (EDM) proponen un modelo nuevo para la predicción del rendimiento de estudiantes con base en métodos de minería de datos. Los datos tenían que ver con características de comportamiento y selección secuencial. Este modelo utiliza clasificadores como máquina de vectores de soporte (SVM), K-vecino más cercano (KNN) y árbol de decisión. Además, se aplicaron métodos de ensamble como *bagging*, *boosting* y *random forest*. Para el modelo, se utiliza un conjunto de datos de 500 estudiantes con dieciséis características diferentes: demográficas, académicas, personales y conductuales. Después del entrenamiento del modelo, se hace la validación cruzada para definitivamente tener los resultados de evaluación según la prueba. Estos datos se presentan en la tabla 39, y el mejor algoritmo es el árbol de decisión, con 91 % de precisión.

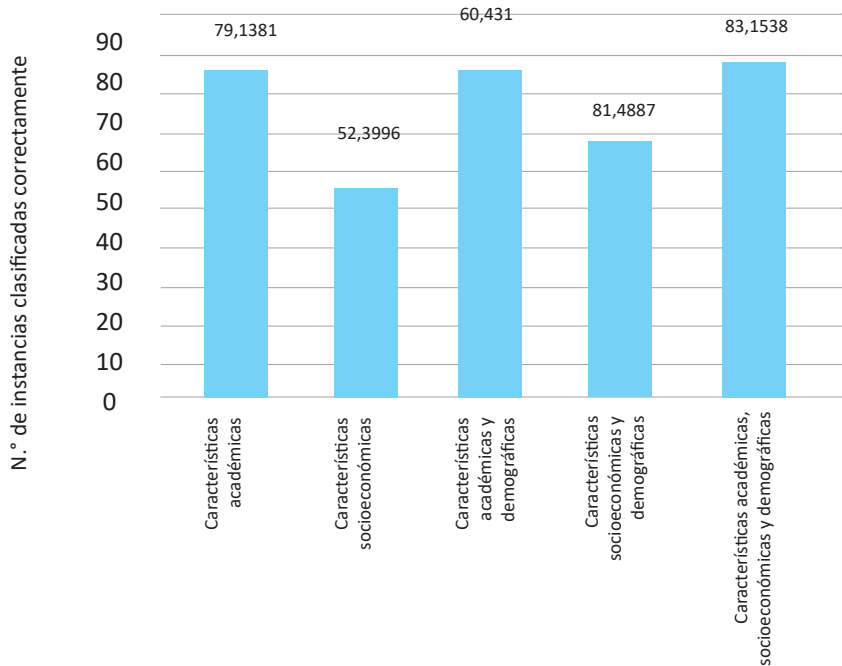
**Tabla 39.** Resultado de validación de los métodos de clasificación

| Evaluation Measure | Testing Results |      |      | Validation Results |      |      |
|--------------------|-----------------|------|------|--------------------|------|------|
| Classifier         | DT              | KNN  | SVM  | DT                 | KNN  | SVM  |
| Accuracy           | 0,87            | 0,82 | 0,86 | 0,91               | 0,85 | 0,88 |
| Precision          | 0,85            | 0,80 | 0,81 | 0,87               | 0,83 | 0,84 |
| Recall             | 0,86            | 0,79 | 0,81 | 0,89               | 0,82 | 0,84 |
| F-Measure          | 0,86            | 0,69 | 0,75 | 0,89               | 0,75 | 0,82 |

Fuente: Ajibade *et al.* (2019).

Jalota y Agrawal (2019) aplican varias técnicas de minería de datos para predecir el desempeño de los estudiantes. Utilizando un conjunto de datos de kalboard 360 en el *software* WEKA. Dentro del conjunto de datos hay dieciséis funciones y 480 registros de estudiantes que implican tres categorías principales: demográficas, educativas y psicológicas. Los clasificadores implementados fueron el algoritmo J48, máquina de vectores de soporte, Naive Bayes, bosque aleatorio y perceptrón multicapa. El perceptrón multicapa es el mejor algoritmo en comparación con los otros, con una precisión del 76,07%.

Aman *et al.* (2019) identifican características clave que influyen en el desempeño de los estudiantes para posteriormente desarrollar un modelo de predicción del desempeño antes de ser admitidos en un programa y decidir la continuación de clases. Se utiliza el método subjetivo para la identificación de características académicas y socioeconómicas; y después se adopta un modelo basado en árboles de decisión y el modelo de árboles logísticos. Para verificar el funcionamiento del modelo, se hace un entrenamiento y prueba a un conjunto de 1021 registros recopilados de la base de datos de exámenes de la Universidad de Peshawar, simulados en el entorno WEKA, y verificados con una validación cruzada de diez veces. Es importante aclarar que se realizan diferentes pruebas con combinaciones de características predictivas. En la figura 58 se presenta la comparación de la precisión: es evidente que la mayor precisión es lograda al combinar las tres categorías de características.

**Figura 58.** Comparación de categoríasFuente: Fazal *et al.* (2019).

Anderson *et al.* (2019) crearon un conjunto de modelos de aprendizaje automático para predecir la graduación de estudiantes universitarios de pregrado en un periodo de seis años. Utilizaron informes estatales y federales, y la evaluación universitaria. Además, aplicaron máquina de vectores de soporte, regresión logística, árboles de decisión y descenso de gradiente estocástico (SGD) a un conjunto de datos de 1400 estudiantes y dividieron las características establecidas en subcategorías: desempeño académico, información financiera, información previa a la admisión y actividades extracurriculares. Los modelos logran un alto rendimiento, y el GPA y las horas de crédito completadas son los predictores más importantes. El algoritmo de descenso de gradiente estocástico es el que arroja mejores métricas ( $AUC=0,824$ ) frente los valores de los otros algoritmos como árbol de decisión ( $AUC=0,786$ ), SVM ( $AUC=0,01$ ) y regresión logística ( $AUC=0,14$ ).

Burman y Som (2019) utilizan un algoritmo de máquina de vectores de soporte (SVM) para clasificar a los estudiantes en las categoría *alto*, *medio* y

*bajo* según sus puntajes académicos. Se utiliza kernel lineal y kernel de base radial para obtener mejores resultados.

El conjunto de datos se compone de registros de estudiantes recopilados de un cuestionario que se basa en parámetros psicológicos, estrategias de aprendizaje, enfoque del aprendizaje y nivel socioeconómico. El conjunto consta de 1000 registros con 29 construcciones no intelectuales de los estudiantes. Para la validación del modelo SVM con sus bifurcaciones, se usan métricas de precisión, sensibilidad y especificidad; estos resultados se presentan en la tabla 40: el SVM base radial es el mejor algoritmo debido a la precisión del 90,97 %.

**Tabla 40.** Resultados del modelo

| Statistics  | Linear Kernel  |        |         | Radial Basis   |        |         |
|-------------|----------------|--------|---------|----------------|--------|---------|
| Accuracy    | 0,6488         |        |         | 0,9097         |        |         |
| 95% CI      | 0,5918, 0,7029 |        |         | 0,8713, 0,9396 |        |         |
| p-value     | 1,58e-06       |        |         | 2,2e-16        |        |         |
|             | High           | Low    | Average | High           | Low    | Average |
| Sensitivity | 0,6536         | 1,0000 | 0,6667  | 0,9150         | 1,0000 | 0,9000  |
| Specificity | 0,6438         | 1,0000 | 0,6667  | 0,9041         | 1,0000 | 0,9182  |

**Fuente:** Burman y Som (2019).

En su trabajo, Das y Rodriguez-Marek (2019) buscan determinar si existe una correlación entre las clases preparatorias que toman los estudiantes de Ingeniería Eléctrica de la Eastern Washington University y sus GPA al momento de graduarse. Para ello, se entrenó un algoritmo de aprendizaje supervisado con variables predictoras previstas demográficas y académicas. La base de datos contiene registros de 279 estudiantes que se graduaron en el periodo entre 2007 y 2016. Los algoritmos aplicados son K-vecino más cercano (KNN) y Naive Bayes. El mejor de estos dos, al compararlos, es KNN; su precisión fue mayor y puntuó 70 % aproximadamente. Las variables que pueden ser utilizadas para definir el desempeño futuro de una estudiante son las calificaciones en las secuencias de matemáticas, física uno, dos y tres.

De La Hoz *et al.* (2019) construye una metodología para clasificar y predecir las actitudes de los usuarios que interactúan en ambientes virtuales

de educación, según el desempeño de los estudiantes en los exámenes y la plataforma educativa. Se utilizaron herramientas de aprendizaje automático como clusterización, lógica difusa y K-vecino más cercano. Para la investigación, se tomó información de una base de datos de un total de 403 usuarios del Sistema de Aprendizaje Virtual Interactivo (SAVIO) de la Universidad Tecnológica de Bolívar. El análisis se desarrolló en R y el paquete *dendexted*. Como resultado, se encontró que el algoritmo KNN supera en un 7,1 % el desempeño del clasificador lineal y cuadrático.

Mohammadi *et al.* (2019) implementan diferentes algoritmos como K-vecino más cercano (KNN), Naive Bayes y árbol de decisión. Inicialmente se prepararon los datos recolectados de una encuesta realizada a 230 estudiantes de la Facultad de Ciencias de la Universidad de Kabul. Posteriormente, se seleccionaron y transformaron características con el *software* WEKA. Luego se implementó minería de los algoritmos nombrados anteriormente. El mejor resultado lo arroja el KNN, con un 54,64 % de exactitud, seguido del árbol de decisión con un 53,25 % de exactitud. Por último, el algoritmo Naive Bayes con un 46,16 % de exactitud.

Bhutto *et al.* (2020) presentan un modelo de predicción del rendimiento académico que utiliza aprendizaje automático supervisado con la aplicación de los algoritmos máquina de vectores de soporte (SVM) y regresión logística. El conjunto de datos utilizados se adquirió del sistema de aprendizaje electrónico Kalboard 360. Se recolectaron quinientos registros y dieciséis características demográficas y académicas. El mejor resultado alcanzado por los modelos en el *software* WEKA es para el SVM, con un 79 % de exactitud frente a un 73,4 % de exactitud para la regresión logística.

Viloria *et al.* (2020) usan métodos que permitan explorar y comprender a los estudiantes y a los entornos de aprendizaje. La metodología es la recopilación de datos. Se preprocesaron 1268 datos de estudiantes con 89 atributos. Se aplicaron los algoritmos árbol de decisión J48, simplecart, ADTree, random tree y REPTree. Finalmente, se realizó una validación cruzada de los modelos con los mejores atributos para determinar el algoritmo más adecuado. En la tabla 41 se muestran los resultados: simplecart es el mejor por su precisión del 97,8 %.



**Tabla 41.** Validación cruzada con los mejores atributos

| Algorithm  | Success<br>Approved<br>(%) | Success<br>Suspended<br>(%) | Accuracy<br>Total (%) | Geometric<br>Mean (%) |
|------------|----------------------------|-----------------------------|-----------------------|-----------------------|
| JRip       | 97,3                       | 94,2                        | 96,0                  | 93,7                  |
| NNge       | 97,3                       | 72,7                        | 95,8                  | 83,0                  |
| OneR       | 95,1                       | 71,0                        | 93,7                  | 80,5                  |
| Prism      | 98,5                       | 38,7                        | 95,5                  | 54,1                  |
| Ridor      | 95,9                       | 57,4                        | 93,4                  | 73,0                  |
| ADTree     | 97,2                       | 82,7                        | 96,5                  | 89,2                  |
| J48        | 94,7                       | 81,2                        | 95,3                  | 87,1                  |
| RandomTree | 94,5                       | 69,3                        | 94,0                  | 81,3                  |
| REPTree    | 96,4                       | 66,0                        | 92,6                  | 78,1                  |
| SimpleCart | 96,3                       | 91,4                        | 97,8                  | 94,5                  |

**Fuente:** Vilorio *et al.* (2020)

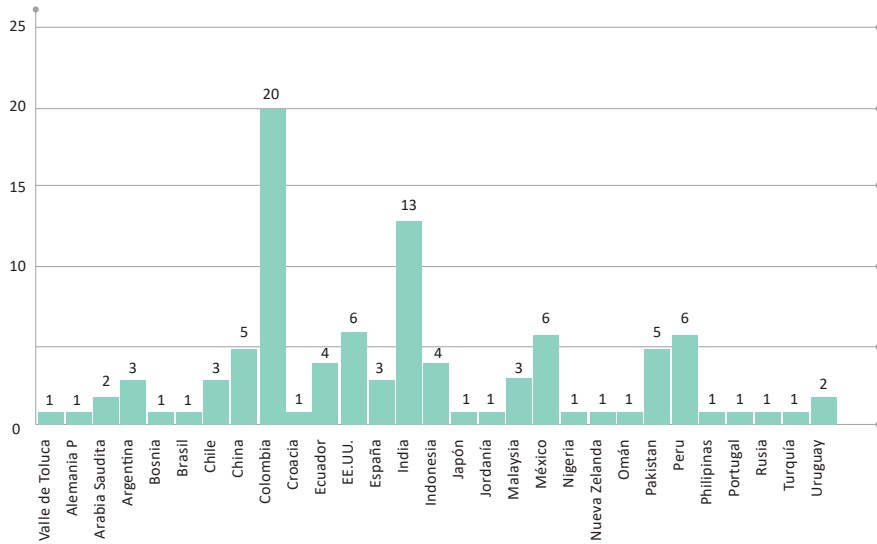
## 8.4. Análisis de las producciones académicas relacionadas con aprendizaje supervisado

Un aspecto que vale la pena señalar luego de la breve descripción y el análisis de cada uno de los artículos correspondientes al periodo 2010-2020 es la nacionalidad de las investigaciones. Es importante identificar cuáles son los países que están trabajando esta parte de la temática en publicaciones en revistas científicas de alto impacto. En la figura 59 se presenta el diagrama de barras de las investigaciones del sector educación y su país de publicación.

Al analizar la figura 60, se observa que, a lo largo de los últimos diez años, los trabajos e investigaciones referentes al aprendizaje supervisado se han desarrollado exponencialmente en el continente americano, en especial en países latinoamericanos. Es en estos países que la educación ha presentado falencias causadas por la pobreza, la falta de apoyo gubernamental y el acceso a herramientas tecnológicas. Esta técnica permite con facilidad identificar factores y variables influyentes que describen y predicen el comportamiento actual y futuro del sistema educativo. Colombia se destaca por ser uno de los líderes de este tipo de investigaciones con un total de 20 artículos en total.



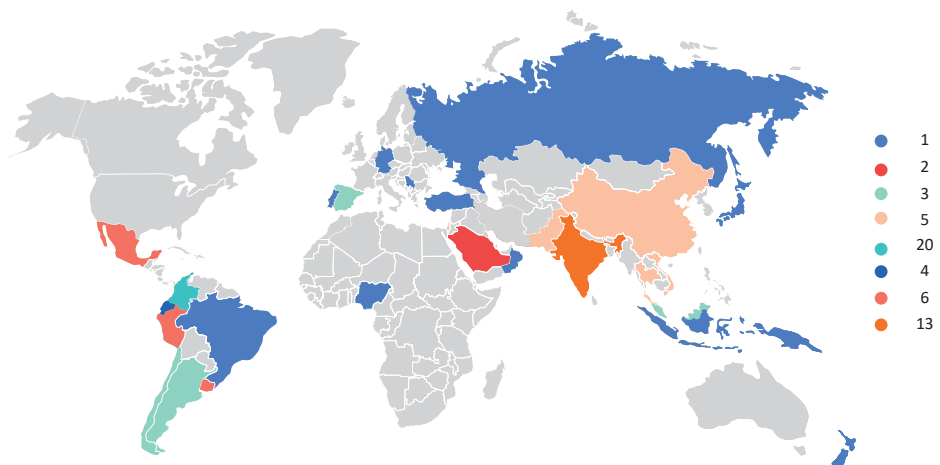
**Figura 59.** Investigación referencial y su representación en el mundo



Fuente: elaboración propia.

**Figura 60.** Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado

Artículos por país



Fuente: elaboración propia.

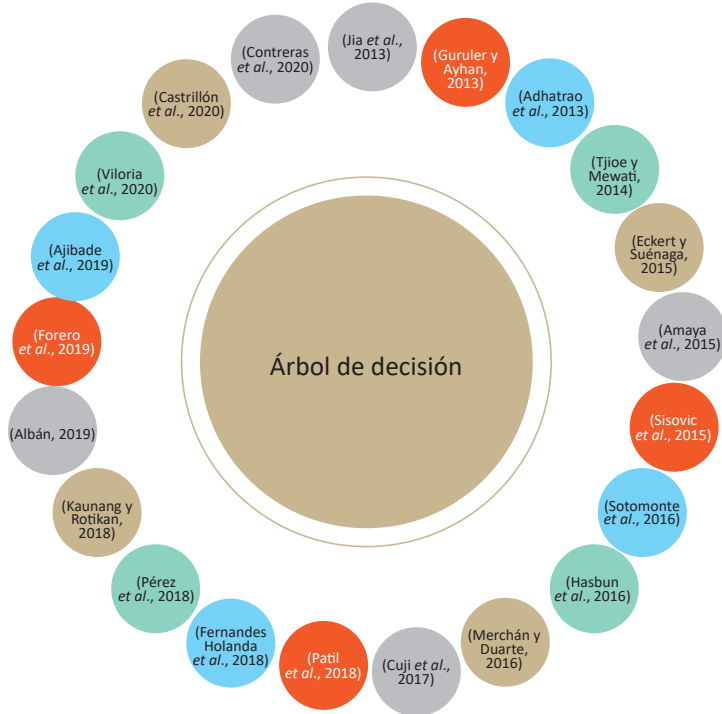


# Capítulo 9

## Aprendizaje supervisado: Algoritmos implementados

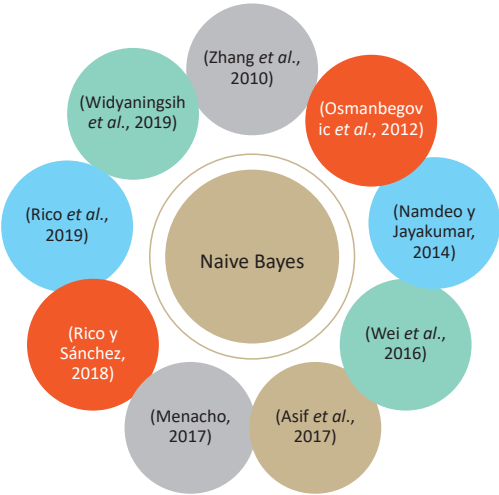
Este capítulo tiene el fin de profundizar en la aplicación de las diferentes técnicas que se revisaron en el estado del arte y que están relacionadas con el aprendizaje supervisado. Existen distintos algoritmos clasificados dentro del aprendizaje supervisado. A continuación, se presenta un conjunto de figuras que permiten identificar los algoritmos más utilizados, así como los diferentes documentos estudiados. Se señalan los nombres de los artículos que usaron el algoritmo en específico y el autor. Este ejercicio se realiza para conocer la eficiencia de cada algoritmo y así ser comparado con la aplicación propia objetivo de la investigación específica.

**Figura 61.** Investigaciones que tuvieron como mejor algoritmo al árbol de decisión



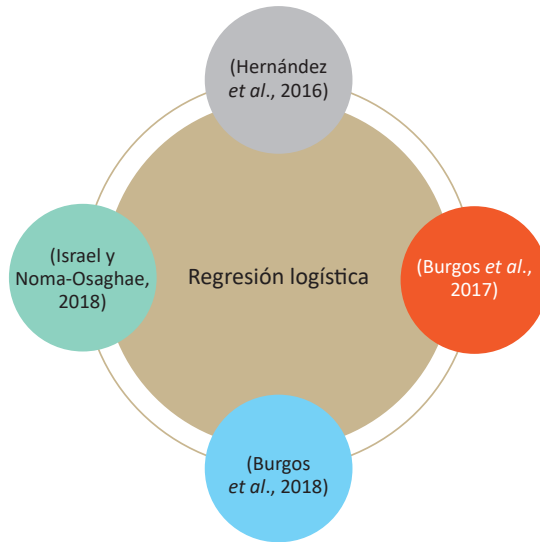
Fuente: elaboración propia.

**Figura 62.** Investigaciones que tuvieron como mejor algoritmo a Naive Bayes



Fuente: elaboración propia.

**Figura 63.** Investigaciones que tuvieron como mejor algoritmo a regresión logística



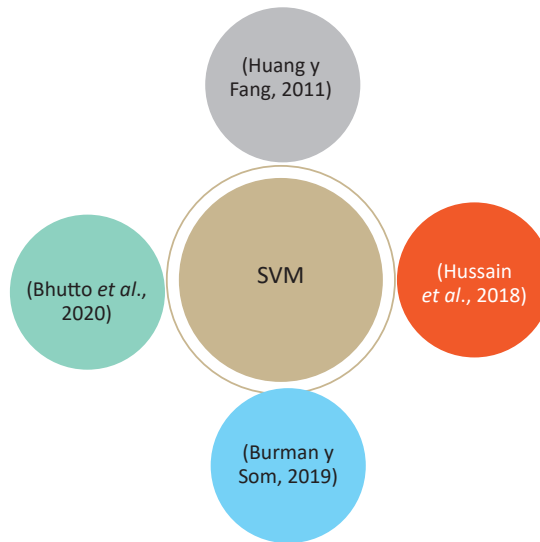
**Fuente:** elaboración propia.

**Figura 64.** Investigaciones que tuvieron como mejor algoritmo a K-vecino más cercano



**Fuente:** elaboración propia.

**Figura 65.** Investigaciones que tuvieron como mejor algoritmo a máquina de vectores de soporte (SVM)



**Fuente:** elaboración propia.

El aprendizaje supervisado se origina cuando cada una de las observaciones o muestras del conjunto de datos tiene una variable relacionada o un dato que indica lo que sucedió, lo que pasó; en otras palabras, las entradas están etiquetadas. Este tipo de aprendizaje se subdivide en clasificación y regresión. En clasificación, las salidas del sistema son finitas y discretas, y son interpretadas como la clase a la que pertenece cada una (García, 2014). Existen, dentro del campo, diversos algoritmos utilizados para realizar predicciones sobre valores de salida o clases: estos son clasificados según el tipo de variable de salida. Cuando la variable de salida es categórica, se denomina a dice que se tiene un algoritmo supervisado para tareas de clasificación. El otro grupo está constituido por algoritmos cuya variable de salida es numérica; se denominan algoritmos supervisados para tareas de regresión. A continuación, se detallan cada una de las técnicas de aprendizaje no supervisado, así como el uso de estas técnicas en cada una de las investigaciones consultadas en la última década.

## 9.1. K-vecinos más cercanos (KNN)

Es un algoritmo de aprendizaje basado en instancias. Almacena los ejemplos de entrenamiento de datos históricos. Si se requiere clasificar un nuevo objeto, se extraen los más similares y se usa esa agrupación para crear la respectiva clasificación. Los vecinos más cercanos a una instancia se obtienen, para el caso de los atributos continuos, con la distancia euclidiana sobre los  $n$  posibles atributos. El resultado de la clasificación por medio de este algoritmo puede ser discreto o continuo. En el caso discreto, el resultado de la clasificación es la clase más común de los K-vecinos (Valero *et al.*, 2010).

## 9.2. Naive Bayes

Es un algoritmo basado en técnicas de clasificación estadística que se agrupan bajo el teorema de Bayes. Se trata de una forma de calcular una probabilidad contando la frecuencia de valores y combinaciones de valores en los datos históricos (Namdeo y Jayakumar, 2014). El principio de clasificación bayesiano es utilizar una probabilidad previa de un objeto y calcular la posterior; luego se escoge la probabilidad posterior máxima como objeto clase (Wei *et al.*, 2016).

## 9.3. Máquina de vectores de soporte (SVM)

Las máquinas vectoriales de soporte (SVM) pertenecen a una familia de modelos lineales que logran una decisión de clasificación o regresión a partir del valor de la combinación lineal de características (Delen, 2010). Se basan en la teoría de aprendizaje estadístico que permite la resolución de problemas de clasificación y regresión de manera eficiente. El éxito de las máquinas de soporte vectorial radica en tres ventajas fundamentales:

- Tienen una sólida fundamentación matemática.
- Se basan en el concepto de minimización del riesgo estructural: al usarlos se minimiza la probabilidad de una clasificación errónea sobre nuevos ejemplos. Es particularmente importante cuando se dispone de pocos datos de entrenamiento.
- Su uso implica contar con potentes herramientas y algoritmos para hallar la solución de manera rápida y eficiente (García, 2014).

## 9.4. Árboles de decisión (*decision trees*)

Un árbol de decisión es un diagrama de flujo con estructura de árbol: los nodos internos representan validaciones sobre los atributos, las ramas son las salidas de las validaciones, y los “nodos hoja” las clases. El nodo en la parte superior del árbol se le conoce como nodo raíz. Para clasificar una instancia “desconocida”, se sigue el flujo del árbol hacia abajo, de acuerdo con los valores que tengan los atributos para cada nodo, y cuando se llega a un “nodo hoja”, la instancia se clasifica de acuerdo con la clase asignada por dicho nodo. Existen diversos métodos para la inducción de árboles de decisión (ID3, C4, C4.5, Bayesiano, CART, etc.), y cada uno de ellos ofrece diferentes capacidades. En general, dichos algoritmos son apropiados para solucionar *problemas de clasificación* (Amaya *et al.*, 2014). A continuación, se hará una breve descripción de los tres algoritmos más usados de árboles de decisión.

- ID3: La idea básica del algoritmo ID3 tiene su fundamento en la iteración. Un subconjunto del conjunto total de datos de entrenamiento, al cual se le conoce como ventana, es elegido de manera aleatoria para formar un árbol de decisión; este árbol clasifica de manera correcta todos los objetos que pertenecen a las ventanas. El resto de los objetos, dentro del conjunto, es clasificado con el árbol. Si el árbol da una respuesta correcta para todos estos objetos, entonces también es correcto para el conjunto total de datos de entrenamiento y termina el proceso (Amaya *et al.*, 2014).
- C4.5: Se trata de un algoritmo de aprendizaje basado en semejanzas. Los árboles de decisión son uno de los algoritmos más sencillos de implementar y a su vez de los más eficaces. Este algoritmo genera un árbol de decisión de forma recursiva al considerar el criterio de la mayor proporción de ganancia de información; es decir, elige al atributo que mejor clasifica a los datos (Valero *et al.*, 2010). El algoritmo forma parte de la familia de los *árboles inducidos de arriba hacia abajo* (top down induction of decision trees o TDIDT, por sus siglas en inglés). Estos árboles aprenden a partir de ejemplos preclasificados. Tanto el algoritmo ID3 como el C4.5 fueron propuestos por Ross Quinlan. El algoritmo C4.5 es una extensión del algoritmo ID3, el cual trabaja únicamente con valores discretos en



los atributos. En cambio, el algoritmo C4.5 permite trabajar con valores continuos para separar los posibles resultados en dos ramas: una para aquellos  $A_i \leq N$ , y otra para todos los  $A_i > N$ . De esta forma, C4.5 genera un árbol de decisión a partir de datos mediante particiones realizadas de manera recursiva (Amaya *et al.*, 2014).

- CART: El algoritmo CART de Leo Breiman realiza particiones binarias. El objetivo es que la media de cada rama sea diferente y, por tanto, discrimine con suficiente precisión un número adecuado de particiones para asignar a cada hoja un valor cercano a la media de los elementos que caen en ella. Este algoritmo genera árboles de fácil interpretación con resultados óptimos, lo que se considera una ventaja al crear modelos predictivos (Cuji *et al.*, 2017).

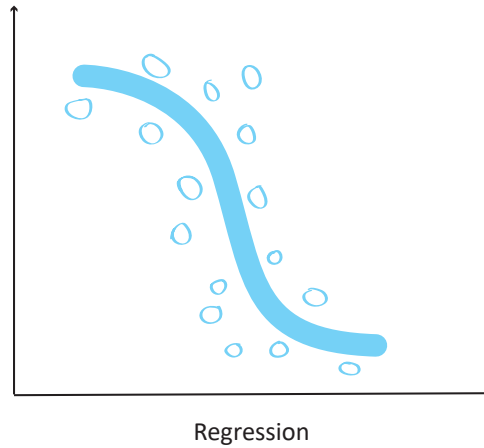
## 9.5. Regresión logística

Es un algoritmo de clasificación que se utiliza para predecir la probabilidad de una variable dependiente categórica. La regresión logística es un tipo especial de regresión que se utiliza para explicar y predecir una variable categórica binaria en función de varias variables independientes que a su vez pueden ser cuantitativas o cualitativas (García, 2014).

## 9.6. Algoritmos de regresión

Un algoritmo de regresión ajusta una línea o curva a un conjunto de objetos (datos). Para hacerlo, explora los atributos o características conocidas de antemano. La regresión es básicamente una clasificación que permite pronosticar un número en lugar de una categoría. A continuación, en la figura 66, se explica brevemente cómo funciona cada uno de los métodos más populares para tareas de regresión: regresiones lineales, polinomiales, Ridge y Lasso.

**Figura 66.** Algoritmos de regresión

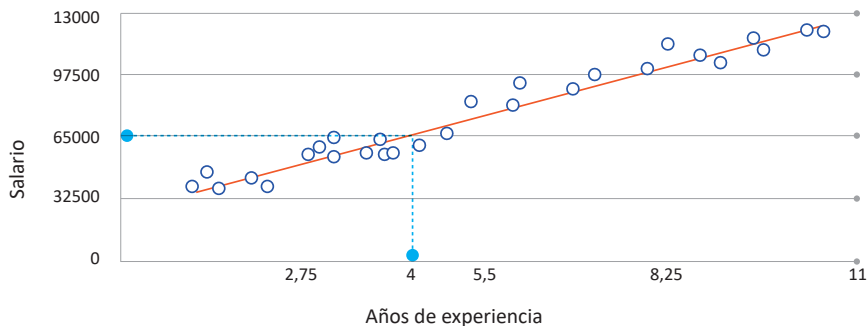


**Fuente:** elaboración propia.

## 9.7. Regresión lineal

La regresión lineal es un método para modelar una respuesta a partir de un conjunto de variables explicativas. Modela la relación entre unas variables de entrada (X) con una variable de salida (Y) de tipo numérico (figura 67).

**Figura 67.** Regresión lineal



**Fuente:** elaboración propia.

La regresión lineal utiliza el modelo de recopilación de datos con el fin de predecir los parámetros desconocidos de una relación lineal supuesta. Los parámetros que se aprenden se pueden usar para predecir el valor numérico de la variable dependiente en función de un nuevo conjunto de variables

independientes (Vladimir *et al.*, 2019). Un modelo de regresión lineal tiene como fin predecir una variable llamada variable dependiente (Y) a partir de la información de la otra variable, llamada variable independiente (x), como se muestra a continuación:

$$Y=b_0+b_1 x$$

En este caso, y son incógnitas que se pueden conocer a partir del análisis de regresión. La regresión lineal tiene como objetivo descubrir los coeficientes que describen la relación existente entre las variables de entrada (variable explicativa) y las de salida (variable objetivo), de manera que puedan ser utilizados para predecir nuevos datos que no formen parte del subconjunto de entrenamiento. Es decir, el modelo trata de encontrar la línea que mejor se ajusta al subconjunto de entrenamiento.

Como se tienen muchos datos, es necesario calcular diferentes propiedades estadísticas, y a partir de ellas poder determinar los coeficientes de la regresión lineal.

$$b_1 = \frac{\sum_{i=1}^n (x_i - \text{media}(x)) * (y_i - \text{media}(y))}{\sum_{i=1}^n ((x_i - \text{media}(x))^2)}$$

$$\sum_{i=1}^n ((x_i - \text{media}(x))^2)$$

$$b_0 = \text{media}(y) - b_1 * \text{media}(x)$$

La media y la varianza de las variables de entrada y salida a partir de los datos de entrenamiento.

$$\frac{\sum_{i=1}^n x_i}{\text{cantidad}(x)}$$

$$\text{varianza} = \sum_{i=1}^n (x_i - \text{media}(x))^2$$

La regresión lineal es un modelo en el que se usa una línea recta para modelar la relación entre los valores de entrada y salida. Sin embargo, cuando se tienen múltiples variables de entrada, el modelo es de cierta manera similar, solo que aparecerán más coeficientes  $B_n X$  en la ecuación. La predicción se realiza como una combinación de los valores de entrada para predecir el

valor de salida (y) se genera a partir de la ponderación de cada atributo de entrada (x) y el uso de un coeficiente (b).

$$y=b_0+ b_1 X_1+ b_2 X_2+\cdots b_n$$

Para hallar estos coeficientes existen diferentes métodos que pueden ser utilizados en las áreas de la estadística y el aprendizaje automático:

- Mínimos cuadrados ordinarios: Trabaja con propiedades estadísticas como media, desviación y correlación; se recomienda usar cuando existen pocos datos. Esto significa que, dada una línea de regresión a través de los datos, calculamos la distancia desde cada punto de datos hasta la línea de regresión, la elevamos al cuadrado y sumamos todos los errores al cuadrado. Esta es la cantidad que los mínimos cuadrados ordinarios buscan minimizar.
- Descenso de gradiente estocástico (*stochastic gradient descent*): Este método es un algoritmo de optimización utilizado para encontrar los valores de parámetros (coeficientes) de una función (f) que minimiza una función (costo). El objetivo es continuar probando diferentes valores para los coeficientes, evaluar su costo y seleccionar nuevos coeficientes que tengan un costo ligeramente mejor (más bajo). Se recomienda usarse cuando no se pueden usar métodos analíticos (con álgebra lineal), así como cuando se tiene un gran volumen de datos. La forma en que funciona este algoritmo de optimización tiene que ver con que cada instancia de entrenamiento se muestra al modelo de una en una. El modelo realiza una predicción para una instancia de entrenamiento, el error se calcula y el modelo se actualiza para reducir el error para la próxima predicción. Este proceso se repite para un número fijo de iteraciones.

$$\begin{aligned}\text{costo} &= f(\text{coeficientes}) \\ \text{costo} &= \text{evaluar}(f(\text{coeficientes}))\end{aligned}$$

Este procedimiento puede usarse para encontrar el conjunto de coeficientes en un modelo que da como resultado el error más pequeño para el modelo en los datos de entrenamiento. En cada iteración, los coeficientes (b) en el

lenguaje de aprendizaje automático se actualizan cuando se usa la siguiente ecuación:

$$b = b + \text{learning rate} * \text{error} * X$$

En este caso,  $b$  es el coeficiente o el peso que se está optimizando; *learning rate* es la tasa de aprendizaje que debe configurar (por ejemplo, 0,01); *error* es el error de predicción para el modelo en los datos de entrenamiento atribuidos al peso (es la diferencia entre la predicción realizada con los coeficientes candidatos y la salida esperada); y  $x$  es el valor de entrada. Hay un coeficiente para ponderar cada atributo de entrada, y estos se actualizan de manera consistente; por ejemplo:

$$b_1(t+1) = b_1(t) + \text{learning rate} * \text{error}(t) * X_1(t)$$

La diferencia entre usar un método conocido como descenso de gradiente y descenso de gradiente estocástico radica en que, en el segundo caso, la variación por medio del procedimiento descrito se ejecuta para la actualización de los coeficientes para cada instancia de entrenamiento, en lugar de al final realizar una actualización para un lote de instancias.

## 9.8. Regresión polinomial

Con el fin de mejorar la función hipotética que representa la relación entre las características y la variable de salida, es posible usar una regresión polinómica o polinomial; es decir, mejorar la función de hipótesis convirtiéndola en una función de raíz cuadrática, cúbica o de cualquier otra forma o grado. Por ejemplo, si nuestra función de hipótesis es:

$$Y = b_1 x_1 + b_2 x_2 + \dots + b_n x_n + c$$

entonces, podemos crear características adicionales basadas en  $x_1$  para obtener una función cuadrática

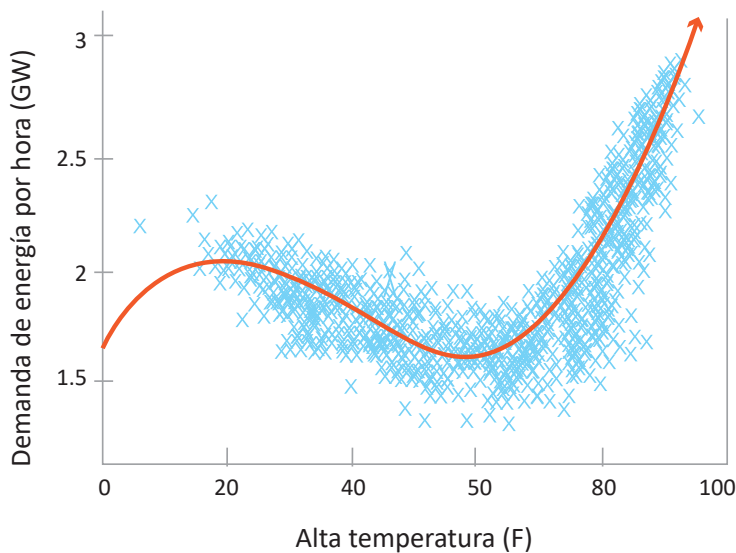
$$Y=b_0+b_1 x_1+b_2 x_2^2$$

o, tal vez, una función cúbica:

$$Y=b_0+b_1 x_1+b_2 x_2^2+b_3 x_3^2$$

Este algoritmo es usado cuando los datos provenientes no tienen una relación lineal. Entonces, se deben implementar polinomios de grado  $n$  para obtener el modelo (figura 68); es decir, mediante ensayo y error, se debe determinar cuál es el grado del polinomio (curva) que se ajusta más a los datos. No siempre aumentar el grado del polinomio no hará que la curva se ajuste mejor al modelo.

**Figura 68.** Ejemplo de regresión polinomial para conjunto de datos



**Fuente:** González (2020).

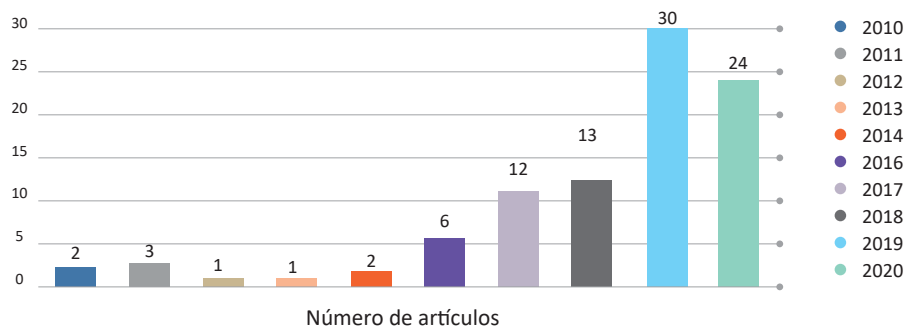
## Capítulo 10

# Redes Neuronales: factores y variables que influyen en el rendimiento académico

Antes de proceder a establecer las variables y factores que se han tenido en cuenta en los trabajos relacionados con rendimiento académico y que usan algoritmos de aprendizaje profundo, daremos un vistazo a la información recopilada.

A partir de la información recopilada y filtrada por medio de los criterios mencionados para el periodo 2010-2020, y en cuanto al uso del aprendizaje no supervisado para determinar rendimiento académico de los estudiantes en educación superior, se pudo generar, como se muestra la figura 69, un panorama general de sus usos. Así mismo, la cantidad de artículos encontrados al aplicar las ecuaciones de búsqueda es de 94.

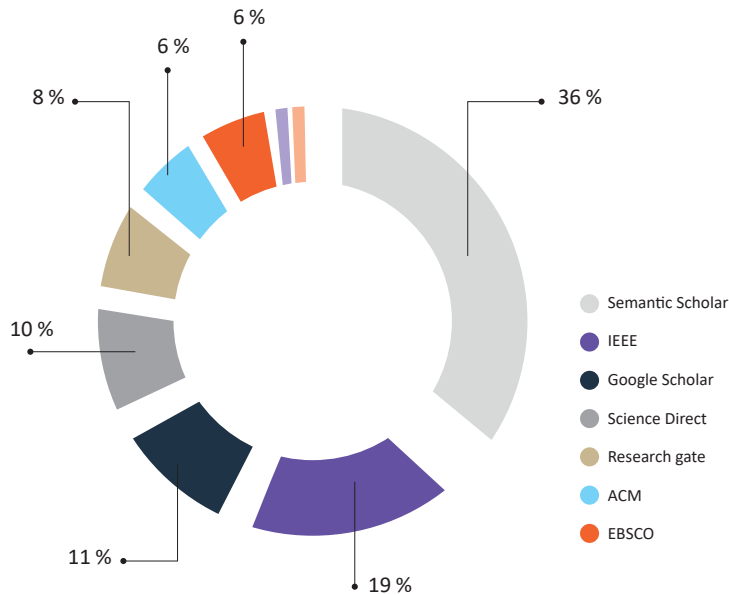
**Figura 69.** Agrupación de referencias bibliográficas por año



**Fuente:** elaboración propia.

Se realizó un análisis frente a cuáles eran los diferentes motores de búsqueda y bases de datos que más suministraron información: luego de haber filtrado y aplicado criterios de exclusión, se obtienen los resultados mostrados en la figura 70

**Figura 70.** Cantidad de artículos encontrados en cada motor de búsqueda



**Fuente:** elaboración propia.

Uno de los objetivos que comúnmente tienen las instituciones educativas es el éxito académico. Sin embargo, si analizamos en detalle, ¿qué es el éxito académico?, ¿cómo se mide?, ¿cómo se puede asegurar que un estudiante ha logrado obtener o alcanzar el éxito académico? Ciertamente es un concepto subjetivo, pero en algunos casos surgen definiciones que pueden llegar a ser cercanos a lo general, como la siguiente, que define al éxito académico diciendo que consiste en lograr participación en actividades, satisfacción, adquisición de conocimientos, habilidades y competencias deseados, y define al rendimiento académico como el grado en el que un estudiante obtiene buenas o malas calificaciones (Alyahyan y Düşteğör, 2020).

Por otro lado, el éxito académico y la deserción tienen relación: si hay deserción no hay éxito académico porque no se lograron las competencias, conocimientos y habilidades deseadas. Por el contrario, si el rendimiento



académico es bajo, no se logra la adquisición de ellas. La deserción de estudiantes en una institución educativa está relacionado directamente con el rendimiento académico (Will *et al.*, 2019).

Gracias al crecimiento del uso de las TIC, se han diseñado herramientas analíticas que pretenden, mediante el uso de datos generados en entornos académicos y de del análisis de estos datos, mejorar el proceso educativo. Así pues, aparecen herramientas como Academic Analytics (AA), Learning Analytics (LA) y Educational *Data Mining* (EDM).

Para tratar la predicción de estos dos indicadores de interés en el campo educativo (deserción y rendimiento) en los nuevos paradigmas tecnológicos, una de las opciones de uso son las redes neuronales artificiales (ANN, por sus siglas en inglés) y los árboles de decisión (DT, por sus siglas en inglés). La ventaja de las redes neuronales es que pueden identificar patrones no obvios en la entrada (Prenkaj *et al.*, 2020), aunque según Rastrollo-Guerrero *et al.* (2020) las técnicas más usadas para predecir el rendimiento académico son las del aprendizaje supervisado, en especial las máquinas de vectores de soporte (SVM, por sus siglas en inglés) por su fiabilidad y precisión de predicción.

En este capítulo, se presenta de forma detallada un conglomerado de variables que han sido trabajadas por diferentes autores de múltiples referencias bibliográficas y que encontramos en las bases de datos mencionadas anteriormente. Estas variables se trabajan con el propósito de predecir el rendimiento y la deserción académica de estudiantes en educación superior. Las investigaciones estudiadas por lo general construyen modelos de aprendizaje automático mediante redes neuronales. Las variables se clasifican en seis grupos principales.

En las últimas dos décadas, las instituciones de educación superior han evolucionado con la implementación de sistemas de gestión académica (SGA). Esta utilización de nuevo *software* y *hardware* implica el almacenamiento masivo de datos de los educandos. Se trata de datos tanto tomados al momento de la inscripción a un sistema educativo, como de datos que se toman durante el proceso de aprendizaje. Estos extensos volúmenes de datos rara vez son aprovechados, y simplemente quedan almacenados como registros. Entonces, se vuelve imprescindible hacer uso de esas bases de datos

generadas en el ámbito académico gracias a las nuevas tecnologías para tomar medidas o diseñar herramientas que permitan mejorar el desempeño de esta labor de instrucción y mejoren la experiencia para todos sus actores.

Aquí surge una duda: ¿qué es el desempeño del aprendizaje?, y ¿cómo se mide? De nuevo, se trata de conceptos subjetivos, y su definición depende de los métodos de enseñanza del docente, las condiciones administrativas de las instituciones educativas, los objetivos del estudiante y demás. Por esta razón, no hay una teoría definitiva acerca del desempeño académico y su medición. Diversos autores muestran diferentes teorías al respecto: por ejemplo, se señala que el rendimiento académico es una variable multifactorial y que, a pesar de ello varios de los estudios del rendimiento solo incluyen factores socioeconómicos y personales (Reyes *et al.*, 2018). En Zambrano Matamala *et al.* (2011) se usa un esquema de siete dimensiones de análisis para lograr llegar a una definición:

- Alumno: Todo lo referente a datos personales de los estudiantes y su estado.
- Asignatura: Datos de asignaturas vistas y condición para ingresar a la universidad.
- Región: El origen de los estudiantes.
- Oportunidad: Referente a la posibilidad de cursar materias.
- Tiempo de avance: Relación de tiempo en la carrera vs semestres cursados.
- Zona geográfica: Lugar de residencia.
- Cohorte: semestre de cada estudiante.

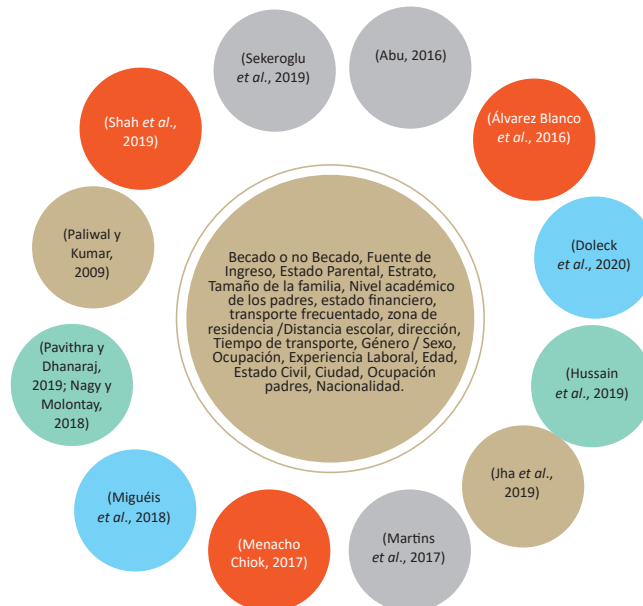
Por lo general, la forma de medir el rendimiento se basa en estas siete dimensiones. Sin embargo, al no existir una teoría consensuada, las variables dependen más de la información disponible. Por ejemplo, a algunas de estas variables se les dará mayor peso si el estudio se realiza en un entorno virtual. En un entorno así, se tendrán en cuenta variables como el número de clics, el tiempo en la plataforma, entre otros para entornos virtuales de aprendizaje (EVA) (Cui *et al.*, 2020).

Así mismo, Alyahyan y Düşteğör (2020) realizan una clasificación de factores que podrían predecir el rendimiento académico en cinco grupos principales: actividades estudiantiles *E-Learning*, factor psicológico, factor socioeconómico, factor académico y factor entorno estudiantil. Para poder realizar la implementación del modelo (trabajo futuro) y así ofrecer una predicción más acertada sobre el rendimiento académico, se decide agrupar las variables de acuerdo con los diversos factores que afectan al estudiante (psicológico, académico, socioeconómico, etc.).

### 10.1. Factor demográfico, económico y social

Este grupo de variables señalan datos con información demográfica, situación económica, familia, entre otras. En la figura 71 se señalan variables demográficas, económicas y sociales que se agruparon, ya que no son variables académicas, pero se cree que afectan en gran medida el rendimiento académico. Las variables de este grupo que se usaron con mayor frecuencia en los resultados obtenidos en la revisión del estado del arte son el género, la zona de residencia y la edad.

**Figura 71.** Nube de palabras variables demográficas, económicas y sociales

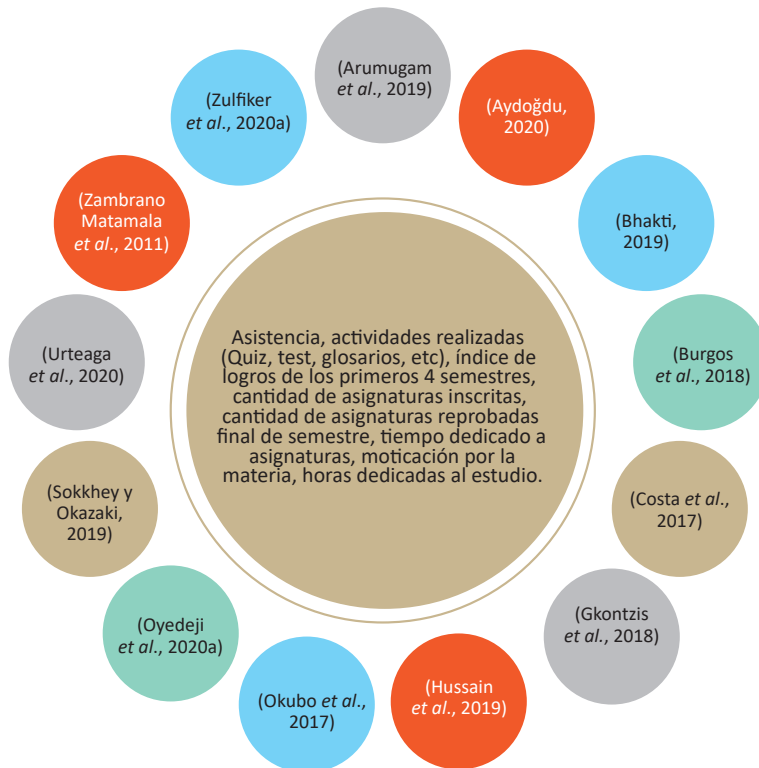


**Fuente:** elaboración propia basada en autores consultados.

## 10.2. Factor psicológico

Estas variables son las que hacen referencia a la posible motivación al realizar las actividades de aprendizaje por parte del estudiante, el tiempo que le dedica al estudio y los logros obtenidos. La variable predominante en este grupo es la asistencia a clases, como se muestra en la figura 72.

**Figura 72.** Investigaciones que consideran variables del Factor psicológico



**Fuente:** elaboración propia.

## 10.3. Factor entorno académico

Este factor engloba las variables que dan información general sobre todos los estudiantes en cuanto integrantes de la institución educativa; en este caso, la única variable que fue encontrada en distintos artículos fue el año de admisión a la carrera (figura 73).

**Figura 73.** Nube de palabras variables de entorno académico



**Fuente:** elaboración propia.

## 10.4. Factor gestión académica

Este factor engloba todas las variables que le permiten a una institución medir, de una u otra forma, el nivel de aprendizaje de un estudiante; es el foco central de la mayoría de los trabajos revisados. Una de las variables fundamentales son las notas de las materias vistas. De este grupo de variables, las que fueron mencionadas más veces, son las calificaciones y el rendimiento académico por semestre (figura 74).

**Figura 74.** Nube de palabras variables gestión académica

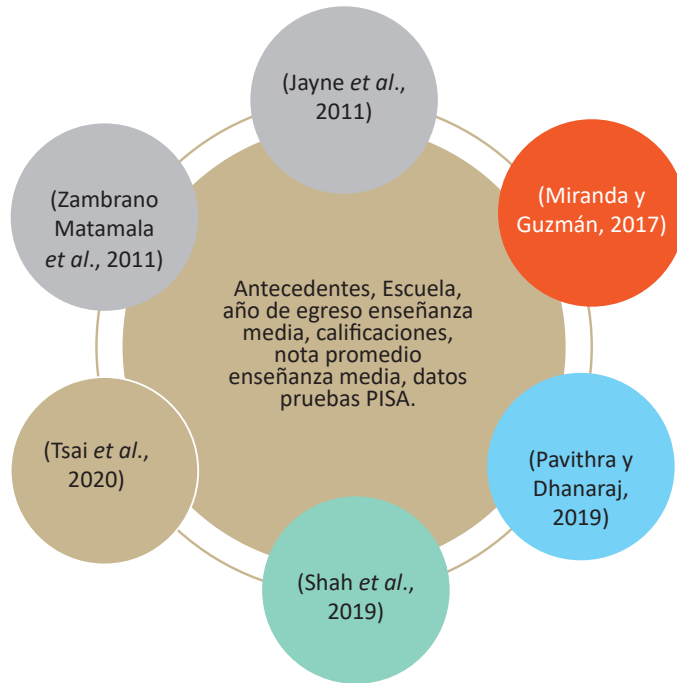


**Fuente:** elaboración propia.

## 10.5. Factor desempeño académico universitario y preuniversitario

Las variables de desempeño preuniversitario corresponden al grupo de información resultante al momento de ingresar a una institución educativa y se muestran en la figura 75, Estos registros históricos son datos como resultado del ICFES o Pruebas PISA, escuela en la que terminó sus estudios bachilleres, etc. La variable mayormente frecuentada es la nota promedio de enseñanza media. En cuanto a las variables de desempeño universitario, son las notas que el estudiante ha registrado durante su ejercicio como estudiante en la institución educativa. Estas notas pueden ser las registradas en un curso, en un semestre o a lo largo de toda su trayectoria académica en la institución.

**Figura 75.** Nube de palabras variables desempeño académico preuniversitario y universitario



**Fuente:** elaboración propia.

## 10.6. Factor *E-Learning*

Este factor engloba la información que, a diferencia de lo que sucede en instituciones con un método clásico de aprendizaje, se obtiene mediante el uso de entornos virtuales de aprendizaje (EVA) y se muestra en la figura 76. Comprende datos como número de clics, tiempo en línea y tiempo entre ingresos. La variable más encontrada en los estudios revisados fue el tiempo total en línea.

**Figura 76.** Nube de palabras variables *E-Learning*



**Fuente:** elaboración propia.

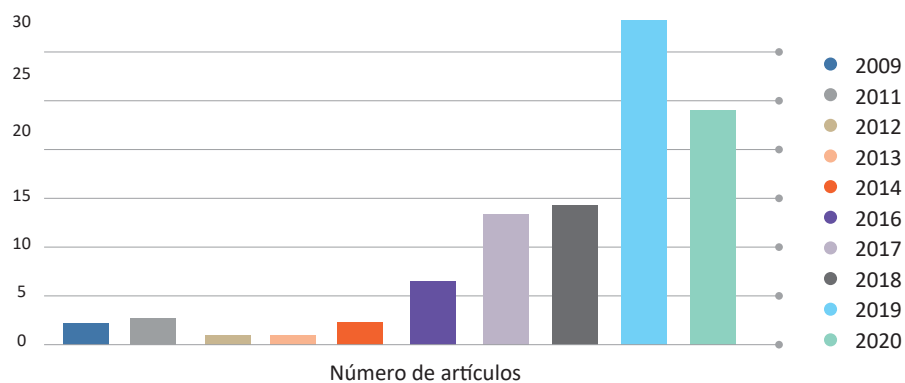


# Capítulo 11

## Redes neuronales: Investigación referencial

A continuación, se realiza una breve descripción y análisis detallado de cada uno de los artículos investigados durante el periodo 2010-2020. Se muestran con especial énfasis los algoritmos de aprendizaje automático no supervisado aplicados, variables utilizadas en los modelos, eficiencia de las técnicas o métodos aplicados, lenguaje de programación o *software* utilizado y resultados finales de cada estudio. Estos artículos se agruparon por periodos de tiempo (2009-2013, 2014-2017, 2018-2020) como se muestra en la figura 77.

**Figura 77.** Gráfico de estados del arte por año



**Fuente:** elaboración propia.

## 11.1. Investigaciones realizadas entre 2009 y 2013

Paliwal y Kumar (2009), en una escuela de negocios de la India, plantearon un modelo que busca predecir el rendimiento académico de los admitidos de tal manera que fuera posible analizar su potencial y tomar decisiones en el futuro sobre los requisitos de admisión. Para ello, se utilizaron modelos estadísticos convencionales y un modelo de red neural (feedforward multilayered). Se utilizaron variables como el puntaje de pruebas, los resultados académicos y la experiencia laboral. El modelo de red neuronal arrojó mejores resultados que la regresión logística.

En 2009, una IES griega en un entorno de *E-Learning* toma las respuestas de los estudiantes en evaluaciones o actividades de opción múltiple como datos de entrada para construir modelos de predicción del rendimiento académico en un curso de diez semanas. Se descubre que la predicción es posible desde la tercera semana. Se propone un modelo de regresión lineal y otro de red neuronal, siendo este último más efectivo (Lykourantzou *et al.*, 2009).

En 2011, en Inglaterra, en una IES se realiza un estudio en forma de mapeo de un elemento a varios elementos. En otras palabras, se utiliza como dato de entrada la calificación de un estudiante en cierta materia para predecir su calificación en otros cursos. Los modelos utilizados fueron el perceptrón multicapa (MLP), funciones de base radial (RBF), redes de densidad de mezcla (MDN), y el método no lineal variable latente de cartografía (GTM). Los mejores resultados fueron encontrados para MLP y RBF, pues se adaptan mejor a los datos [4].

En 2013, en una universidad de Nigeria, se realiza un modelo de red neuronal artificial en Matlab, para predecir la calificación de un estudiante antes de graduarse. Se utilizaron los registros históricos de treinta estudiantes y se obtuvo una precisión del 92 % (Usman *et al.*, 2013).

En 2011, se construyen modelos con base en bosques aleatorios y redes neuronales, en una IES de Estados Unidos para predecir el rendimiento académico de estudiantes universitarios inscritos en cursos de ciencias básicas, con el fin de tomar acciones correctivas y lograr que alcancen el éxito académico. Se toman como variables los datos de estado inicial y los datos recolectados a través de instrumentos elaborados específicamente para

evaluar las habilidades del estudiante en matemáticas y física. Se descubre que, cuando los estudiantes se inscriben a un curso, pero no realizan ese curso los resultados de la predicción son afectados. La manera de medir la efectividad del modelo de red neuronal fue a través del porcentaje de falsos positivos: 7,2% (Schalk *et al.*, 2011).

En 2012, en una IES de Estados Unidos se realiza un trabajo para comparar cuatro tipos de modelos de predicción del rendimiento académico de estudiantes de ingeniería. Los modelos son regresión lineal múltiple, red percepción multicapa, modelo de red de función de base radial y modelo de máquina de vector de soporte. La máquina de vector de soporte fue el modelo de mejores resultados (Huang y Fang, 2013).

Según Zambrano Matamala *et al.* (2011) en una IES latinoamericana ubicada en Chile, se realiza la implementación de una arquitectura de red neuronal para predecir el rendimiento académico de los estudiantes de Ingeniería Civil en áreas de la computación e informática. Este modelo puede predecir cómo será el rendimiento del estudiante tan solo utilizando los datos del rendimiento durante el periodo académico anterior al que se quiere estudiar. Se concluye que, al utilizar más datos del estudiante, como por ejemplo factores socioeconómicos, la predicción podría mejorar.

## **11.2. Investigaciones realizadas entre 2014 y 2017**

En el 2014, un artículo de Australia habla sobre la primera conferencia de *Data Mining*, en 2008. En este artículo también se introduce el tema del EDM en el campo educativo; se explica que, con el potencial de este campo, se puede descubrir cómo las personas aprenden con el objetivo de diseñar tecnología de aprendizaje mejor y más inteligente para informar a estudiantes y educadores (Baker, 2014).

En 2014, en un artículo escrito en Taiwán, se hace un estudio que asegura que, por medio de portafolios grabados (datos históricos), y técnicas como árboles de decisión, se pueden llegar a generar sistemas de alerta temprana para estudiantes en riesgo. Sin embargo, este trabajo no utiliza redes neuronales (Hu *et al.*, 2014).

En 2016, en Taiwán, se hace un estudio comparativo de dos modelos de predicción de rendimiento académico en estudiantes de postgrado de

Licenciatura en Matemáticas. Se implementó un modelo de análisis de regresión logística (LRA) y otro modelo de red neuronal (ANN). Se tomaron como variables la necesidad de servicios de discapacidad, la necesidad de servicios de apoyo y la edad del estudiante cuando habían presentado la solicitud a la universidad. Se identificó previamente que estos tres eran los factores que más habían contribuido a resultados de graduación exitosos y no exitosos. El modelo más eficiente es el de ANN, con una tasa de predicción promedio del 93,02%, frente a la tasa de predicción promedio del modelo LRA, que fue de 90,75% (Bahadir, 2016).

En China, se elabora un modelo de clasificación que busca predecir el rendimiento académico en una IES. Se utiliza un algoritmo no supervisado de red neuronal. Los grupos de variables utilizadas fueron los datos demográficos, los datos del rendimiento en el colegio y los datos personales. El modelo de MLP muestra una exactitud de entre el 72,2% y el 78,4% (Guo *et al.*, 2015).

En 2017, en Indonesia, se realizó un estudio aplicado a alumnos de ingeniería informática. Tan solo el 10% de los estudiantes en 2013 se habían graduado a tiempo. Se implementó un modelo de red neuronal utilizando como variables todos los datos registrados por la universidad acerca del estudiante, como créditos acumulados, estatus económico y laboral. La forma en que se evaluó el modelo fue por medio del error cuadrático medio (MSE), que fue del 0,00643 (Nurhuda y Rosita, 2017).

En Japón se elabora un estudio que propone como modelo de predicción de rendimiento una red neuronal recurrente utilizando los datos del estudiante registrados por la institución. La precisión del modelo fue de un 95% (Okubo *et al.*, 2017).

En un estudio de India, se utilizan técnicas de aprendizaje profundo para predecir el desempeño de los estudiantes. Se construye una red neuronal con retropropagación y una red neuronal recurrente. Las variables usadas fueron los datos generados por el registro académico de los estudiantes. En este trabajo se compararon varias arquitecturas de redes neuronales: redes neuronales recurrentes con una sola capa oculta, red de memoria a corto plazo, a largo plazo, red de memoria con múltiples capas ocultas bidireccional, red de memoria a corto plazo con múltiples capas ocultas. Las mejores

arquitecturas son las de redes neuronales recurrentes. Los modelos con mejor precisión fueron la red neuronal con alimentación hacia adelante, con un 68 %, y la red neuronal recurrente (LSTM), con un 76 % (Patil *et al.*, 2017).

Otro estudio hecho en Grecia, en 2016, utiliza los datos generados a través de la plataforma Moodle, basados en cuatro pilares de aprendizaje: creación de contenido colaborativo con wiki, interacción a través de correo electrónico, interacción de contenido medida por archivos vistos y autoevaluación a través de cuestionarios en línea. El modelo de MLP usado mostró una tasa de clasificación correcta del 98,3 % (Zacharis, 2016).

En 2017, en una IES de Irlanda, se implementa una red neuronal recurrente para analizar la deserción académica en todos los estudiantes de la institución, independientemente del programa en el cual estuvieran registrados. Las variables de entrada del modelo son los datos generados durante la interacción en la plataforma virtual de aprendizaje que maneja la institución y los datos de registro de cada estudiante. El modelo se evalúa con la curva ROC<sup>1</sup>: el ROC mínimo es del 0,6 (Corrigan y Smeaton, 2017).

En 2016, en Latinoamérica, se realizan dos estudios: uno realizado en México, en el que se utiliza el algoritmo de red neuronal de Microsoft para predecir la deserción de los estudiantes. Como variables del modelo se tienen los datos personales y académicos de los estudiantes. Este algoritmo arrojó una exactitud del 98 % (Hernandez Gonzalez *et al.*, 2016).

El otro estudio es elaborado en Cuba. Se crean dos redes neuronales con la misma estructura para predecir los resultados que alcanzarán los estudiantes en las asignaturas Estructuras de Datos I y II. Las variables que se tienen en cuenta son las notas de las materias prerrequisito, las notas en ciencias básicas, el número de veces que suspendió alguna de estas materias y el género. Los modelos fueron elaborados en MATLAB. Las redes neuronales solo tienen una capa oculta, pues se hicieron pruebas con más capas y la efectividad de predicción resultó menor. La efectividad de predicción de estos modelos fue del 78 % y el 75 % (Álvarez Blanco *et al.*, 2016).

---

<sup>1</sup> La curva ROC es uno de los métodos estadísticos para determinar qué tan efectivo han sido los resultados de un modelo de red neuronal artificial.

En 2017, en Brasil, se elaboraron dos estudios en IES: por un lado, se utiliza el EDM para hacer una predicción temprana de la deserción estudiantil. Se utilizaron modelos de *deep learning* (DL) y otros de modelos que pertenecen al paradigma clásico del EDM. Las variables usadas tienen que ver con factores sociodemográficos, económicos y académicos. Se logró predecir hasta en un 71,1 % los casos de deserción (Martins *et al.*, 2017). Otro estudio busca evaluar la efectividad de las técnicas para predecir la deserción en estudiantes de programación. Se utiliza un algoritmo de red neuronal perceptrón multicapa (MLP). Se tienen en cuenta dentro de las variables el rendimiento de los estudiantes en seminarios y exámenes. Esta vez la efectividad de precisión es del 75 % (Costa *et al.*, 2017).

En una institución chilena, se realiza un estudio que trata de determinar las razones por las cuales una persona abandona sus estudios universitarios. Los datos se obtuvieron de carreras de ingeniería e incluye toda la historia académica de los estudiantes, incluso los beneficios académicos, si los tiene. Se utilizaron tres diferentes clasificadores: el primero enfocado en árboles de decisión, el segundo es un modelo bayesiano y por último un algoritmo de redes neuronales. En cuanto a la construcción de la red neuronal, se implementó un modelo de perceptrón multicapa (MLP) y se concluye que tanto el rendimiento académico, el puntaje de ingreso a la universidad (PSU) y la situación económica de un estudiante, son factores determinantes para que un estudiante decida o no desertar (Miranda y Guzmán, 2017).

En Perú, se desarrolló un estudio que emplea diferentes técnicas de EDM (regresión logística, árboles de decisión, redes bayesianas y redes neuronales) para predecir el éxito o el fracaso de un estudiante inscrito al curso de Estadística General. Se aplica regresión logística para determinar cuáles variables son más significativas y útiles para los modelos de predicción. De esta manera, se establece que las variables influyentes en el resultado del curso de Estadística General fueron el promedio ponderado, la situación del curso, la nota en diferencial y el número de veces que repitió diferencial. Finalmente, se halló que el método que mejor funcionaba para la predicción era la red bayesiana, con un 71 % de exactitud (Menacho Chiok, 2017).

### 11.3. Investigaciones realizadas entre 2018 y 2020

En 2018, en Indonesia, se realizó un estudio para predecir el desempeño académico por medio de técnicas como el modelo lineal generalizado (GLM), aprendizaje profundo y árbol de decisión. Las variables utilizadas fueron los datos del rendimiento del estudiante en primer semestre. La precisión que tuvieron los algoritmos fue de 66,6% para GLM, 67,6% para la red neuronal profunda, y 60,6% para el árbol de decisión (Tanuar *et al.*, 2018).

Los estudios encontrados en China buscan predecir el rendimiento académico: uno de ellos utiliza la red neuronal de propagación hacia atrás (BP-NN) en los datos de sesenta estudiantes de secundaria. No se especifica exactamente un valor de rendimiento del modelo, pero, de acuerdo con la información presentada, los valores de predicción fueron muy cercanos a los valores reales. Otro estudio trata de predecir cuáles son las limitaciones que tienen los estudiantes de un curso de Diseño Digital. Para ello, se utilizaron algoritmos de aprendizaje automático entre los que se incluyen redes neuronales artificiales (ANN), máquinas de vectores de soporte (SVM), logística regresión, clasificadores Naive Bayes y árboles de decisión. Las variables de entrada fueron tiempo promedio, número total de actividades, promedio tiempo de inactividad, número medio de pulsaciones de teclas y actividad relacionada total. Estas variables se evaluaron en cada actividad de las sesiones del curso de Diseño Digital. Se utilizó MATLAB como herramienta tecnológica. La precisión del modelo de red neuronal fue de 85% (Hussain *et al.*, 2019b).

En Taiwán, la tasa de natalidad ha venido disminuyendo los últimos años, por ende, la población estudiantil también. Por esta razón, se han dedicado esfuerzos a retener a los estudiantes, pues la deserción académica resulta seis veces más costosa que un tratamiento de reducción. En un estudio se elaboró una red neuronal para la predicción de la deserción académica con base en un algoritmo llamado *stacked denoising autoencoder*. Se utilizaron datos de los estudiantes de los últimos años: el 50% de los datos corresponde a estudiantes que abandonaron la academia, y el otro 50% de los datos corresponde a estudiantes que no abandonaron. La estructura de la red neuronal consiste en cuatro capas: la primera capa tiene sesenta neuronas, mientras que la segunda y la tercera capa tienen 128 neuronales cada una, y la última capa



tiene dos neuronas. El error entre la salida de los datos y la salida esperada se calcula mediante el error de entropía, la función de pérdida. Se utiliza el descenso del gradiente para minimizar el error. Para la construcción de la RNN se utilizó Python y TensorFlow API. En el preentrenamiento se utilizaron los algoritmos denoising autoencoder, algoritmo del error cuadrático medio, sparsity algorithm, regularization algorithm y *backpropagation*. De las capas ocultas se descubrió que, cuando el número de unidades ocultas es inferior a 64, o el número de capas ocultas es más de cuatro, el modelo no puede tener buena precisión para los datos de prueba. Cuando el número de capas ocultas es cuatro y tiene 128 unidades ocultas, la precisión del modelo para datos de prueba es el más alto. Si el ruido de los datos es muy alto, el modelo no puede predecir con exactitud. La exactitud de este modelo fue de 93,21 % (Özkan *et al.*, 2018).

En 2019, en China se continúan haciendo investigaciones, esta vez sobre la predicción del rendimiento académico en MOOC. Se plantea un modelo de *deep learning* que consiste en la combinación de una red neuronal convolucional, redes de memoria a corto plazo y una máquina de vectores de apoyo. Como variables de entrada se toman los registros de eventos en un determinado intervalo de tiempo. Esta red neuronal tuvo una precisión promedio de 96 % (Wu *et al.*, 2019).

Otro estudio realizado también en China busca, por medio de técnicas de aprendizaje automático, predecir el rendimiento académico en cursos en línea por medio de la observación de los estudiantes ante la exposición al Internet. Así pues, se centra en cómo los modelos utilizan los algoritmos de árbol de decisión, red neuronal y máquina de vectores de apoyo. Dentro de las variables que se evalúan están el tiempo en línea, el tiempo sin conexión, el volumen de descarga, el volumen de carga. Este estudio resalta que la disciplina en los estudiantes es una variable muy representativa en el rendimiento de los estudiantes. Por otro lado, se concluye que el punto de vista en torno a las características del tiempo en línea, y el tiempo consumido en Internet da lugar a un rendimiento inesperado entre diferentes conjuntos de datos. Los autores consideran que la información recopilada por la actividad de un alumno por medio de internet es totalmente diciente acerca del rendimiento académico de un estudiante. Después de haber probado el modelo, se llega



a que el modelo de SVM tiene entre un 69,55 % y un 72,75 % de exactitud, y el modelo de la red neuronal tiene una exactitud de entre 67,75 % y 70,95 %. En este caso, el modelo de SVM resultó funcionar mejor (Xu *et al.*, 2019).

En India se encontraron varios estudios acerca de la predicción del rendimiento académico. Uno de ellos lo hace a través de diferentes métodos de clasificación de *Machine Learning*. Entre los modelos implementados están J48, árboles aleatorios, perceptrón multicapa (MLPNN) y red neural de base radial (RBFN). Las variables de entrada son el registro de actividades en DEEDS (simulador estudiantil de electrónica digital); es decir, los datos generados de los ejercicios y actividades dentro de la plataforma de estudio. Se logró una precisión de entre el 70 % y 80 % (Arumugam *et al.*, 2019).

En otra institución llamada Sree Saraswathy Thyagaraja College de la India, el estudio de predicción se hizo utilizando varios modelos de EDM, dentro de los cuales se encuentran Naive Bayes, árboles aleatorios, J48, árbol de decisión y percepción multicapa (MLP). Las variables de entrada son datos de los estudiantes de Licenciatura en Ciencias de la Computación. Estos datos son sexo, lugar de residencia, ocupación del padre, ingresos anuales del estudiante, notas de secundaria del estudiante, y porcentaje cursado del semestre. Los modelos se construyen en el *software* WEKA y son medidos a través de las curva ROC; en este caso, para los modelos de red neuronal, este valor es de 0,6892 (Pavithra y Dhanaraj, 2019).

Nuevamente en la India se realiza un estudio con la información de estudiantes de tres instituciones educativas. En total se trabajó con los datos de 10 140 estudiantes. Las variables eran los datos generados del registro de su rendimiento académico. En este estudio se utiliza una red neuronal secuencial con el método de optimización de Adam. Esa red neuronal se compara con otros métodos de clasificación como Adaboost. La red neuronal alcanza una precisión del 95,34 % (Hussain, Muhsion *et al.*, 2019).

Un último estudio realizado en este mismo país busca predecir el rendimiento académico por medio de la utilización de modelos de *Machine Learning*. En este estudio se tienen en cuenta dentro de las variables de entrada la segmentación psicográfica del estudiante, dentro de la cual se incluyen variables IAO (actividad, interés, opinión). Se evalúan treinta y tres atributos diferentes entre los que se incluyen escuela, género, edad, dirección de

residencia, tamaño de la familia, estado de los padres, nivel de educación de los padres, trabajo de los padres, tiempo de viaje a la escuela, entre otros. Los algoritmos aplicados son SV model, bosques aleatorios, LGR, adaboost, gradient boosting classifier, red neuronal recurrente RNN y red neuronal artificial. La precisión de dichos algoritmos se registró así: DT, 88,2%; SVM, 89,8%; random forest, 91,7%; LGR, 59,8%; gradient boosting, 93,8%; adaboost, 85,13%; ANN, 88,89%; y RNN, 69,65% (Shah *et al.*, 2019).

En Indonesia se realiza un estudio para predecir el rendimiento académico mediante la aplicación de redes neuronales artificiales. La variable utilizada es el valor del índice de rendimiento semestral (IPS) de los cuatro primeros semestres de los estudiantes. Se implementó una red neuronal con retropropagación. La efectividad de predicción de este modelo se registró de la siguiente manera: MSE de entrenamiento, 0,016175; coeficiente de relación de entrenamiento, 0,94353; MSE de prueba, 0,12188; y coeficiente de relación de prueba, 0,56071 (Bhakti, 2019).

Otro estudio realizado en Indonesia busca predecir el rendimiento estudiantil y de esta manera tomar decisiones al respecto para lograr que la deserción disminuya. En esta ocasión, se implementa un modelo de red neuronal de avance, y las variables de entrada corresponden a los datos del estudiante registrados a través del rendimiento académico. La red neuronal fue programada en Python y tuvo una precisión por debajo del 50%, valor que se considera bajo pues la predicción debe ser lo más acertada posible. Dentro de las razones que se consideran influyentes en la baja precisión de la red neuronal se destaca el tamaño de los datos y la relevancia de las actividades de los estudiantes usadas para predecir (Yunita *et al.*, 2019).

En Corea del Sur, se hizo un estudio comparativo de modelos de predicción del rendimiento académico en la materia de matemáticas. Se incluye en esta investigación una técnica de análisis estadístico, algoritmos de *Machine Learning* como algoritmos de aprendizaje de máquinas (ML), y una de arquitectura de aprendizaje profundo. Como variables de entrada se trabajan 43 datos dentro de los que se incluye información doméstica, académica e información sobre la actitud de cada estudiante. Como variable objetivo se analiza la calificación de cada estudiante en matemáticas. La precisión de cada algoritmo fue de entre 56% y 63% para SEM; 79% y 89% para MLP;

75 % y 93 % para SMO; 80 % y 93 % para C5; 77 % y 89 % para LR; y 75 % y 88 % para DL (Sokkhey y Okazaki, 2019).

En Japón se realiza un estudio para predecir el rendimiento académico con el fin de disminuir la tasa de deserción en un curso cuya modalidad es semipresencial. Se utiliza una red neuronal alimentada con los datos generados a través de la plataforma virtual de aprendizaje y el rendimiento en los últimos cursos de los estudiantes. Finalmente, el modelo arrojó una precisión del 65 % (Sukhbaatar *et al.*, 2019).

En Arabia Saudita se desarrolla un modelo en Python para predecir el rendimiento académico. Para ello, se utilizan los datos de Open University Learning Analytics, que son datos de libre acceso. La predicción se hace a través de dos arquitecturas, una de regresión lineal y la otra de red neuronal de memoria a corto plazo LSTM. La red neuronal arrojó muchos mejores resultados: un 97,25 % para aprendizaje, y un 92,79 para precisión (Hassan *et al.*, 2019).

En Malasia también se hace un estudio sobre la predicción del rendimiento académico. Esta vez, el ambiente es un entorno virtual. Los datos de entrada son los recogidos a través de la plataforma Moodle: la muestra a analizar son los novecientos estudiantes de diez diferentes carreras universitarias. La arquitectura de red neuronal propuesta es una feedforward neural network (FFNN), la cual tuvo una precisión del 90 % (Altaf *et al.*, 2019).

En 2020, en Taiwán, se encontraron dos estudios: uno de ellos busca predecir el rendimiento académico de los estudiantes de programación de *software*, pues en este tipo de programas los alumnos suelen desmotivarse por la complejidad de los temas y las consecuencias van desde bajo rendimiento hasta deserción. Se encontró que el factor más influyente es la nota de matemáticas de secundaria y la aptitud para programar. Se utilizaron dos clases de redes: una red neuronal de avance y una red neuronal complementaria. Esta última arrojó mejores resultados, una exactitud del 78,96 % (Mahaweerawa *et al.*, 2020).

El otro estudio busca retener, en la mayor medida de lo posible, a los estudiantes en Taiwán, debido a la baja tasa de natalidad. La consecuencia de este fenómeno es que las universidades admitan fácilmente a las personas sin tener en cuenta sus aptitudes o historial académico. En este estudio se

busca predecir el rendimiento académico. Para ello se utilizó un modelo estadístico y un modelo de redes neuronales. Después de varios experimentos, este estudio desarrolló una recomendada estrategia de predicción de la siguiente manera: primero, se utiliza un análisis de regresión logística para identificar factores que influyen en las fallas de aprendizaje; posteriormente, el aprendizaje profundo se emplea para que, con cálculos efectivos, se logren identificar patrones en los datos y se puedan hacer predicciones. Gracias a la regresión logística, se establecieron cuatro variables correlacionadas significativamente con la deserción: rendimiento académico (es decir, porcentaje de clasificación de la clase), solicitud de préstamo estudiantil, número de ausencias de la escuela y número de sujetos alertados. La precisión del modelo estadístico fue de un 68 % y la precisión de la red neuronal fue de 77 % (Tsai *et al.*, 2020).

Un estudio en Tailandia compara varias técnicas de minería de datos educativa para predecir el rendimiento académico. Se utilizan métodos como Naive Bayes, optimización secuencial, red neural artificial, K-nearest neighbor, REPTree, árboles de decisión parciales y bosques aleatorios. Las variables que se utilizaron en este estudio fueron año de admisión, género, facultad programa, estado (graduado, activo), certificado de grado, tipo de admisión, estado civil de los padres, nivel académico de los padres, ingresos económicos de los padres y puntaje al momento de la inscripción. En este estudio se encontró que el algoritmo más eficiente fue el de bosques aleatorios, con una precisión del 94,7 %, superando a la red neuronal, que tuvo una precisión de 87,4 % (Nuankaew y Thongkam, 2020).

En Pakistán se elaboró un estudio con el fin de observar el rendimiento académico de los alumnos que pertenecen a una modalidad de *M-Learning* —es decir, que reciben sus clases principalmente por medio de aparatos electrónicos móviles—. Para esto, se implementaron diferentes modelos como el de bosques aleatorios y red neuronal profunda. Las variables evaluadas fueron tiempo promedio de estudio, número de veces que se accede al recurso de texto, número de veces que se accede al recurso de video, rendimiento del módulo 1.2.3, número de visitas a lugares académicos, número de visitas a lugares sociales, participación en el grupo de discusión en línea, número de veces que se publicó el problema, número de veces que se resolvió un

problema, número de veces que se intentó una prueba, tasa de repetición de tema, calificaciones finales derivadas de la puntuación de desempeño final, característica dependiente, categórico (a, b, c, d, f). La precisión promedio de la red neuronal fue de 85 % (Adnan *et al.*, 2020).

En Pakistán también se estudió el rendimiento académico en entornos virtuales. Se utilizan métodos como la regresión lineal (RL), la máquina de vectores de soporte (SVM) y la red neuronal profunda (ANN). Las variables de entrada de estos modelos fueron los datos de interacción del estudiante con la VLE (entorno virtual de aprendizaje) y sus datos demográficos. El estudio demuestra que la red neuronal superó tanto al SVM como a la RL. Mientras que la red neuronal tuvo una precisión de clasificación entre 84 %-93 %, la regresión logística alcanza una precisión del 79,82 %-85,60 %, y la máquina de vectores de soporte muestra un 79,95 %-89,14 % (Waheed *et al.*, 2020).

En India se utiliza el modelo de regresión lineal y la red neuronal multicapa para predecir el rendimiento estudiantil. Se lleva a cabo un preprocesamiento de datos, transformación y conjunto de datos y finalmente la preparación. Esto se hizo por medio SQL server management studio (SSMS) y SQL server servicio de integración (SSIS). El conjunto de datos transformado se importó en R. Los datos posprocesados se importan al H20 para la validación de modelos de aprendizaje. El conjunto de datos se recopiló de las tres universidades diferentes de Assam, India: Ellosson Digboi College, Duliajan College y Doomdooma College. El conjunto se divide en 80 % para entrenamiento y 20 % para pruebas. El modelo de red neuronal fue creado con dos capas ocultas y 64 unidades para mitigar el sobreajuste. Se hace la última capa lineal para que la red sea libre de predecir valores en cualquier rango. Las variables de entrada son los datos generados del rendimiento académico del estudiante en cada semestre. Los modelos son evaluados mediante la puntuación media absoluta (MAE) y la pérdida. Para el caso de la red neuronal, estos valores son de 1,61 para el MAE; y 4,7 para la pérdida. Para el caso de la regresión lineal, el MAE es de 1,97; y la pérdida es de 6,7. El modelo de red neuronal superó los resultados de la regresión lineal (Hussain *et al.*, 2020).

Un estudio en China identificó que había una laguna en la investigación de EDM: siempre se hablaba de métodos clásicos para predecir el rendimiento, pero en escasas ocasiones se utilizaban algoritmos de *deep learning*. Lo que busca este trabajo es fortalecer el poder de predicción en el ámbito académico con respecto al rendimiento académico. Este estudio fue llevado a cabo para el aprendizaje K-12, Los investigadores encontraron que la predicción es personalizable, puesto que no intervienen los mismos factores en K-12 que en la universidad, que además está altamente influenciado por instructores, estudiantes y todos los actores del entorno de aprendizaje. Los hallazgos indican que el tiempo total gastado en sistemas de aprendizaje o cuestionarios, la frecuencia total del comportamiento, la frecuencia de acceso al contenido, la frecuencia de lectura de la discusión, la frecuencia de publicación de discusión y la frecuencia de publicación de respuesta, y la frecuencia de intentos en el cuestionario son las variables de comportamiento más importantes. En cuanto al rendimiento de la red neuronal se indica que tuvo una precisión global del 86,8% (Hung *et al.*, 2020).

Unos investigadores chinos utilizaron un modelo *Knowledge Tracing* (KT), que hace uso de una red neuronal LSTM. Este modelo recibe el nombre de NAKTM. La ventaja de esta arquitectura es que puede predecir el comportamiento del estudiante tanto a corto como a largo plazo, y no necesita hacer un preprocesamiento de datos antes de introducirlos a la red neuronal. Se utilizaron diferentes *datasets* de plataformas educativas gratuitas. Las variables de entrada corresponden a los registros de las interacciones que los alumnos tienen a través de la plataforma. Las métricas de evaluación del modelo para este caso fueron el AUC, que en promedio fue de 0,799, y el R2 cuyo promedio es 0,244 (Zhang *et al.*, 2020).

En la universidad del suroeste de China se usó un modelo de red neuronal profunda para calcular la tasa de aprobación de los estudiantes en la modalidad *online*, y se combina con algoritmos de árbol de decisión (DT) y máquinas de vectores de soporte (SVM). Las variables que utilizaron fueron sexo, años, título, tiempo de estudio, actividad de comunicación, número de opcionales (cursos), número de revistas publicadas, número de temas, número de respuestas al tema, temas de preguntas y respuestas, número de recursos compartidos, número de tareas asignaciones planteadas por consejeros,



número de tareas planteadas por expertos, curso requerido completado, calificaciones requeridas del curso, grado electivo, grados de trabajo, resultados de la actividad, calificaciones numéricas sin conexión, nota final numérica. Se hicieron varios experimentos en los que se modificaba la cantidad de datos de entrenamiento y de datos de validación. La precisión de la predicción osciló entre un 80 % y un 96 % (Ma *et al.*, 2018).

En Bangladesh se han utilizado diferentes algoritmos de clasificación para estimar de manera oportuna el rendimiento de un estudiante antes de que pueda reprobar. Se utilizan cuatro clasificaciones para evaluar el desempeño: excelente, buena, mala y reprobada. Los algoritmos implementados fueron máquina de vectores de soporte (SVM), K-vecino más cercano (KNN), regresión logística, árbol de decisión, adaboost, clasificador de voto ponderado y perceptrón multicapa (MLP). Las variables que se toman en cuenta son todas las relacionadas con la actividad académica del estudiante: asistencia, si ha repetido materias, porcentaje de evaluaciones presentadas, promedio de calificaciones, envío de tareas. El modelo MLP tuvo una precisión del 75 %, superada solo por los modelos de SVM, con una precisión del 75,96 %, y el clasificador de voto ponderado, con una precisión del 81,73 % (Zulfiker *et al.*, 2020).

En Arabia Saudita se realizó un estudio para determinar si se podía predecir el rendimiento de los estudiantes antes de cursar una carrera universitaria, con el fin de que los directivos de la universidad pudieran tomar decisiones en cuanto a los requerimientos de admisión. Se utilizaron varias herramientas de EDM, y se tomaron como variables el rendimiento promedio en la secundaria, el puntaje de examen de aptitud y el puntaje de prueba de admisión. Los resultados concluyeron que el factor más influyente en la predicción del rendimiento académico es el puntaje de prueba de admisión, y que el modelo que tuvo una mayor efectividad fue el de red neuronal, con una precisión por encima del 79 % (Mengash, 2020).

En 2020, en Australia, se propuso hacer una red de predicción de rendimiento. Para ello, se propone una red de predicción del rendimiento académico con base en el compromiso secuencial (SEPN), que comprende principalmente dos componentes: detector de compromiso y predictor secuencial. Por un lado, el compromiso detector funciona como una red

neuronal convolucional (CNN) para detectar patrones de participación a través de las actividades diarias de los estudiantes. Por otro lado, el predictor secuencial tiene la capacidad de guardar memoria a corto plazo (LSTM), y aprende la interacción del compromiso y características demográficas. Para esto se tomaron como variables de entrada los datos generados por la interacción de un estudiante en la plataforma virtual de aprendizaje. Este modelo dio buenos resultados en comparación con otros modelos de *Machine Learning*. Su tasa de precisión máxima fue de 61 % (Xiangyu Song *et al.*, 2020).

En 2018, en la Universidad Tecnológica de Budapest, Hungría, se realizó un estudio para predecir la deserción estudiantil con diferentes herramientas de EDM. Para ello, se tuvo en cuenta el rendimiento académico durante la secundaria. Se utilizaron los clasificadores y algoritmos basados en árbol de decisión, Naive Bayes, K-vecino más cercanos, modelos lineales y aprendizaje profundo con diferentes configuraciones de entrada. Los métodos se probaron utilizando validación cruzada diez veces y el AUC<sup>2</sup> de los mejores modelos fue de 0,808 del modelo de *gradient boosted trees* y 0,811 del modelo de *deep learning*. Se encontró que las áreas de humanidades estaban muy relacionadas con el éxito académico de un estudiante, a pesar de que la universidad estuviera enfocada en carreras técnicas (Nagy y Molontay, 2018).

Por otro lado, la Universidad Hellenic Open, en Grecia, fue una de las primeras instituciones en brindar educación terciaria por medio de cursos masivos en línea (conocidos como MOOC<sup>3</sup>, por sus siglas en inglés). Por eso, en esta universidad se ha trabajado por adoptar un modelo de *learning analytics* que ayude a mejorar los niveles de servicio. En esta ocasión se realizó un estudio con un algoritmo de MLP y se aprovecharon los datos generados por los estudiantes durante la interacción dentro de la plataforma virtual. Los datos indicaban variables como género, cantidad de inicios sesión en el módulo, cantidad de inicios sesión en el foro, respuestas del foro, tiempo dedicado en la plataforma, notas de cuestionarios principales, notas de cuestionarios de opción múltiple por semana, autopruebas de evaluación.

---

2 (O curva ROC) es una forma gráfica de evaluar este tipo de modelos.

3 Massive Open Online Course.



Para medir el modelo, se utilizó como indicador el error cuadrático medio o MSE, el cual fue de 2,45 (Gkontzis *et al.*, 2018).

En 2019, en un estudio realizado en Serbia utilizó técnicas de aprendizaje automático supervisado para descubrir qué estudiantes se encontraban en riesgo de deserción, y así predecir el rendimiento a futuro. Se hizo un análisis tanto de regresión como de clasificación con métodos tales como K-NN, SVM, ANN, árboles de decisión, regresión bayesiana y regresión lineal. Para el caso de los modelos de red neuronal, se aplicó el método de entrenamiento de retropropagación con validación cruzada, con el que se probaron diferentes redes neuronales *feedforward* de una y dos capas ocultas (Tomasevic *et al.*, 2020).

En Turquía, algunas investigaciones utilizan redes neuronales para pronosticar el rendimiento de los alumnos. En un estudio se utilizan dos clases de redes neuronales: radial basis function neural network o función base radial, en español (RBFNN), y *backpropagation* o retropropagación (BP). Las variables que se utilizan como datos de entrada son el conjunto de datos relacionados con el rendimiento académico (estado parental, tamaño de la familia, etc.). El BP tiene cuatro capas ocultas y quinientas neuronas por capa. La función de activación sigmoidea se utiliza para la activación de cada neurona en cada capa, y la tasa de aprendizaje y la tasa de impulso fueron de 0,0009 y 0,90, respectivamente. Las iteraciones máximas se determinan como 1000, lo que produjo el error mínimo. Para RBFNN, se usa la neurona de salida única. Máximo 3000 iteraciones son determinadas para ser utilizadas con el fin de lograr un error mínimo y sin un ajuste excesivo. El método más efectivo fue la RBFNN, con 85 % de exactitud (Sekeroglu *et al.*, 2019a).

Otro estudio realizado en Turquía trata de predecir el rendimiento académico en entornos virtuales. Se tomaron como variables de entrada el tiempo dedicado al contenido, el número de entradas al contenido, la puntuación de los deberes, el número de asistencia a sesiones en directo, el tiempo total dedicado a sesiones en directo, el número de asistencia a los cursos archivados y el tiempo total dedicado a los cursos archivados. Estos datos de 3518 estudiantes alimentan la red neuronal con un precisión de 80,47 % (Aydoğdu, 2020).

Un estudio más realizado en Turquía aplica tres algoritmos de aprendizaje automático para hacer una predicción del rendimiento académico.

Los modelos que se aplican son *backpropagation* (BP), *support vector regression* (SVR) y memoria a corto plazo (LSTM). También se usa en la etapa de clasificación el clasificador *gradient boosting* (GBC). El algoritmo *backpropagation* es una red neuronal supervisada que se usó tanto para clasificar como para predecir datos. LSTM es otro tipo de red neuronal que obtuvo mejores resultados en la predicción (este tipo de red neuronal funciona muy bien cuando se trata de datos de series temporales que necesitan memorizar las entradas anteriores de la red neuronal). Estos modelos de *Machine Learning* fueron aplicados para predecir el rendimiento en cursos de matemáticas y en un curso de portugués, con una ratio de 40 % y 30 %. Para la ratio de 40 % del curso de matemáticas, el LSTM obtuvo un MSE de 0,014 y un R2 de 0,775, y la red neuronal BP obtuvo un MSE de 0,019 y un R2 de 0,694. Para la ratio de 30 %, el LSTM tiene un MSE de 0,019 y un R2 de 0,727, y la red neuronal BP registró un MSE de 0,024 y un R2 de 0,646. Para la ratio de 40 % del curso de portugués, el LSTM obtuvo un MSE de 0,0055 y un R2 de 0,824, y la red neuronal BP obtuvo un MSE de 0,085 y un R2 de 0,728. Para la ratio de 30 %, el LSTM tiene un MSE de 0,0063 y un R2 de 0,765, y la red neuronal BP registró un MSE de 0,0079 y un R2 de 0,708.

Como modelos de clasificación para una ratio de 40 %, se obtuvo una precisión de 80,91 % para BP, 79,38 % para SVM y 74,04 % para GBC. Para un ratio del 30 %, los modelos de clasificación registraron una precisión de 87,78 % para BP, 83,20 % para SVM y 82,44 % para GBC (Sekeroglu *et al.*, 2019b). El rendimiento de las redes neuronales se midió a través de la precisión, el MSE y el R2. Para el caso del modelo de *backpropagation*, la precisión fue de 87,78 %, el MSE fue de 0,0079, el R2 fue de 0,708. En este trabajo se consideraron dos conjuntos de datos para la predicción y la clasificación del rendimiento de los estudiantes, respectivamente, y se utilizaron cinco algoritmos de aprendizaje automático para dieciocho experimentos.

En Irlanda, se realiza un estudio para predecir la deserción académica de los estudiantes de cursos en línea. Las variables que se tienen en cuenta son de información demográfica, el historial de notas y la interacción con el entorno virtual de aprendizaje (VLE). Se plantea un modelo de red neuronal *backpropagation*. Esta tiene un AUC de 75 % (Jha *et al.*, 2019).

Nuevamente los cursos en línea son motivo de estudio. En Alemania se propone un modelo de red neuronal convolucional para predecir el abandono de los estudiantes de MOOC. La razón es la tasa tan alta de abandono en este tipo de educación. Gracias a la interacción que el estudiante tiene en las plataformas virtuales, se crean datos que se pueden examinar: flujo de clics, envío de trabajos, información de rendimiento, plan de estudios, publicaciones en foros, información interactiva e información demográfica de los estudiantes. Esta red neuronal fue diseñada en WEKA. A diferencia de otras predicciones de abandono basadas en métodos de aprendizaje automático y en métodos disponibles en la literatura, este modelo procesa directamente los datos sin la necesidad de preprocesamiento o extracción manual de características. La comparación de DP-CNN con otros algoritmos de referencia no conduce a una conclusión intuitiva. La precisión de este modelo fue del 86 % (Qiu *et al.*, 2019).

En 2020, en una universidad en Italia, se utilizó el *Deep Learning* para predecir la deserción académica. Se usaron tres arquitecturas de redes neuronales convolucionales: ResNetV2 (RNV2), ResNetV4 (IRNV4) y DFSV1 (creada por los autores). Para hacer todas las pruebas, se utilizó el contenido de la biblioteca bnlearn. Se utilizaron como variables los datos estadísticos de la universidad como datos administrativos, datos administrativos académicos y datos académicos. Se encontró que los datos académicos son el grupo de datos más efectivos para predecir la deserción. La precisión de los modelos de aprendizaje profundo varió desde el 67,1 %, para los estudiantes de primer año, hasta el 94,3 %, para los estudiantes de tercer año (Agrusti *et al.*, 2020).

En la Universidad de Salamanca en España se utilizaron varios algoritmos de *Machine Learning* como método de predicción del rendimiento académico. Dentro de esos algoritmos está el árbol de decisión, el bosque aleatorio, el XGBoost y el perceptrón multicapa. Las variables que se evaluaron tienen relación con la actividad del alumno en la plataforma virtual de estudio. El número de veces que los estudiantes hayan accedido a los recursos disponibles en las plataformas VLE se identificó como un factor clave que puede afectar el desempeño de los estudiantes. El MLP superó a los demás algoritmos con una exactitud de 0,782 % (Rivas *et al.*, 2021).

En Inglaterra, para disminuir la deserción en los MOOC, se desarrolló un modelo predictivo para conocer el desempeño de los estudiantes en un futuro. Se tomaron como variables la interacción en la página de inicio, el tiempo en el foro, las respuestas en el foro, la sesión del quiz y las notas de las evaluaciones. El modelo, que consiste en una red neuronal multicapa, tuvo un RMSE del 29,6 y un R2 de 0,42 (Alshabandar *et al.*, 2020).

En 2019, en Nigeria, se hace un estudio para averiguar qué tan efectivo es usar métodos de predicción con datos de los estudiantes de ingeniería, teniendo en cuenta como variables influyentes el programa de estudios, el año de ingreso y el promedio de calificaciones (GPA) de los tres primeros años de permanencia en la universidad. Se utilizaron seis algoritmos de *Machine Learning*, y uno de los destacados es red neural probabilística (PNN), que se basa en el ajuste dinámico de la descomposición (DDA). Se utilizaron también los algoritmos de bosque aleatorio, árbol de decisiones, Naive Bayes, conjunto de árboles y regresión logística. El modelo de red neuronal tuvo una exactitud del 85,89%, el coeficiente de Kappa fue de 0,767 y el error fue de 14,105%. La red neuronal frente a los otros modelos fue la que obtuvo menos exactitud en sus resultados. El modelo con el porcentaje más alto de exactitud fue el de regresión logística con el 89,15% (Adekitan y Salau, 2019).

En Marruecos, se desarrolló un estudio en torno a los MOOC. La razón es la alta tasa de deserción. Se utilizó una base de datos de la Universidad de Standford que comprende datos de características demográficas, calificaciones de quices, actividad en los foros, número de videos vistos, cantidad de repeticiones de los videos, de devolución y adelantos, número de descargas del material que hay en la plataforma, número de veces que el estudiante se conecta a la plataforma, el tiempo que gasta dentro de la plataforma, el número de quices 100% bien hechos, el tiempo que demora el estudiante en completar un quiz, el porcentaje de avances, el rendimiento semanal, entre otras características. Se compararon varios algoritmos de aprendizaje automático: el modelo DT (árbol de decisiones), SVM (máquina de vectores de soporte), K-NN (vecinos más cercanos) y DL (aprendizaje profundo). El modelo de DF fue el que dio unos resultados más altos de predicción con un AUC de 0,992, una exactitud de 0,965 y una precisión de 0,957 (Mourdi *et al.*, 2019).

En otro estudio realizado en Marruecos, se busca predecir el rendimiento de los estudiantes por medio de una red neuronal. Como bien se sabe, el rendimiento académico es un problema multifactorial; es decir, existen muchos factores que lo pueden afectar. Por esta razón, también se ha implementado el análisis de componentes principales para hacer una reducción de la dimensionalidad. Se propone un modelo de red neuronal multicapa (MLP) y las variables que se estudian son: secciones de estudiantes, notas de la primera prueba, notas de selección, segundas notas de prueba, número de conexiones, puntuación de la entrevista, participación de los estudiantes en el foro. Este modelo fue desarrollado en WEKA y tuvo una exactitud de predicción del 92 % (El Fouki *et al.*, 2019).

En 2020 se elaboraron dos estudios sobre la aplicación de redes neuronales para la predicción del rendimiento académico en Nigeria. Uno de ellos utilizó dos herramientas de EDM: la clasificación bayesiana y la red neuronal profunda. Las variables de entrada fueron el puntaje del examen de ingreso, la modalidad de ingreso, el número de convocatorias aplicadas. Se hicieron varios experimentos con cuatro redes neuronales diferentes. Todo indicó que las redes neuronales superan al algoritmo de clasificación bayesiana. Las redes neuronales tuvieron una precisión promedio del 83 % frente a un 71 % promedio de la clasificación bayesiana (Olaekan *et al.*, 2020).

El otro estudio implementó tres modelos: regresión lineal con aprendizaje supervisado, regresión lineal con *deep learning* y red neuronal con *deep learning*. Se evaluaron variables como edad, distribución demográfica, antecedentes familiares y actitud para estudiar. La métrica de medición fue el MAE, y la red neuronal profunda arrojó el mejor resultado: 6,23 (Oyedeki *et al.*, 2020).

En 2018, en Estados Unidos, también se hicieron varios estudios. En uno de ellos se aplicó un modelo de red neuronal tipo LSTM. Sin embargo, la red neuronal no se construyó desde cero, sino que se utilizó *network builder*, un paquete de envoltura escrito por Anthony Botelho y Seth Adjei. El uso de este modelo permitió construir un modelo de predicción personalizado con unas pocas líneas de código. Se determinó que cuatro de los factores que más pueden afectar a un estudiante son la confusión, el aburrimiento, la concentración y la frustración. Las respuestas ante las evaluaciones virtuales

son las variables que se tuvieron en cuenta. La precisión de la red neuronal fue del 87 %, por lo que los autores de este trabajo consideran que, aunque los resultados no son malos, se podrían mejorar (Gu *et al.*, 2018).

En el mismo país se hizo un trabajo para predecir la deserción de los estudiantes en MOOC, pues la modalidad de aprendizaje en línea es la que tiene más altas tasa de deserción (el 90 % de los estudiantes que están en MOOC desertan) y predecirlo ayuda a implementar estrategias para que el aprendizaje sea más personalizado. Las variables que se utilizan en este estudio son número de veces que los estudiantes ven los anuncios, número de veces que los estudiantes acceden a las tareas, número de veces que los estudiantes ven el calendario, número de veces que los estudiantes acceden a las páginas del módulo, número de veces que los estudiantes acceden a los cursos, número de veces que los estudiantes acceden al foro de discusión, número de veces que los estudiantes acceden a los archivos, número de veces que los estudiantes revisan los libros de calificaciones, número de veces que los estudiantes acceden a las pruebas, número de veces que los estudiantes envían tareas, número de veces que los estudiantes acceden a las páginas de wikis, número de días que el estudiante interactúa con el curso y la semana en la que el alumno visita el curso por última vez.

Se estudian varios modelos: modelo de predicción de abandono a plazo fijo (que asume desde un principio que todos los estudiantes van a desertar) y modelo de rendimiento de predicción. El modelo con un plazo fijo de tiempo no puede determinar los estudiantes que necesitan apoyo inmediato, solo puede predecir con éxito los resultados de la primera semana, mas no de las siguientes semanas. Los modelos de rango temporal (predicciones por semana) son más acertados, pero de igual manera no fue posible personalizar la intervención del instructor. El beneficio que brinda es que pone en evidencia a aquellos que tienen más probabilidad de seguir o de desertar que otros estudiantes. En cuanto a la optimización del rendimiento del modelo, se han aplicado varios algoritmos, pero se cree que la eficiencia se puede ver mejorada con redes neuronales. Primero se implementan estos algoritmos: K-vecinos más cercanos (KNN), máquinas vectoriales de soporte (SVM) y árbol de decisión (DT). Luego, se desarrolla un modelo de red neuronal profunda (MLP) para comparar el rendimiento de los anteriores modelos



con la red neuronal. Por último, para evitar el sobreajuste de la red neuronal, se hace la validación cruzada de diez veces. En ese momento se miden los rendimientos de varias rondas de validación cruzada y promediado durante las rondas (Xing y Du, 2019).

Kim *et al.* (2018a) very few prior studies have explored this problem from a deep learning perspective. In this paper, we recast the student performance prediction problem as a sequential event prediction problem and propose a new deep learning based algorithm, termed GritNet, which builds upon the bidirectional long short term memory (BLSTM) buscar predecir cuáles estudiantes de Udacity se gradúan y cuáles no. Para ello, se utiliza un algoritmo de *deep learning* llamado Gritnet, que se basa en la memoria a corto plazo bidireccional (BLSTM). Gritnet es un algoritmo basado en redes neuronales que ayuda a mejorar el rendimiento de un modelo de regresión lineal. En este caso, el modelo fue mejorado en un 7,7%. Cualquier dato que se obtiene con las interacciones que se conocen del alumno en la plataforma se toman como variables de entrada. Este modelo no se evalúa con precisión, sino con ROC.

El otro modelo utiliza una red neuronal *backpropagation*. Este modelo consta de tres capas ocultas. Se usó R para su programación, se usó el paquete mxnet y RELU como función de activación. Las variables de entrada fueron el registro del rendimiento académico del estudiante. El 90% de los datos se destinaron para entrenamiento y el 10% para pruebas. La precisión en este modelo fue del 87% (Özkan *et al.*, 2018).

En 2019, en Estados Unidos, se trata de buscar una solución a la alta tasa de deserción en el aprendizaje en línea. Para ello, se propone predecir la deserción en los MOOC con modelos de aprendizaje automático tradicionales y modelos de aprendizaje automático profundo. Se implementan algoritmos como el modelo oculto de Markov, bosques aleatorios, *Gradient Boosting Decision Tree* (GBDT), XGBoost y un modelo de red neuronal llamado GRU-RNN. Las variables de entrada fueron el tiempo total en línea del estudiante, el promedio de tiempo por sesión, el intervalo de tiempo promedio entre un cierre de sesión y un inicio de sesión, el tiempo total de los intervalos entre inicios de sesión, la duración de un estudiante cuando ve un video, la cantidad de tiempo de un estudiante para desarrollar un ejercicio,

el número de participaciones en los foros y el tiempo de respuesta promedio. La precisión del modelo de red neuronal fue superior al resto de modelos con valores entre 70 % y 85 % (Sun *et al.*, 2019).

Otro estudio realizado también en Estados Unidos busca detectar tempranamente comportamientos en el estudiante que lo podrían conducir a ser un posible desertor. Este trabajo infiere que la perseverancia de un alumno no es lo mismo que persistencia académica, pues la persistencia puede ser improductiva. Se implementó un modelo de red neuronal LSTM realizado en Python, cuyas variables de entrada fueron la Interacción con el entorno virtual de aprendizaje y los datos registrados en el sistema. La efectividad de la predicción se midió en AUC y en RMSE. Los valor máximo de la red neuronal fue 0,6 para AUC, y el valor menor de RMSE fue de 0,221 (Botelho *et al.*, 2019).

En este mismo país, se hizo un estudio que se enfoca en tratar de predecir el rendimiento académico de un estudiante en un nuevo programa (sin datos históricos), de acuerdo con su rendimiento en otro programa. La población objetivo son los estudiantes de Udacity. Se trata de un trabajo que también se enfoca en el sector de los MOOC. Se propuso un modelo de red neuronal no supervisado llamado Gritnet 2 que pretende mostrar el rendimiento académico en tiempo real, teniendo en cuenta el comportamiento del estudiante en otros cursos y, antes de que suceda, predecir si el estudiante va a abandonar el curso o va a seguir. Las variables que se tienen en cuenta son las que se generan durante la interacción en la plataforma. Esta red neuronal arrojó una precisión promedio del 70 % (Kim *et al.*, 2018b).

En 2020, un estudio realizado en Canadá estudia qué tan viable resulta la idea de desarrollar un modelo predictivo general, es decir, que sea aplicable no solo a un curso o a una carrera, sino a varios escenarios. Para ello, se basan en los datos arrojados por el sistema de gestión del aprendizaje (LMS) y tratan de determinar atributos o actividades comunes en diferentes cursos o carreras, así como también, algoritmos que se desempeñen mejor en dicha generalización. El modelo de la red neuronal tuvo un AUC promedio del 75 %. Se concluyó que, aunque los modelos generales no muestran malos resultados, no se recomienda hacer generalizaciones pues cada método disminuye su efectividad (Cui *et al.*, 2020).



A nivel Latinoamérica, en 2018 se lleva a cabo un estudio en la Universidad Central del Ecuador. Se diseña un modelo predictivo del rendimiento académico de los estudiantes con base en redes neuronales. La información utilizada son los datos generados en la plataforma virtual de aprendizaje; se evalúan las siguientes variables: número total de clics, número de consultas al profesor, cantidad de sesiones en línea, participación en los foros, tiempo total en línea (min), participación en chat, número de visitas a la página del curso, participación en videocolaboración, número de comentarios, número de recursos vistos, irregularidad del tiempo de estudio, número de enlaces vistos, irregularidad del intervalo de estudio, número de páginas de contenido vistas, mayor periodo de inactividad, número de publicaciones de discusión vistas, tiempo hasta la primera actividad (min), número total de publicaciones de discusión, tiempo promedio por sesión (min), número de cuestionarios iniciados, número de intentos por prueba, número de asignaciones enviadas, número de cuestionarios vistos, número de presentaciones vistas, número de cuestionarios pasados, número de ediciones en la wiki, número de wiki vistas y grado promedio de evaluación. Se determinó que el grado del cuestionario fue la característica más relevante, con un 12,4%; la regularidad del estudio y el promedio de clics por sesión fueron los siguientes mejores predictores, con un 12,4% y un 10,9%, respectivamente. El modelo se aplicó a cada uno de los cursos y la exactitud osciló entre el 58% y el 91% (Reyes *et al.*, 2018).

En 2020, en una universidad de Argentina se indagó acerca de métodos de predicción temprana de la deserción académica con herramientas de *Machine Learning* que pudieran aplicarse al campo del aprendizaje en línea, en cursos de entre 6 y 16 semanas de duración. Se tomaron como variables de entrada los datos registrados en la plataforma virtual Moodle, durante las primeras dos semanas de los cursos. En cuanto a los modelos implementados, se utilizaron algunos del paquete CORELearn: random forest, KNN, random forest con KNN en las hojas con el modelo RFNear y KNN con Bayes ingenuo. Se trabajó también el paquete rpart con los métodos Anova, Class y Poisson para algoritmos de árboles de decisión. Se utilizó el paquete NeuralNet con los modelos rprop+ y rprop- para algoritmos de redes neuronales. Las redes neuronales tenían diez capas

ocultas y tuvieron una precisión promedio de 28 % superando a otros modelos (Urteaga *et al.*, 2020).

La búsqueda de literatura muestra una tendencia de crecimiento de investigación en cuanto a la aplicación de las redes neuronales artificiales al contexto educativo. Según esta búsqueda, los lugares en los que más se ha tenido participación ha sido en países asiáticos y europeos, aunque se han extendido este tipo de investigaciones poco a poco al resto del mundo. Se puede notar cómo el *deep learning* es una rama con un crecimiento incipiente desde este último siglo. Aún hace falta investigar más este campo en varias partes del mundo, sobre todo teniendo en cuenta que, al ser utilizado para predecir el rendimiento académico, las conclusiones a las cuales se han llegado hasta ahora es que no es recomendable diseñar un modelo generalizado sino un modelo para cada uno de los casos, pues el rendimiento académico está influenciado por múltiples variables que cambian de acuerdo con el entorno en el que se esté realizando.

En el caso de Colombia, al parecer, el uso de redes neuronales artificiales aplicadas a la predicción del rendimiento o la deserción académica es un tema inexplorado. Si bien se realizó una búsqueda en Kujane, que es un repositorio de material académico y científico en el que colaboran diferentes universidades del país, no se registró ningún resultado que relacionara las redes neuronales artificiales con el ámbito académico. Sin embargo, se encontró un trabajo de maestría elaborado en 2017 en la Universidad Nacional en el que se hace uso de otro tipo de algoritmos de EDM como árboles de decisión y Naive Bayes para predecir la deserción académica. Este trabajo hace énfasis en cómo los programas de bienestar institucional pueden aportar a este tipo de investigaciones pues apoyan el desarrollo físico, psicoafectivo, espiritual y social de los estudiantes (Cuadrado Montaña, 2017).

También se hizo una búsqueda en los repositorios de la Universidad Distrital Francisco José de Caldas, y se encontró solo el registro de un artículo (Castro *et al.*, 2016) en la revista científica CIDC (de esta universidad). En el artículo se utilizaba un algoritmo de árbol de decisión, que es otra técnica de EDM para construir un modelo predictivo de deserción académica. El campo del *deep learning* en la educación no se ha explorado lo suficiente ni a nivel nacional ni internacional. Hay otras técnicas de EDM que, al ser parte

de modelos clásicos de *Machine Learning* se han estudiado un poco más. Se podría tomar información valiosa de estos otros modelos para aplicarlos a las redes neuronales artificiales.

Un ejemplo de ello es un estudio elaborado en 2016 en Emiratos Árabes Unidos. A pesar de que Abu (2016) no implementa modelos de red neuronal, sino que implementa otros modelos de algoritmos de EDM como árbol de decisión, clasificador bayesiano y algoritmo CART, también busca predecir el rendimiento de los estudiantes y hace una hallazgo que vale la pena tener en cuenta: al utilizarse variables cualitativas, como por ejemplo personalidad del estudiante y motivación, se descubre que estas variables tienen un gran impacto en los resultados de la predicción. Estas variables se podrían implementar también en modelos de redes neuronales para mejorar los resultados de dichos modelos.

## **11.4. Análisis de las producciones académicas relacionadas con redes neuronales**

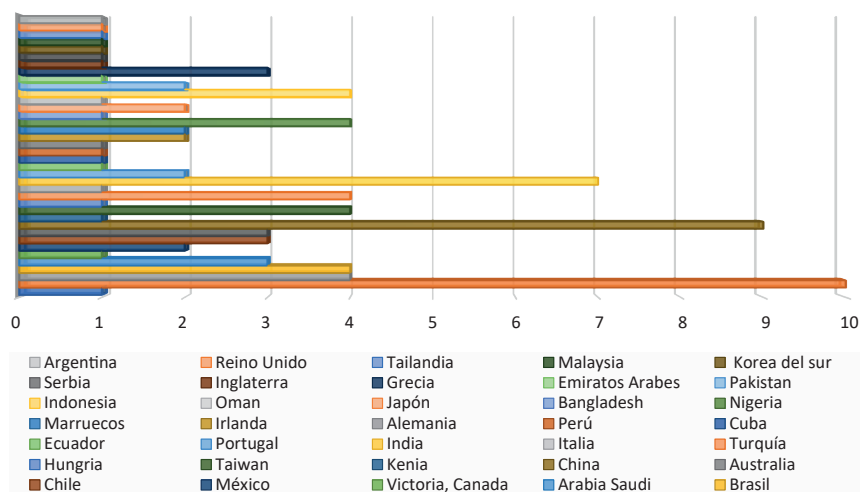
Un aspecto que vale la pena resaltar luego de la breve descripción y el análisis de cada uno de los artículos correspondientes al periodo 2010-2020, es la identificación de cuáles son los países que están trabajando esta parte de la temática por medio de sus publicaciones en revistas científicas de alto impacto. En la figura 78 se presenta el diagrama de barras de las investigaciones del sector educación y su país de publicación.

Con base en la figura 78, se puede decir que las investigaciones en aprendizaje profundo aplicado al ámbito de la educación se centran en Estados Unidos, China y la India, con una cantidad de artículos encontrados de 10, 9 y 7, respectivamente. En el resto de los países se publicaron entre uno y cuatro trabajos de investigación sobre este tema en la última década.

Los países con mayor número de proyectos investigativos (figura 79) son los países con las economías más importantes del mundo. Este factor que puede resultar significativo para el crecimiento y el fortalecimiento de la educación de un país, pues al existir una economía estable, existen mayores posibilidades de invertir en recursos tanto para la investigación como para la innovación. Como ya se ha mencionado, uno de los mayores retos en el

ámbito educativo a nivel mundial es disminuir la deserción estudiantil. Las instituciones educativas se han valido de herramientas como la estadística y el *Machine Learning* clásico para tratar de encontrar una forma acertada de pronosticar el rendimiento del estudiante y evitar la deserción. Sin embargo, gracias al surgimiento de nuevas tecnologías, el aprendizaje automático o *Deep Learning* ha ganado terreno en el ámbito de la educación; este conocimiento permite estudiar variables y conceptos que no son visibles en los métodos tradicionales.

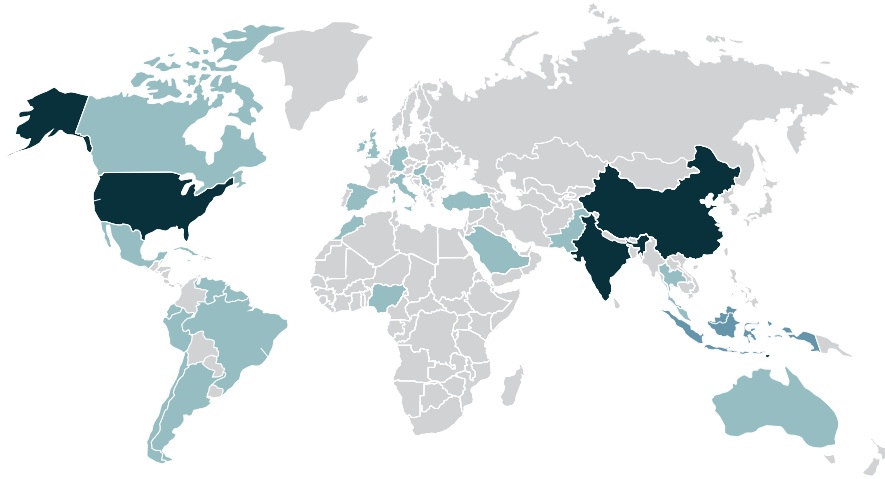
**Figura 78.** Investigación referencial y su representación en el mundo



**Fuente:** elaboración propia.

No obstante, para su implementación se debe tener en cuenta todo lo que implica, desde la inversión en conocimiento y el desarrollo de procesos, hasta la adaptación de nueva infraestructura tecnológica. Es en este sentido que existen evidencias de ventajas que tienen algunos países desarrollados frente a otros países; por ejemplo, los países desarrollados usualmente cuentan con bancos de datos públicos, un fenómeno que facilita la recolección de información. Sin embargo, para que esto sea posible en otros países, se tendría que entrar en un proceso “anonimización”, tal y como lo indica Prenkaj *et al.* (2020), lo que a su vez se traduce en costos que algunos países no han contemplado como una estrategia para mejorar la educación.

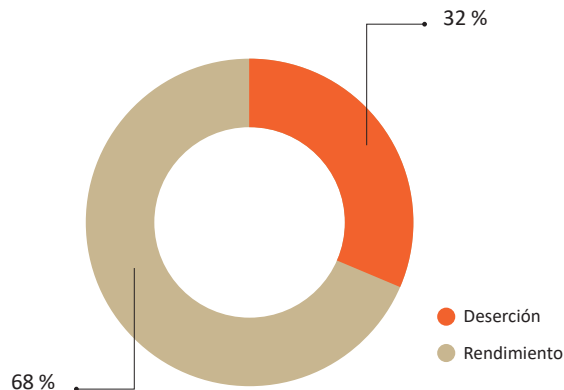
**Figura 79.** Mapa de calor geográfico de investigaciones sobre aprendizaje no supervisado



**Fuente:** elaboración propia.

Una de las características más importantes a la hora de realizar una clasificación de los trabajos es el enfoque académico de cada estudio. Se decidió separar los estudios en dos grupos: del primero forman parte los artículos que hablan sobre deserción (28 artículos); y del segundo forman parte los artículos que se centran en el rendimiento académico (63 artículos). Por otro lado, ambos tipos de artículos se centran en el uso de herramientas de *Machine Learning* (figura 80).

**Figura 80.** Clasificación según enfoque



**Fuente:** elaboración propia, basado en el estado del arte.



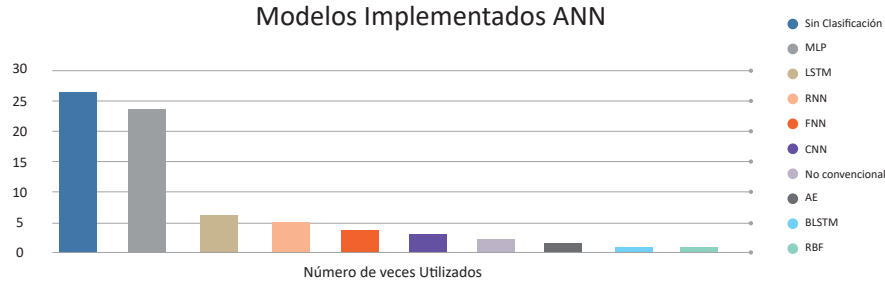
# Capítulo 12

## Redes neuronales: algoritmos implementados

El *deep learning* (aprendizaje profundo), es un área de investigación subcampo del *Machine Learning*. Su foco es el uso de redes neuronales para modelar el más alto nivel de abstracción en los datos (Hernández-Blanco *et al.*, 2019). La palabra *deep* en *aprendizaje profundo* se refiere a los distintas capas de transformación de la información entre las entradas y las salidas de la red neuronal (Hernández-Blanco *et al.*, 2019); hablamos de redes con varias capas ocultas. Según Hernández-Blanco *et al.* (2019), el éxito del *Deep Learning* (DL) es que se trata de un método que evita el paso de selección de la información representativa, lo cual ahorra tiempo y permite encontrar representaciones necesarias para una tarea en cuestión. Según el mismo estudio, DL es el área de investigación de mayor tendencia en el campo de la inteligencia artificial para el 2019.

Actualmente existen múltiples algoritmos clasificados como propicios para el aprendizaje profundo, así como múltiples autores que combinan modelos o técnicas de aprendizaje supervisado, no supervisado y aprendizaje profundo. En estos casos, las técnicas se ejecutan de forma paralela para medir el rendimiento y así poder optimizar su utilización en trabajos posteriores o similares. A continuación (figura 81), se muestra cada una de las técnicas de aprendizaje profundo: *deep neural network* (DNN), *convolutional neural network* (CNN), *probabilistic neural network* (PNN), *multilayer perceptrón* (MLP), *feed forward neural network* (FNN), *recurrent neural network* (RNN), *memory networks* (MN) y *fuzzy genetic algorithm*.

**Figura 81.** Algoritmos de aprendizaje profundo (redes) utilizadas por las investigaciones consultadas



**Fuente:** elaboración propia.

Según Hernández-Blanco *et al.* (2019), dependiendo el tipo de entrada de la red neuronal, hay diferentes arquitecturas de redes neuronales más adecuadas para procesar la información (tabla 42). Por ejemplo, dependiendo de si las entradas son de texto, audio, imágenes, etc., cambia la técnica. A continuación, un resumen de algunas de las clasificaciones de arquitecturas de DL recopiladas por Cuevas-Tello (2018) y Hernández-Blanco *et al.* (2019):

**Tabla 42.** Algunas arquitecturas de DL

| Arquitectura                           | Siglas |
|----------------------------------------|--------|
| Perceptrón multicapa                   | MLP    |
| Memoria a corto plazo                  | LSTM   |
| Word Embeddings                        | WE     |
| Redes neuronales convolucionales       | CNN    |
| Red neuronal de retroalimentación      | FNN    |
| Red neuronal recurrente                | RNN    |
| Codificador automático (auto-enconder) | AE     |
| LSTM bidireccional                     | BLSTM  |
| Redes de memoria                       | MN     |

**Fuente:** basado en Hernández-Blanco *et al.* (2019).

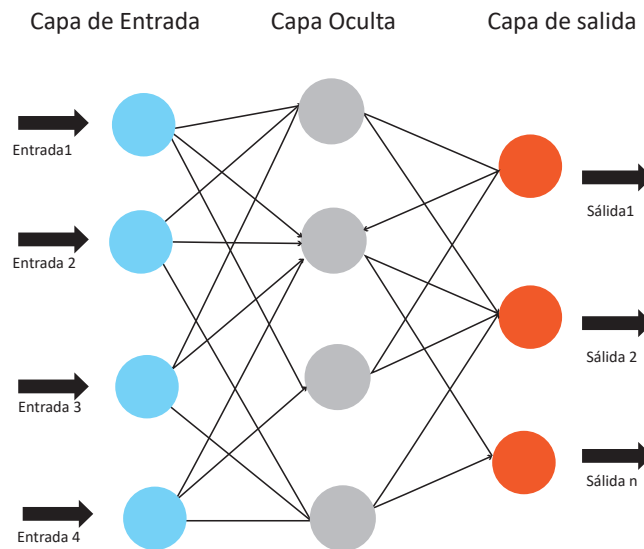
### 12.1. Red Neuronal Perceptrón Multicapa (MLP)

El modelo se basa en “un conjunto de unidades neuronales simples (neuronas artificiales), de forma aproximadamente análoga al comportamiento



observado en los axones de las neuronas en los cerebros biológicos” (Ruby y David, 2015). Este modelo computacional está conformado por un número de unidades computacionales básicas (neuronas) que están conectados entre sí y que forman una red de comunicación que permite realizar computaciones complejas; de tal manera, una neurona se conecta con otras anteriores y posteriores, y estas con muchas otras en diferentes capas. Los datos ingresan al modelo por medio de la capa de entrada, se intercomunican por medio de las capas ocultas y salen por la capa de salida (figura 82).

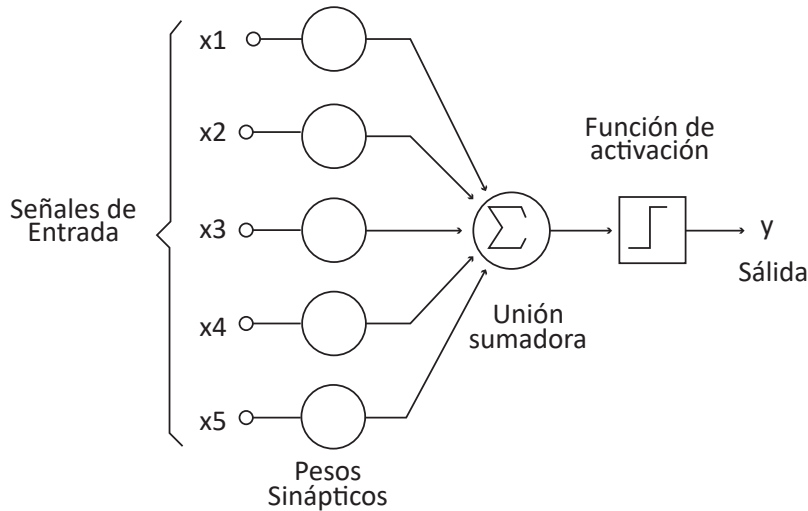
**Figura 82.** El perceptrón multicapa con una sola capa oculta



**Fuente:** elaboración propia.

**Perceptrón:** Este tipo de red (figura 83) tiene como característica que una neurona acepta señales de entrada ( $n$  entradas) a través de sus dendritas (datos de entrenamiento), y transmiten la señal eléctrica al cuerpo celular (neurona); estas señales son pesadas y se combinan en una ecuación lineal (función de activación) para proporcionar una respuesta (variable de salida).

**Figura 83.** Ejemplo de perceptrón



Fuente: Wikipedia (2019).

- **Perceptrón multicapa (MLP):** Este tipo de red comprende una capa de entrada y varias capas intermedias y una capa de salida. Surgió como consecuencia de las limitaciones Del perceptrón simple a la hora de clasificar conjuntos de datos que no eran linealmente separables. Esta red recibe las señales de entrada desde otra capa anterior y envía señales de salida a una capa posterior. Por ese motivo, se les denomina *de conexión hacia adelante* (feedforward). Este tipo de red emplea el algoritmo de aprendizaje de retropropagación: la red calcula los pesos entre capas a partir de valores aleatorios y hace cambios graduales y progresivos posterior a una evaluación de errores de las salidas hasta que los resultados converjan a una aproximación de error aceptable (Polo *et al.*, 2015).

El elemento básico de una red neuronal es un nodo o capa que actúa en paralelo con otros nodos o capas de la red. Los nodos aceptan valores de entrada de otros nodos y proporcionan una salida. La primera tarea de la capa es procesar los datos de entrada para crear un valor resumen que es la suma de todas las entradas multiplicadas por sus ponderaciones. Este valor resumen se procesa a continuación mediante una función de activación para generar una salida que se envía a la siguiente capa del sistema (red). Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 43.

**Tabla 43.** Autores consultados que han usado el algoritmo MLP durante la última década

| Algoritmo | Referencias bibliográficas            |
|-----------|---------------------------------------|
| MLP       | (Doleck <i>et al.</i> , 2020)         |
|           | (Miranda y Guzmán, 2017)              |
|           | (Tsai <i>et al.</i> , 2020)           |
|           | (Álvarez Blanco <i>et al.</i> , 2016) |
|           | (Menacho Chiok, 2017)                 |
|           | (Xing y Du, 2019)                     |
|           | (Hussain <i>et al.</i> , 2020)        |
|           | (El Fouki <i>et al.</i> , 2019)       |
|           | (Pavithra y Dhanaraj, 2019)           |
|           | (Zulfiker <i>et al.</i> , 2020)       |
|           | (Arumugam <i>et al.</i> , 2019)       |
|           | (Huang y Fang, 2013)                  |
|           | (Usman y Adenubi, 2013)               |
|           | (Zacharis, 2016)                      |
|           | (Jayne <i>et al.</i> , 2011)          |
|           | (Rivas <i>et al.</i> , 2021)          |
|           | (Sokkhey y Okazaki, 2019)             |
|           | (Guo <i>et al.</i> , 2015)            |
|           | (Costa <i>et al.</i> , 2017)          |
|           | (Cui <i>et al.</i> , 2020)            |
|           | (Gkontzis <i>et al.</i> , 2018)       |
|           | (Olalekan <i>et al.</i> , 2020)       |
|           | (Alshabandar <i>et al.</i> , 2020)    |
|           | (Urteaga <i>et al.</i> , 2020)        |

Fuente: elaboración propia.

## 12.2. Red Neuronal de Realimentación (FNN)

La Red Neuronal de Realimentación (FNN) representa la entrada en juego de la primera generación de red neuronal. Entre las características de esta

red se encuentra que la información siempre se propaga hacia adelante o, mejor dicho, desde los nodos de la primera capa hasta los nodos de la última capa. Unos claros ejemplos de este tipo de red son el perceptrón y el perceptrón multicapa MLP Hernández-Blanco *et al.* (2019). Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 44.

**Tabla 44.** Autores consultados que han usado el algoritmo FNN durante la última década

| Algoritmo | Referencias bibliográficas          |
|-----------|-------------------------------------|
| FNN       | (Lykourantzou <i>et al.</i> , 2009) |
|           | (Paliwal & Kumar, 2009)             |
|           | (Patil <i>et al.</i> , 2017)        |
|           | (Altaf <i>et al.</i> , 2019)        |

**Fuente:** elaboración propia.

### 12.3. Red Neuronal Convolutacional (CNN)

Son una variación del MLP. Estas redes son útiles principalmente en el procesamiento de imágenes (Hernández-Blanciencio *et al.*, 2019) por lo que no se tomarán en cuenta. Además tienen las variantes VGG16 y AlexNet que son arquitecturas para análisis de video (Hernández-Blanco *et al.*, 2019). Dicho de otra forma, una red neuronal convolutacional es un tipo de red neuronal derivada del perceptrón multicapa a la cual se le entrena para que aprenda a reconocer patrones en imágenes y video. Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 45.

**Tabla 45.** Autores consultados que han usado el algoritmo CNN durante la última década

| Algoritmo | Referencias bibliográficas     |
|-----------|--------------------------------|
| CNN       | (Agrusti <i>et al.</i> , 2020) |
|           | (Wu <i>et al.</i> , 2019)      |
|           | (Xing y Du, 2019)              |
|           | (Qiu <i>et al.</i> , 2019)     |

**Fuente:** elaboración propia.

## 12.4. Red Neuronal Recurrente (RNN)

Es una red neuronal cuya estructura permite conexiones que realimenten una capa anterior o la misma capa; esto significa que las salidas de una capa pueden alimentar una capa anterior o a la misma capa en cuestión, pero, a diferencia de una FNN, esta estructura permite a la red tener memoria. Sus aplicaciones están enfocadas en el modelado de texto o reconocimiento de voz, pero han sido utilizadas también en el contexto de EDM para predecir deserción estudiantil (Hernández-Blanco *et al.*, 2019). Se clasifican según su mecanismo de retroalimentación, ya que esta puede presentarse en capas ocultas o en la de salida. Una posible clasificación de estas redes, por ejemplo, podría darse en cuanto a la memoria a corto plazo o a las redes de memoria, teniendo en cuenta que estas últimas están diseñadas para abordar el problema de aprendizaje automático para tendencias a largo plazo (Hernández-Blanco *et al.*, 2019). Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 46.

**Tabla 46.** Autores consultados que han usado el algoritmo RNN durante la última década

| Algoritmo | Referencias bibliográficas   |
|-----------|------------------------------|
| RNN       | (Sun <i>et al.</i> , 2019)   |
|           | (Kim <i>et al.</i> , 2018)   |
|           | (Okubo <i>et al.</i> , 2017) |
|           | (Corrigan y Smeaton, 2017)   |
|           | (Shah <i>et al.</i> , 2019)  |

**Fuente:** elaboración propia.

## 12.5. Bidirectional Long Short-Term Memory (BLSTM)

Es un tipo de red neuronal similar a la red recurrente o a la LSTM, por sus siglas en inglés. Básicamente es una estructura de red neuronal enfocada en la capacidad de la red de tener memoria, y específicamente en tener memoria a largo plazo. En este caso, el flujo de la información es bidireccional. En otras palabras, la BLSTM (bidirectional long short term memory) es una red que puede emplear resultados posteriores para ajustar sus predicciones pasadas. Se ha empleado en la predicción del rendimiento de estudiantes, y

a diferencia de LSTM, no solo conserva información pasada, sino que ejecuta la información del pasado al futuro y del futuro al pasado (Hernández-Blanco *et al.*, 2019). Esto implica que comportamientos actuales de mis datos pueden corregir o afectar los parámetros de la red; así, esos comportamientos pueden afectar previos resultados en un contexto que implique un amplio rango de tiempo. Los diferentes autores que utilizaron este tipo de algoritmo se enumeran en la tabla 47.

**Tabla 47.** Autores consultados que han usado el algoritmo BLSTM durante la última década

| Algoritmo | Referencias bibliográficas |
|-----------|----------------------------|
| BLSTM     | (Kim <i>et al.</i> , 2018) |

**Fuente:** elaboración propia.

## Capítulo 13

# Análisis de *software* usados/relacionados con *Machine Learning* aplicado al rendimiento académico en educación superior

En este capítulo se presenta una amplia variedad de *software* y lenguajes de programación utilizados por los diferentes autores para someter a prueba los algoritmos mencionados en capítulos anteriores. El fin de este capítulo es exponer los *softwares* más utilizados en la última década. Cada uno de ellos es explicado de una forma que sea fácil de entender para los lectores de este documento, ya que no solo está redactado para el personal del área de la ingeniería, sino para docentes y estudiantes que deseen profundizar en el tema de la analítica de datos en el campo educativo.

### 13.1. Breve resumen de *software* o lenguajes de programación

- Python

Es un lenguaje de programación creado a principio de la década de los noventa por Guido Van Rosum. Es un lenguaje interpretado o de script, fuertemente tipado, con tipado dinámico, orientado a objetos y de múltiple plataforma, cuyo nombre está inspirado en el grupo de cómicos ingleses “Monty Python”. Según Gonzalez (2010) los lenguajes interpretados o de script, como Python, se ejecutan directamente con un lenguaje

interprete (lenguaje compilado), que tiene ventajas frente a otro tipo de lenguajes en cuanto a la rapidez de ejecución, la flexibilidad y la portabilidad; la característica de tipado dinámico es una ventaja debido a que no es necesario declarar el tipo de variable que se pretende ejecutar. Sin embargo, como en cualquier lenguaje de programación, es imposible realizar operaciones o formular con variables de diferente tipo (fuertemente tipado).

En los últimos años, el lenguaje se ha hecho muy popular por razones como las que se enumeran a continuación:

- La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje; estos elementos ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.
- La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de tres a cinco líneas de código menos que su equivalente en Java o C.
- La cantidad de plataformas en las que podemos desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otros.

Python es un lenguaje de programación que se destaca por su código legible y limpio. Una de las razones de su éxito es que cuenta con una licencia de código abierto que permite su utilización en cualquier escenario. Actualmente, Python cuenta con múltiples plataformas de ejecución dentro de las que se incluyen Windows, OS/2, Mac OS, UNIX, Solaris, Linux, DOS, entre otros. A través de un paradigma de programación, se transforman en objetos y clases que se interrelacionan; aunque también se manejan códigos en Python que son de programación imperativa, programación orientada a aspectos y programación funcional.

- **WEKA (Waikato Environment for Knowledge Analysis)**

Según Interoperability (2018), WEKA es un popular *software* de *Machine Learning* escrito en lenguaje de programación Java y desarrollado por la Universidad de Waikato, Nueva Zelanda, en 1993, Las características principales del *software* son una interfaz gráfica muy amigable para acceder a las



diferentes funcionalidades y una colección de herramientas de visualización y algoritmos para análisis de datos y modelado predictivo. Es ampliamente utilizado para la enseñanza, la investigación y las aplicaciones industriales. WEKA comprende una gran cantidad de herramientas incorporadas para las tareas de aprendizaje automático estándar, y además da acceso transparente a herramientas bien conocidas como scikit-learn, R, y Deeplearning4j (*Weka 3 - Data Mining with Open Source Machine Learning Software in Java*, 2018).

Weka está diseñado para soportar múltiples tareas estándar de minería de datos: *clustering*, regresión, clasificación, selección, visualización y pre-procesamiento de datos conectado a bases de datos vía SQL gracias a la colección Java Database Connectivity (JDBC). Actualmente, con este *software* es imposible realizar minería de datos multirelacional y no cubre algoritmos de modelado de secuencias. Se trata de un *software* libre bajo la licencia publica general de GNU en las versiones 3.6.15 (book), 3.8.3 (stable), 3.9 (development). (Singhal y Jena, 2013).

- **R programming**

Según Santana y Farfán (2014), se trata de un lenguaje de programación interpretado, de tipo estadístico y gráfico. Este *software* corre en distintas plataformas como Windows, MacOS, Linux, entre otros; cuenta con licencia pública de distribución libre bajo Licencia GN, es decir, con R se tiene la libertad de ejecutarlo para cualquier propósito, estudiar cómo trabaja el programa y adaptarlo a sus necesidades —se tiene acceso al código fuente—, redistribuir copias, y mejorar el programa y liberar sus mejoras al público en general.

R fue creado en 1992 en Nueva Zelanda por Ross Ihaka y Robert Gentleman como una adaptación a la sintaxis del lenguaje S desarrollado por Bell Laboratories (*Introducción a estadística con R*, 2018). A modo de broma, Ross y Robert comienzan a llamar “R” al lenguaje que implementaron, por las iniciales de sus nombres y, desde entonces, así se le conoce en la muy extendida comunidad amante de dicho lenguaje. R es una evolución de “S” (Santana y Farfán, 2014).

Se compone del sistema base y demás funciones agrupadas en la librería y la interfaz. El sistema base contiene, como su nombre lo indica, las

herramientas necesarias para su ejecución, compilación e impresión de resultados. Algunos paquetes contenidos de la “base” del sistema son útiles, *stats*, *datasets*, *graphics*, *grDevices*, *grid*, *tools*, *parallel*, *compiler*, *splines*, *tcltk*, *stats4*. También existen paqueterías que permiten manipular y crear datos en distintos formatos como *netCDF*, *Matlab*, *Excel*, entre otros. Según Santana y Farfán (2014), R es muy útil para el trabajo interactivo, pero también es un poderoso lenguaje de programación para el desarrollo de nuevas herramientas, por ejemplo *rclimindex*, *cliMTA-R*, etc.

- **Java**

Según Oracle (2020), Java es una plataforma y un lenguaje de programación informático presentado públicamente en 1995 por Sun Microsystems, apoyado por la comunidad Oracle Technology Network —quienes ajustan constantemente la arquitectura del *software* para brindar a todos los usuarios múltiples aplicaciones y servicios—. Es actualmente la base más grande de aplicaciones de red junto con el estándar global para desarrollar y distribuir aplicaciones móviles, juegos, *software* de empresa y contenido basado en web. Cuenta con un sistema de conexión de múltiple plataforma tanto para escribir *software* como para ejecutarlo, junto con aplicaciones potentes y eficaces para teléfonos móviles, procesadores remotos, microcontroladores, módulos inalámbricos, sensores, *gateways*, productos de consumo y prácticamente cualquier otro dispositivo electrónico. Su descarga es gratuita y la versión más reciente se encuentra en la extensión Java Runtime Environment.

- **SQL (Structured Query Language)**

Según la Universidad Nacional de Colombia (2019), este es un lenguaje de programación orientado a consultas de bases de datos. Su desarrollo comprende múltiples variantes y herramientas como *MySQL*, *Microsoft SQL Server*, *MySQL Workbench*, *Heidi SQL*, *Sequel Pro*, *PostgreSQL*, entre otros, que actúan como un sistema de gestión de bases de datos (SGBD). Fue desarrollado en 1970 en los laboratorios de IBM como un lenguaje para gestionar los datos almacenados en el sistema R. Posteriormente, en 1979, la compañía *Relational Software* (Actualmente Oracle) saca una versión modificada llamada Oracle V2, actualmente *MySQL*, que es propiedad total de Oracle. Este lenguaje se encuentra certificado por ANSI e ISO, y se ha

convertido hoy en día en la base de una gran variedad de aplicaciones en internet. Su uso es altamente demandado en ambientes empresariales, así como también para necesidades académicas y funciona tanto en equipos individuales como en servidores de empresa (Universidad Internacional de Valencia, 2016).

- **Matlab**

Según Poddar y Mukhopadhyay (2020), Matlab es un *software* matemático con lenguaje de programación propio de alto nivel diseñado para ingenieros y científicos, y desarrollado por Cleve Moler a través de MathWorks, en 1984. Combina un entorno de escritorio perfeccionado para el análisis iterativo y los procesos de diseño con un lenguaje de programación, con el fin de procesar señales e imágenes, sistemas de comunicación, comunicaciones inalámbricas y finanzas computacionales, así como asuntos relacionados a la robótica, *deep learning*, inteligencia artificial, entre otros. Actualmente es una red de aplicaciones diseñadas por sus colaboradores con fines académicos, disponibles en la red GitHub y los sitios web oficiales de MatLab. Sus uso se extiende a sistemas empresariales, trabajo en la nube, editor de interfaces de usuario (GUIDE y plataformas de simulación, Simulink: plataforma de simulación multidominio) que logran ejecutarse en dispositivos embebidos. Está disponible para las plataformas Unix, Windows, macOS y GNU/Linux. (MathWorks - Fabricantes de MATLAB y Simulink, 2018). La versión más reciente de MATLAB es la R2020a, publicada el 19 de marzo del 2020, la cual incluye todos los aplicativos de ML creados recientemente.

- **C programming**

C es un lenguaje de programación de propósito general, originalmente desarrollado por Dennis Ritchie, entre 1969 y 1972, en los Laboratorios Bell, como evolución del anterior lenguaje B, y a su vez basado en BCPL (lenguaje de programación básico combinado). Es un lenguaje orientado a la implementación de sistemas operativos, concretamente Unix, considerado de bajo nivel de programación dada su cercanía con el lenguaje máquina. C es apreciado por la eficiencia del código que produce y es el lenguaje de programación más popular para crear *software* de sistema orientado a procedimientos y estructuras, aunque también se utiliza para crear aplicaciones. Era

el lenguaje básico para escribir todo, desde sistemas operativos (Windows y muchos otros) hasta programas complejos como la base de datos Oracle, Git, intérprete de Python y más (Moss *et al.*, 2018).

- **Otros**

- **Rapidminer:** RapidMiner es una herramienta de minería de datos ampliamente usada y probada a nivel internacional en aplicaciones empresariales, de gobierno y academia. Implementa más de quinientas técnicas de preprocesamiento de datos, modelación predictiva y descriptiva, métodos de prueba de modelos, visualización de datos, etc. (Mycosystem Información Inteligente, 2019). En una encuesta realizada por *KDnuggets*, un periódico de minería de datos, RapidMiner ocupó el segundo lugar en herramientas de analítica y de minería de datos utilizadas para proyectos reales en 2009, y fue el primero en 2010. Permite el desarrollo de procesos de análisis de datos mediante el encadenamiento de operadores a través de una GUI (Interfaz Gráfica de Usuario) o desde otros programas a través de APIs de Java. Es de código abierto y es fácil de manejar debido a su interfaz gráfica (Moss *et al.*, 2018).
- **Knime:** Es una plataforma de análisis, generación de informes e integración de datos de código abierto y gratuita que integra varios componentes para el aprendizaje automático a una interfaz gráfica de usuario para modelado, análisis de datos y visualización sin o con una programación mínima. El desarrollo comenzó en la Universidad de Konstanz. El objetivo inicial era crear una plataforma de procesamiento de datos modular, altamente escalable y abierto que permitiera una fácil integración de diferentes módulos de carga de datos, procesamiento, transformación, análisis y exploración visual sin centrarse en ningún área de aplicación en particular (KNIME, 2019).
- **SPSS (Statistical Package for the Social Sciences)** es un programa informático de estadística creado en 1968 por Norman H. Nie, C. Hadlai (Tex) Hull y Dale H. Bent, con una alta capacidad para trabajar con grandes bases de datos y una sencilla interfaz para un amplio análisis (IBM Corporation, 2020). Es usado en las ciencias sociales y aplicadas,

además de las empresas de investigación de mercado (SPSS *Software* y IBM, 2018).

- Excel (Microsoft, 2020) es una hoja de datos que sirve para manejar datos numéricos y alfanuméricos agrupados de forma vectorial en un diseño gráfico en formato de tabla desarrollado en 1985 por Microsoft para Windows, macOS, Android e iOS, con diferentes herramientas como gráficas, tablas circulares, calculo y un lenguaje de programación llamado Visual Basic para desarrolladores.
- Solver es un programa de complemento de Microsoft Excel que se utiliza para optimizar variables de decisión (hallar su valor máximo o mínimo) dentro de un modelo de programación lineal sujeta a restricciones o limitaciones (Microsoft, 2018).
- KH Coder es un programa de código abierto para análisis de contenido cuantitativo o minería de texto. También se utiliza para lingüística computacional. KH Coder comprende funciones para realizar búsquedas y análisis estadísticos más profundos mediante herramientas de apoyo como el etiquetado gramatical Stanford (POS), herramientas de procesamiento de lenguajes naturales FreeLing, lenguaje de programación Snowball, MySQL y R. En *Machine Learning*, KH Coder se utiliza por sus características de redes de coocurrencia, listas de frecuencia, análisis de agrupamiento jerárquico y escalamiento multidimensional (Higuchi, 2021).
- PHP Script: Según The PHP Group (2021), PHP es un lenguaje de programación del lado del servidor. Se utiliza para desarrollar sitios web estáticos o sitios web dinámicos o aplicaciones web. PHP significa pre-procesador de hipertexto, lo que anteriormente significaba que se podía usar exclusivamente para páginas de inicio personales. Los scripts PHP solo se pueden interpretar en un servidor que tenga PHP instalado.
- CTSim es un simulador de recopilación de datos originalmente creado con fines profesionales en el ámbito de la salud para rayos X tomográficos de objetos fantasmas. Fue escrito por Kevin M. Rosenberg, es de código abierto y está cubierto por la Licencia Pública GNU (GPL). Fue desarrollado en 1983 en una escuela de medicina (Rosenberg, 2014).

- Tanagra Tool Kit es un *software* gratuito de *Data Mining* con fines académicos y de investigación. Propone varios métodos de extracción de datos a partir de análisis de datos exploratorios, aprendizaje estadístico, aprendizaje automático y área de bases de datos. Es un *software* de minería de datos fácil de usar, conforme a las normas actuales del desarrollo de *software* en este dominio (especialmente en cuanto al diseño de su GUI y la forma de uso). Permite analizar datos reales o sintéticos (Rakotomalala, 2004).
- MSQl server es un sistema de gestión de base de datos relacional desarrollado por la empresa Microsoft. El lenguaje de desarrollo utilizado es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL. Este lenguaje se usa para manipular y recuperar datos, crear tablas y definir relaciones entre ellas. Es un sistema de gestión de base de datos relacional desarrollado como un servidor que da servicio a otras aplicaciones de *software* que pueden funcionar ya sea en el mismo ordenador o en otro ordenador a través de una red (incluyendo Internet).
- H2O FLOW es una plataforma de *Machine Learning* open-source desarrollada en Java. Ofrece un gran conjunto de algoritmos ML y un UI de procesamiento (H2o Flow). Permite desarrollar aplicaciones ML en diversos lenguajes: Java, Scala, Python y R, y ofrece interfaces con Spark, HDFS, Amazon S3 y BD NoSQL. Es un notebook web (similar a iPython o Zeppelin) para H2O. Permite combinar ejecución de código, texto, fórmulas, gráficos etc. (Un poco de Java, 2016).

## 13.2. *Software*: Aprendizaje no supervisado

Python es el *software* que más auge ha tenido en los últimos años, tal vez por ser de código abierto. Dentro de la revisión al estado del arte, se encontraron once autores que utilizaron este lenguaje de programación en sus diferentes versiones y derivados, como las versiones 2.7, 3.6, 3.7 y scikit-learn. La versión más reciente de este lenguaje de programación es 3.8.6, publicada el 24 de septiembre del 2020 de forma libre, así como varias implementaciones dentro de las que se destacan CPython, Jython, IronPython, PyPy, etc (tabla 48).

**Tabla 48.** Fuentes bibliográficas que usaron Python

| Software | Versión               | Referencia bibliográfica         |
|----------|-----------------------|----------------------------------|
| Python   | Python                | (Guo <i>et al.</i> , 2016)       |
|          | Python                | (Guleria y Sood, 2018)           |
|          | Python 3.6            | (Shetty <i>et al.</i> , 2019)    |
|          | Python                | (Rana y Garg, 2016)              |
|          | Python                | (Liu y D'Aquin, 2017)            |
|          | Python 2.7            | (Zhang <i>et al.</i> , 2017)     |
|          | Python                | (Balakrishnan, 2013)             |
|          | Python 3.7            | (Iatrellis <i>et al.</i> , 2020) |
|          | Python                | (Rana y Garg, 2017)              |
|          | Python (scikit-learn) | (Drishty <i>et al.</i> , 2019)   |
|          | Python                | (Srinivas y Rajendran, 2019)     |

Fuente: elaboración propia.

En las referencias bibliográficas, se encontraron catorce autores que utilizan el *software* WEKA en diferentes versiones como: 3.8, 3.7.9. Tool, como se muestra a continuación en la tabla 49.

**Tabla 49.** Referencias bibliográficas que utilizaron WEKA

| Software | Versión          | Referencia bibliográfica          |
|----------|------------------|-----------------------------------|
| WEKA     | Weka             | (Trandafili <i>et al.</i> , 2012) |
|          | Weka             | (Alonso y Marcela, 2016)          |
|          | Weka versión 3.8 | (Moubayed <i>et al.</i> , 2020)   |
|          | Weka             | (Joshika, y Rajeshwari, 2019)     |
|          | Weka             | (Nguyen y Vo, 2019)               |
|          | Weka software    | (Li y Shi, 2015)                  |
|          | Weka             | (Chau <i>et al.</i> , 2019)       |
|          | WEKA 3.7.9       | (Durairaj y Vijitha, 2014)        |
|          | WEKA/Java        | (Liu y D'Aquin, 2017)             |
|          | WEKA             | (Bapu <i>et al.</i> , 2015)       |
|          | Weka             | (Shukla y Malviya, 2019)          |
|          | Weka             | (Rana y Garg, 2017)               |



| Software | Versión   | Referencia bibliográfica      |
|----------|-----------|-------------------------------|
| WEKA     | Weka Tool | (Nájera <i>et al.</i> , 2017) |
|          | Weka      | (Ahmad <i>et al.</i> , 2019)  |

**Fuente:** elaboración propia.

En cuanto al *software* R, se encontraron diez investigaciones que realizaron modelos de aprendizaje automático no supervisado para realizar sus modelos, como se muestra en la tabla 50.

**Tabla 50.** Referencias bibliográficas que utilizaron R

| Software      | Versión     | Referencia                       |
|---------------|-------------|----------------------------------|
| R programming | R           | (Karnad <i>et al.</i> , 2017)    |
|               | R           | (De La Hoz <i>et al.</i> , 2019) |
|               | R y RStudio | (Aissaoui <i>et al.</i> , 2019)  |
|               | R           | (Cheng y Shwe, 2019)             |
|               | R y Shiny   | (Najdi y Er-Raha, 2016)          |
|               | R           | (Liu y D'Aquin, 2017)            |
|               | R           | (Hidayah <i>et al.</i> , 2016)   |
|               | R           | (Oyelade <i>et al.</i> , 2010)   |
|               | R           | (Najdi y Brahim, 2016)           |

**Fuente:** elaboración propia.

Como se muestra en la tabla 51, este *software* fue usado pocas veces para programar algoritmos de aprendizaje no supervisado. Se encontraron siete referencias bibliográficas que utilizaron herramientas y escritura base a partir de Java.

**Tabla 51.** Referencias bibliográficas que utilizaron Java

| Software | Versión          | Referencia bibliográfica         |
|----------|------------------|----------------------------------|
| Java     | Java programming | (Hua Leong <i>et al.</i> , 2016) |
|          | Java             | (Nguyen y Vo, 2019)              |
|          | Java Application | (Li y Shi, 2015)                 |
|          | Java             | (Chau <i>et al.</i> , 2019)      |



| Software | Versión                    | Referencia bibliográfica    |
|----------|----------------------------|-----------------------------|
| Java     | Java                       | (Durairaj y Vijitha, 2014)  |
|          | WEKA/Java                  | (Liu y D'Aquin, 2017)       |
|          | Java con frameworks Wicket | (Bapu <i>et al.</i> , 2015) |

**Fuente:** elaboración propia.

En la búsqueda de los artículos académicos de educación, encontramos cinco referencias bibliográficas que utilizan varias versiones de este lenguaje, ya sea para manejar las bases de información o para ajustar los datos (tabla 52).

**Tabla 52.** Referencias bibliográficas que usaron SQL

| Software | Versión                 | Referencia bibliográfica          |
|----------|-------------------------|-----------------------------------|
| SQL      | MySQL                   | (Trandafili <i>et al.</i> , 2012) |
|          | SQL Server              | (Islam y Haque, 2012)             |
|          | MySQL                   | (Balakrishnan, 2013)              |
|          | PostgreSQL              | (Santoso y Yulia, 2019)           |
|          | SQL Query1 y SQL Query2 | (Iam-On y Boongoen, 2017)         |

**Fuente:** elaboración propia.

En la revisión del estado del arte, se encontraron únicamente tres referencias bibliográficas publicadas en los últimos cinco años que usaron Matlab, como se muestra en la tabla 53.

**Tabla 53.** Referencias bibliográficas que usaron MatLab

| Software | Versión | Referencia bibliográfica         |
|----------|---------|----------------------------------|
| Matlab   | Matlab  | (Moubayed <i>et al.</i> , 2020)  |
|          | Matlab  | (Liu y D'Aquin, 2017)            |
|          | Matlab  | (Battaglia <i>et al.</i> , 2016) |

**Fuente:** elaboración propia.

C es un lenguaje de programación de propósito general que actualmente es uno de los lenguajes de programación que dan más soporte en los aplicativos de los diferentes sistemas operativos con que enlaza. Sin embargo, en la revisión bibliográfica se encontraron tres autores que manejaron el código

base para ejecutar librerías que agruparan variables de educación, como se muestra en la tabla 54.

**Tabla 54.** Referencias bibliográficas que usaron C programming

| <i>Software</i> | <i>Versión</i> | <i>Referencia bibliográfica</i>  |
|-----------------|----------------|----------------------------------|
| C               | C programming  | (Kumar <i>et al.</i> , 2020)     |
|                 | C programming  | (Battaglia <i>et al.</i> , 2016) |
|                 | C programming  | (Ramadass y Abhraham, 2010)      |

**Fuente:** elaboración propia.

Existen otros *softwares* que se han usado en diferentes investigaciones analizadas de la investigación referencial, pero que parecen no tener mucha difusión o más bien aplicación dentro del campo de preparación, almacenamiento, análisis y programación de algoritmos de *Machine Learning* en el campo educativo (educación superior). Estos se muestran en la tabla 55.

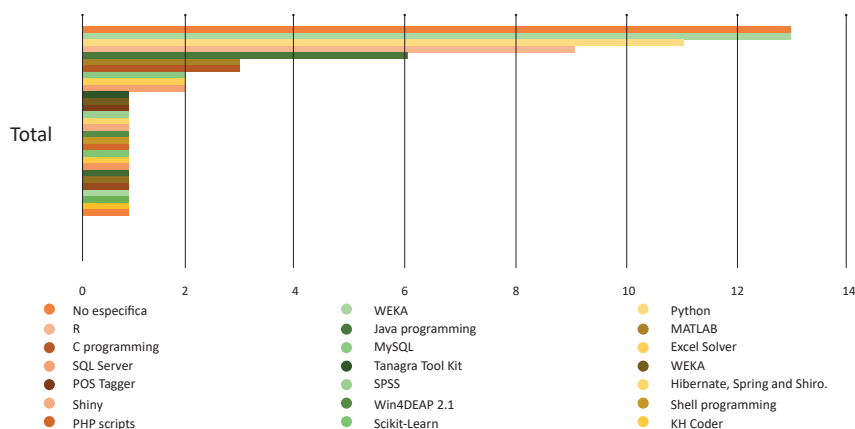
**Tabla 55.** Otras herramientas de análisis de datos

| <i>Clasificación</i>                | <i>Software</i>                      | <i>Referencia bibliográfica</i>           |
|-------------------------------------|--------------------------------------|-------------------------------------------|
| Otros <i>softwares</i> poco comunes | Rapidminer                           | (Martinez-Rodriguez <i>et al.</i> , 2019) |
|                                     | Excel Solver, Win4DEAP 2.1 y MaxDEA, | (Tibay <i>et al.</i> , 2019)              |
|                                     | Shell programming                    | (Bey <i>et al.</i> , 2019)                |
|                                     | KH Coder, POS Tagger                 | (Miller, 2018)                            |
|                                     | Calinski-Harabasz (CH) index         | (Sajjadi <i>et al.</i> , 2018)            |
|                                     | Shiny                                | (Najdi y Er-Raha, 2016)                   |
|                                     | PHP scripts                          | (Cahaya <i>et al.</i> , 2017)             |
|                                     | Hibernate, Spring and Shiro.         | (Bapu <i>et al.</i> , 2015)               |
|                                     | CTSiM                                | (Zhang <i>et al.</i> , 2017)              |
|                                     | Scikit-Learn                         | (Balakrishnan, 2013)                      |
|                                     | Gcc 4.84                             | (Iatrellis <i>et al.</i> , 2020)          |
|                                     | Tanagra Tool Kit                     | (Sabitha y Mehrotra, 2012)                |
|                                     | RapidMiner                           | (Santoso y Yulia, 2019)                   |
|                                     | MS-Excel y SPSS version 22           | (Bhalchandra <i>et al.</i> , 2016)        |

**Fuente:** elaboración propia.

La información acerca del tipo de *software* y el lenguaje que muestran mayor o menor uso dentro del campo de *Machine Learning* enfocado al rendimiento académico a nivel superior puede ser resumida en la figura 84. Se muestra que WEKA, seguido por Python y R, son los más ampliamente usados.

**Figura 84.** Herramientas tecnológicas usadas

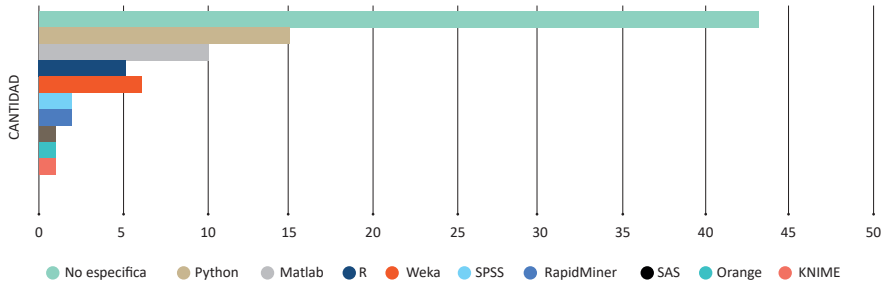


**Fuente:** elaboración propia.

### 13.3. *Software*: Aprendizaje profundo

Luego de analizar los artículos estudiados tanto del tipo teórico, de revisión y aplicación de casos, se realiza la figura 85, que muestra las herramientas implementadas en los trabajos revisados relacionados con algoritmos de aprendizaje profundo, de tal manera que se pueda apreciar cuál o cuáles son las tendencias en cuanto a herramientas que se hayan usado. Lamentablemente en 52,44 % de los trabajos prácticos usados aquí no se especifica ni se menciona el *software* utilizado. Python, más allá del anterior dato, es el *software* más usado, como se pudo mostrar en el ítem anterior de las aplicaciones con aprendizaje no supervisado.

**Figura 85.** Herramientas tecnológicas usadas relacionadas con algoritmos de aprendizaje profundo



Fuente: elaboración propia.

Actualmente existe una amplia variedad de herramientas que sirven para abordar problemas de *Machine Learning* en educación (tabla 56). Estas manejan distintos tipos de lenguajes de programación e interfaces. Estos *frameworks* disponibles ayudan a la creación rápida de modelos de *deep learning* que se pueden ejecutar aprovechando la infraestructura de GPU y de esta forma centrarse en la arquitectura del modelo y no en detalles de bajo nivel (Hernández-Blanco *et al.*, 2019).

**Tabla 56.** Herramientas tecnológicas

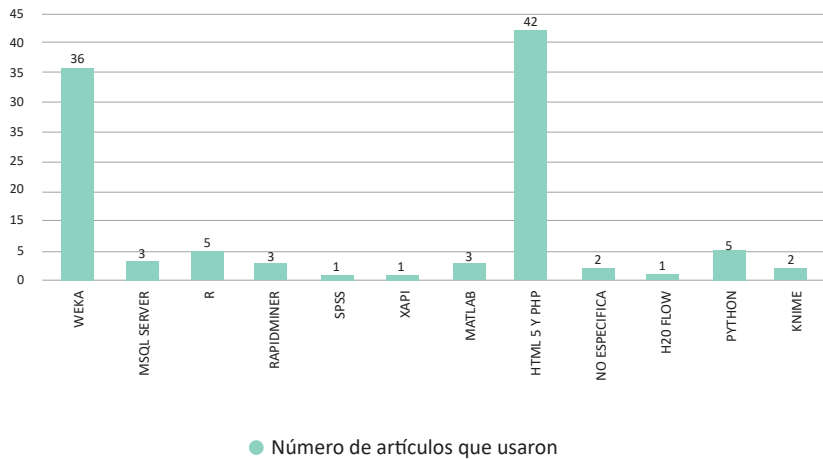
| Framework   | Descripción                                                                                                                                 |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| Keras       | Una interfaz de Python para facilitar la creación de neuronas y de redes profundas; se usa para programar redes tipo CNN y RNN entre otras. |
| Tensor Flow | Está disponible en móvil y escritorio, y se pueden desarrollar modelos de DL en lenguajes de Python, C++ y R. Es apoyado por Google.        |
| Theano      | Es una biblioteca de bajo nivel que admite tanto calculo CPU como GPU.                                                                      |
| Caffe       | Usa lenguaje C++ con interfaz Python, especializada en desarrollo de CNN para procesamiento de imágenes.                                    |
| Otros       | Microsoft Cognitive Toolkit, Deeplearn, Caffe2                                                                                              |

Fuente: basado en Hernández-Blanco *et al.* (2019).

### 13.4. Software: Aprendizaje supervisado

El avance tecnológico ha traído consigo el diseño diversas herramientas para realizar análisis e investigación educativa. Todas estas tienen funcionalidades diferentes y están presentes en los sistemas informáticos, configuradas como *softwares*. Las más conocidas y utilizadas, según la investigación referencial realizada, son Python, R, Matlab, WEKA, entre otros. La figura 86 muestra que las investigaciones consultadas han utilizado algún tipo de *software*. WEKA es el *software* más usado por los investigadores.

**Figura 86.** Frecuencia de utilización de herramientas de análisis



**Fuente:** elaboración propia.



# Referencias

- Un poco de Java (2016). ¿Qué es H2O? <https://unpocodejava.com/2016/03/29/que-es-h2o/#:~:text=H2O%20es%20una%20plataforma%20Machine,Amazon%20S3%20y%20BD%20NoSQL>
- Abu, A. (2016). Educational data mining & students' performance prediction. *International Journal of Advanced Computer Science and Applications*, 7(5). <https://doi.org/10.14569/ijacsa.2016.070531>
- Abubakar, Y. y Nor Bahiah Hj, A. (2017). Prediction of students' performance in e-learning environment using random forest. *International Journal of Innovative Computing*, 7(2), 1-5.
- Acharya, A. y Sinha, D. (2014). Early prediction of students performance using machine learning techniques. *International Journal of Computer Applications*, 107(1), 37-43. <https://doi.org/10.5120/18717-9939>
- Addinsoft. (2021). Agglomerative Hierarchical Clustering (AHC).
- Adekitan, A. I. y Noma-Osaghae, E. (2018). Data mining approach to predicting the performance of first-year student in a university using the admission requirements. *Education and Information Technologies*, 24, 1527-1543. <https://doi.org/10.1007/s10639-018-9839-7>
- Adekitan, A. I. y Salau, O. (2019). The impact of engineering students' performance in the first three years on their graduation result using educational data mining. *Heliyon*, 5(2), e01250. <https://doi.org/10.1016/j.heliyon.2019.e01250>
- Adhatrao, K., Gaykar, A., Dhawan, A., Jha, R. y Honrao, V. (2013). Predicting students' performance using ID3 and C4.5 classification algorithms. *International Journal of Data Mining & Knowledge Management Process*, 3(5), 39-52. <https://doi.org/10.5121/ijdkp.2013.3504>

- Adnan, M., Habib, A., Ashraf, J., Shah, B. y Ali, G. (2020). Improving M-learners' performance through deep learning techniques by leveraging features weights. *IEEE Access*, 8, 131088-131106. <https://doi.org/10.1109/ACCESS.2020.3007727>
- Agrusti, F., Mezzini, M. y Bonavolontà, G. (2020). Deep learning approach for predicting university dropout: a case study at Roma Tre University. *Journal of E-Learning and Knowledge Society*, 16(1), 44-54. <https://doi.org/10.20368/1971-8829/1135192>
- Aher, S. B. y Lobo, L. M. R. J. (2013). Prediction of course selection in e-learning system using combined approach of unsupervised learning algorithm and association rule. *Communications in Computer and Information Science*, 296. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-642-35864-7\\_22](https://doi.org/10.1007/978-3-642-35864-7_22)
- Ahmad A., Nasheen Nur, A. Aileen, B., Erfan Al-Hossami, M., Wenwen Dou, M. y X. N. (2020). Making sense of student success and risk through unsupervised machine learning and interactive storytelling. En I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, E. Millán (Eds), *Artificial Intelligence in Education*. [https://doi.org/10.1007/978-3-030-52237-7\\_1](https://doi.org/10.1007/978-3-030-52237-7_1)
- Ahmad, N. B., Alias, U. Mohamad, N. y Yusof, N. (2019). Principal component analysis and self-organizing map clustering for student browsing behaviour analysis for student browsing behaviour analysis. *Procedia Computer Science*, 163, 550-559. <https://doi.org/10.1016/j.procs.2019.12.137>
- Aissaoui, O., El Madani, Y., Oughdir, L. y Alloui, Y. (2019). Combining supervised and unsupervised machine learning algorithms to predict the learners' learning styles. *Procedia Computer Science*, 148, 87-96. <https://doi.org/10.1016/j.procs.2019.01.012>
- Ajibade, S. S. M., Bahiah Binti Ahmad, N. y Mariyam Shamsuddin, S. (2019). *Educational data mining: enhancement of student performance model using ensemble methods*. IOP Conference Series: Materials Science and Engineering, 551(1). <https://doi.org/10.1088/1757-899X/551/1/012061>
- Al-Barrak, M. A. y Al-Razgan, M. (2016). Predicting Students Final GPA Using Decision Trees: A Case Study. *International Journal of Information and Education Technology*, 6(7), 528-533. <https://doi.org/10.7763/ijiet.2016.v6.745>



- Albán, M. (2019). *Contribuciones a la predicción de la deserción universitaria a través de minería de datos*. [Tesis Doctoral, Universidad Nacional Mayor de San Marcos]. <https://hdl.handle.net/20.500.12672/10776>
- Alloghani, M., Al-Jumeily, D., Hussain, A., Aljaaf, A. J., Mustafina, J. y Petrov, E. (2019). *Application of machine learning on student data for the appraisal of academic performance*. Proceedings - International Conference on Developments in ESystems Engineering, 157-162. <https://doi.org/10.1109/DeSE.2018.00038>
- Almasri, A., Alkhawaldeh, R. S. y Çelebi, E. (2020). Clustering-Based EMT Model for predicting student performance. *Arabian Journal for Science and Engineering*, 45(12), 10067-10078. <https://doi.org/10.1007/s13369-020-04578-4>
- Alonso, A. y Marcela, C. (2016). Unsupervised learning to describe the influence on the academic achievement on students since his learning, *Revista Ingenio*, 10(1), 101-107. <https://orcid.org/0000-0003-2932-3383>
- Alshabandar, R., Hussain, A., Keight, R. y Khan, W. (2020). Students performance prediction in online courses using machine learning algorithms. 2020 International Joint Conference on Neural Networks (IJCNN), 1-7. <https://doi.org/10.1109/IJCNN48605.2020.9207196>
- Altaf, S., Soomro, W. y Rawi, M. I. M. (2019). *Student performance prediction using multi-layers artificial neural networks*. Proceedings of the 2019 3rd International Conference on Information System and Data Mining - ICISDM 2019, 59-64. <https://doi.org/10.1145/3325917.3325919>
- Álvarez Blanco, J., Lau Fernández, R., Pérez Lovelle, S. y Leyva Pérez, E. C. (2016). Predicción de resultados académicos de estudiantes de informática mediante el uso de redes neuronales. *Ingeniare. Revista Chilena de Ingeniería*, 24(4), 715-727. <https://doi.org/10.4067/S0718-33052016000400015>
- Alyahyan, E. y Düşteğör, D. (2020). Predicting Academic Success in Higher Education: Literature Review and Best Practices. *International Journal of Educational Technology in Higher Education*, 17(1), 3, <https://doi.org/10.1186/s41239-020-0177-7>
- Aman, F., Rauf, A., Ali, R., Iqbal, F. y Khattak, A. M. (2019). A Predictive Model for Predicting Students Academic Performance. 10th International Conference on Information, Intelligence, Systems and Applications, IISA 2019, 1-4. <https://doi.org/10.1109/IISA.2019.8900760>

- Amaya, K., Barrientos, E. y Heredia, J. (2014). Modelo predictivo de deserción estudiantil utilizando técnicas de minería de datos. *Mining Techniques*, 13(9), 3127-3134.
- Anderson, H., Afshan, B. y Baker, R. (2019). Predicting Graduation at a Public R1 University Predicting Graduation at a Public R1 University. 2012(February), 1-4. <https://doi.org/10.13140/RG.2.2.26852.83840>
- Arumugam, S., Kovalan, A. y Narayanan, A. E. (2019). *A learning performance assessment model using neural network classification methods of e-learning activity log data*. 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), 441-448. <https://doi.org/10.1109/ICSSIT46314.2019.8987958>
- Asif, R., Merceron, A., Ali, S. A. y Haider, N. G. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers and Education*, 113, 177-194. <https://doi.org/10.1016/j.compedu.2017.05.007>
- Aydoğdu, Ş. (2020). Predicting student final performance using artificial neural networks in online learning environments. *Education and Information Technologies*, 25(3), 1913-1927. <https://doi.org/10.1007/s10639-019-10053-x>
- Badr, G., Algobail, A., Almutairi, H. y Almutery, M. (2016). Predicting students' performance in university courses: a case study and tool in KSU Mathematics Department. *Procedia Computer Science*, 82, 80-89. <https://doi.org/10.1016/j.procs.2016.04.012>
- Bahadir, E. (2016). Using neural network and logistic regression analysis to predict prospective mathematics teachers' academic success upon entering graduate education. *Educational Sciences: Theory & Practice*, 16(3), 943-964. <https://doi.org/10.12738/estp.2016.3.0214>
- Baker, R. S. (2014). Educational data mining: an advance for intelligent systems in education. *IEEE Intelligent Systems*, 29(3), 78-82. <https://doi.org/10.1109/MIS.2014.42>
- Balakrishnan, G. (2013). *Predicting student retention in massive open online courses using Hidden Markov Models* [Reporte técnico]. University of California at Berkeley. <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-109.html>
- Bapu, G., Ashok, M., Shamrao, S. y Tanaji, S. (2015). Clustering Moodle data as a tool for profiling students existing system. *International Journal*

- of *Advanced Research in Computer Engineering & Technology (IJARCET)*, 4, 121-126.
- Barrios, J. (2022). *Fusionando lastecnologías digitalesyla salud*. <https://www.juan-barrios.com/inteligencia-artificial-y-machinE-Learning-para-todos/>
- Based, C. (2009). Modelos G. *Encyclopedia of Biometrics*. [https://doi.org/10.1007/978-0-387-73003-5\\_334](https://doi.org/10.1007/978-0-387-73003-5_334)
- Battaglia, O. Paola, B. y Fazio, C. (2016). A New Approach to Investigate Students' Behavior by Using Cluster Analysis as an Unsupervised Methodology in the Field of Education. *Applied Mathematics*, 7(15), 1649-1673. <https://doi.org/10.4236/am.2016.715142>
- Bey, A., Pérez-Sanagustín, M. y Broisin, J. (2019). Unsupervised Automatic Detection of Learners' Programming Behavior. En M. Scheffel, J. Broisin, V. Pammer-Schindler, A. Ioannou, J. Schneider (Eds), *Transforming Learning with Meaningful Technologies. EC-TEL 2019. Lecture Notes in Computer Science*, 11722. [https://doi.org/10.1007/978-3-030-29736-7\\_6](https://doi.org/10.1007/978-3-030-29736-7_6)
- Bhakti, H. D. (2019). Aplikasi Artificial Neural Network (ANN)sar untuk Memprediksi Masa Studi Mahasiswa Program Studi Teknik Informatika Universitas Muhammadiyah Gresik. *Eksplora Informatika*, 9(1), 88-95. <https://doi.org/10.30864/eksplora.v9i1.234>
- Bhalchandra, P., Muley, A., Joshi, M., Khamitkar, S., Darkunde, N., Lokhande, S. y Wasnik, P. (2016). Prognostication of Student's Performance: An Hierarchical Clustering Strategy for Educational Dataset. *Advances in Intelligent Systems and Computing*, 410, 149-157. [https://doi.org/10.1007/978-81-322-2734-2\\_16](https://doi.org/10.1007/978-81-322-2734-2_16)
- Bhutto, S., Siddiqui, I. F., Arain, Q. A. y Anwar, M. (2020). Predicting Students' Academic Performance Through Supervised Machine Learning. *ICISCT 2020. 2nd International Conference on Information Science and Communication Technology*. <https://doi.org/10.1109/ICISCT49550.2020.9080033>
- Botelho, A. F., Varatharaj, A., Patikorn, T., Doherty, D., Adjei, S. A. y Beck, J. E. (2019). Developing Early Detectors of Student Attrition and Wheel Spinning Using Deep learning. *IEEE Transactions on Learning Technologies*, 12(2), 158-170. <https://doi.org/10.1109/TLT.2019.2912162>
- Burgos, C., Campanario, M. L., Peña, D. de la, Lara, J. A., Lizcano, D. y Martínez, M. A. (2018). Data mining for modeling students'

- performance: A tutoring action plan to prevent academic dropout. *Computers and Electrical Engineering*, 66, 541-556. <https://doi.org/10.1016/j.compeleceng.2017.03.005>
- Burman, I. y Som, S. (2019). *Predicting students academic performance using support vector machine*. Proceedings 2019 Amity International Conference on Artificial Intelligence, AICAI 2019, 756-759. <https://doi.org/10.1109/AICAI.2019.8701260>
- Cahaya, L., Hiryanto, L. y Handhayani, T. (2017). *Student graduation time prediction using intelligent k-medoids algorithm*. Proceeding 3rd International Conference on Science in Information Technology: Theory and Application of IT for Education, Industry and Society in Big Data Era, ICSITech 2017, 263-266. <https://doi.org/10.1109/ICSITech.2017.8257122>
- Castrillón, O., Sarache, W. y Ruiz, S. (2020). Predicción del rendimiento académico por medio de técnicas de inteligencia artificial. *Revista Formación Universitaria*, 13(1), 93-102. <https://doi.org/10.4067/S0718-50062020000100093>
- Castro, J. E. S., Rodríguez, C. C. R., Marín, C. E. M., García, P. A. G. y Castellanos, J. G. (2016). Hacia la construcción de un modelo predictivo de deserción académica basado en técnicas de minería de datos. *Revista Científica*, 3(26), 35-48.
- Chau, L. M., Chau, V. T. N. y Phung, N. H. (2019). *On temporal cluster analysis for early identifying in-trouble students in an academic credit system*. NICS 2018, Proceedings 5th NAFOSTED Conference on Information and Computer Science, 171-176. <https://doi.org/10.1109/NICS.2018.8606827>
- Cheng, W. y Shwe, T. (2019). *Clustering analysis of student learning outcomes based on education data*. Proceedings - Frontiers in Education Conference, FIE, 2019-Octob. <https://doi.org/10.1109/FIE43999.2019.9028400>
- Christian, T. M. y Ayub, M. (2014). *Exploration of classification using NBTree for predicting students' performance*. Proceedings of 2014 International Conference on Data and Software Engineering, ICODSE 2014, 0-5. <https://doi.org/10.1109/ICODSE.2014.7062654>
- Ciubotariu, G. y Crivei, L. M. (2019). Analysing the academic performance of students using unsupervised data mining. *Studia Universitatis Babeş-Bolyai Informatica*, 64(2), 34-48. <https://doi.org/10.24193/subbi.2019.2.03>

- Coelho, O. B. y Silveira, I. (2017). *Deep learning applied to learning analytics and educational data mining: a systematic literature review*. Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)143. <https://doi.org/10.5753/cbie.sbie.2017.143>
- Contreras, L., Fuentes, H. y Rodríguez, J. (2020). Academic interruption model using automatic learning algorithms. *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, 10(3), 16075-16086.
- Corrigan, O. y Smeaton, A. F. (2017). *A course agnostic approach to predicting student success from VLE log data using recurrent neural networks*. European Conference on Technology Enhanced Learning, 545-548. [https://doi.org/10.1007/978-3-319-66610-5\\_59](https://doi.org/10.1007/978-3-319-66610-5_59)
- Costa, E. B., Fonseca, B., Santana, M. A., de Araújo, F. F. y Rego, J. (2017). Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. *Computers in Human Behavior*, 73, 247-256. <https://doi.org/10.1016/j.chb.2017.01.047>
- Crivei, L. M., Czibula, G., Ciubotariu, G. y Dindelegan, M. (2020). *Unsupervised learning based mining of academic data sets for students' performance analysis*. SACI 2020 - IEEE 14th International Symposium on Applied Computational Intelligence and Informatics, Proceedings, 11-16. <https://doi.org/10.1109/SACI49304.2020.9118835>
- Cuadrado Montaña, J. R. (2017). *Modelo para la gestión de indicadores y análisis de permanencia estudiantil de usuarios de Bienestar Universitario de la Universidad Nacional de Colombia*. [Tesis de Maestría, Universidad Nacional de Colombia]. <https://repositorio.unal.edu.co/handle/unal/60137>
- Cuevas-Tello, J. C. (2018). Apuntes de Redes Neuronales Artificiales. *arXiv:1806.05298*. <https://doi.org/10.48550/arXiv.1806.05298>
- Cui, Y., Chen, F. y Shiri, A. (2020). Scale up predictive models for early detection of at-risk students: A feasibility study. *Information and Learning Sciences*, 121(3-4), 97-116. <https://doi.org/10.1108/ILS-05-2019-0041>
- Cuji, B., Gavilanes, W. y Sanchez, R. (2017). Modelo predictivo de deserción estudiantil basado en arboles de decisión Predictive model of student dropout based on decision trees. *Espacios*, 38(55), 17-26. <https://www.revistaespacios.com/a17v38n55/a17v38n55p17.pdf>

- Das, A. K. y Rodriguez-Marek, E. (2019). *A predictive analytics system for forecasting student academic performance: Insights from a pilot project at Eastern Washington University*. 2019 Joint 8th International Conference on Informatics, Electronics and Vision, ICIEV 2019 and 3rd International Conference on Imaging, Vision and Pattern Recognition, IcIVPR 2019 with International Conference on Activity and Behavior Computing, ABC 2019, 255–262. <https://doi.org/10.1109/ICIEV.2019.8858523>
- Daza, A. (2016). Un modelo basado en árboles de decisión para predecir la deserción estudiantil en la educación superior privada. *Ucv-Scientia*, 8(1), 59-73. <https://doi.org/10.18050/revucv-scientia.v8n1a7>
- De La Hoz, E. J. y Fontalvo, T. J. (2019). Methodology of Machine Learning for the Classification and Prediction of Users in Virtual Education Environments. *Información Tecnológica*, 30(1), 247-254. <https://doi.org/10.4067/S0718-07642019000100247>
- Delen, D. (2010). A Comparative Analysis of Machine Learning Techniques for Student Retention Management. *Decision Support Systems*, 49(4), 498-506. <https://doi.org/10.1016/j.dss.2010.06.003>
- Dhanalakshmi, D. y Saravanan, A. (2016). Opinion Mining from Student Feedback Data Using Supervised Learning Algorithms. *Integrated Series in Information Systems*, 36, 183-206. [https://doi.org/10.1007/978-1-4899-7641-3\\_8](https://doi.org/10.1007/978-1-4899-7641-3_8)
- Di Gresia, L., Fazio, M. V., Porto, A., Ripani, L., y Sosa Escudero, W. (2005). Rendimiento y productividad de los estudiantes. El caso de las universidades públicas argentinas. En A. Porto (Ed.), *Economía de la Educación Universitaria: Argentina, Brasil y Perú*. Edulp.
- Doleck, T., Lemay, D. J., Basnet, R. B. y Bazelais, P. (2020). Predictive analytics in education: A Comparison of deep learning frameworks. *Education and Information Technologies*, 25(3), 1951-1963. <https://doi.org/10.1007/s10639-019-10068-4>
- Duan, L. y Xu, L. Da. (2012). Business intelligence for enterprise systems: A survey. *IEEE Transactions on Industrial Informatics*, 8(3), 679-687. <https://doi.org/10.1109/TII.2012.2188804>
- Durairaj, M. y Vijitha, C. (2014). Educational data mining for prediction of student performance using clustering algorithms. *International Journal of Computer Science and Information Technologies*, 5(4), 5987-5991.



- Dursun, M. & Karsak, E. (2010) A fuzzy MCDM approach for personnel selection. *Expert Systems with Applications*, 37(6), 4324-433. <https://doi.org/10.1016/j.eswa.2009.11.067>
- Eckert, K. B. y Suénaga, R. (2015). Análisis de deserción-permanencia de estudiantes universitarios utilizando técnica de clasificación en minería de datos. *Formacion Universitaria*, 8(5), 3-12. <https://doi.org/10.4067/S0718-50062015000500002>
- El Fouki, M., Aknin, N. y El Kadiri, K. E. (2019). Multidimensional approach based on deep learning to improve the prediction performance of DNN models. *International Journal of Emerging Technologies in Learning (IJET)*, 14(02), 30. <https://doi.org/10.3991/ijet.v14i02.8873>
- Escudero, T. (1999). Indicadores del rendimiento académico una experiencia en la Universidad de Zaragoza. In Ministerio de Educación y Cultura. Centro de Publicaciones (Ed.), *Indicadores en la Universidad, información y definiciones* (pp. 251-262). Ministerio de Educación y Cultura, Consejo de Universidades.
- Estrada, J. y Quintero, R. (2015). *Bajo rendimiento académico en la Universidad Distrital Francisco José de Caldas*. Editorial UD.
- Fernandes, E., Holanda, M., Victorino, M., Borges, V., Carvalho, R. y Erven, G. Van. (2019). Educational data mining: predictive analysis of academic performance of public-school students in the capital of Brazil. *Journal of Business Research*, 94, 335-343, <https://doi.org/10.1016/j.jbusres.2018.02.012>
- Ferreira, M. M., Avitabile, C., Álvarez, J. B., Francisco Haimovich Paz y Urzúa, S. (2017). *Momento decisivo: La educación superior en América Latina y el Caribe*. Banco Mundial.
- Forero, L., Reina, Y., Molano, J. (2019). Machine Learning for the Identification of Students at Risk of Academic Desertion. En L. Uden, D. Liberona, G. Sanchez, S. Rodríguez-González (Eds), *Learning Technology for Education Challenges*. LTEC 2019, Communications in Computer and Information Science, 1011. [https://doi.org/10.1007/978-3-030-20798-4\\_40](https://doi.org/10.1007/978-3-030-20798-4_40)
- Formia, S., Lanzarini, L. y Hasperué, W. (2013). Caracterización de la deserción universitaria en la UNRN utilizando minería de datos: Un caso de estudio. *TE & ET: Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, 11, 92-98. <https://doi.org/10.24215/18509959.0.p.%2092-98>

- Forteza, J. (1975). Modelo instrumental de relaciones entre variables motivacionales y rendimiento. *Revista de Psicología General y Aplicada*, 132, 75-91.
- Franzese, M. y Iuliano, A. (2018). Hidden Markov Models. *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, 1, 753-762. <https://doi.org/10.1016/B978-0-12-809633-8.20488-3>
- Gad, S. C. y Rousseaux, C. G. (2002). Use and misuse of statistics in the design and interpretation of studies. *Handbook of Toxicologic Pathology*, 327-418. <https://doi.org/10.1016/b978-012330215-1/50016-8>
- Garbanzo, M. G. (2007). Factores asociados al rendimiento académico en estudiantes universitarios, una reflexión desde la calidad de la educación superior pública. *Revista Educación*, 31(1), 43-63. <http://www.redalyc.org/articulo.oa?id=44031103>
- García, D. (2016). *Construcción de un modelo para determinar el rendimiento académico de los estudiantes basado en learning analytics (análisis del aprendizaje), mediante el uso de técnicas multivariantes*. [Tesis Doctoral, Universidad de Sevilla]. <http://hdl.handle.net/11441/40436>
- García, G. (2014). *Modelo de Machine Learning para la clasificación de pacientes en términos del nivel asistencial requerido en una urgencia pediátrica con área de cuidados mínimos* [Tesis de maestría, Universidad de Sevilla]. <https://hdl.handle.net/20.500.12585/1200>
- García Tinisaray, D. K. (2016). *Construcción de un modelo para determinar el rendimiento académico de los estudiantes basado en learning analytics (análisis del aprendizaje), mediante el uso de técnicas multivariantes*. [Tesis Doctoral, Universidad de Sevilla]. <https://idus.us.es/handle/11441/40436>
- Gkontzis, A. F., Kotsiantis, S., Tsoni, R. y Verykios, V. S. (2018). *An effective LA approach to predict student achievement*. Proceedings of the 22nd Pan-Hellenic Conference on Informatics, 76-81.
- Goethals, B. y Van den Bussche, J. (2002). Relational Association Rules: Getting WARMER. En D. J. Hand, N. M. Adams y R. J. Bolton (Eds.), *Pattern Detection and Discovery. Lecture Notes in Computer Science*. Springer. [https://doi.org/10.1007/3-540-45728-3\\_10](https://doi.org/10.1007/3-540-45728-3_10)
- Gomes, P. C. T. (2019). *Inicio. Machine Learning para Todos*. <https://machine-learningparatodos.com/>
- González, D., Picie, J., González, M., Hernández, C. y Onofre, E. (2019). Nodes: Plataforma para la predicción de deserción escolar



utilizando técnicas de inteligencia artificial. *Journal CIM*, 7(1). <http://doi.org/10.5281/zenodo.4315354>

González, R. (2010). *Python para todos*.

Gu, D., Shen, J. y Wang, X. (2018). *Deep learning for Education*. Digital WPI.

Guleria, P. y Sood, M. (2018). *Predictive data modeling: educational data classification and comparative analysis of classifiers using Python*. PDGC 2018 5th International Conference on Parallel, Distributed and Grid Computing. <https://doi.org/10.1109/PDGC.2018.8745727>

Guo, B. Zhang, R. Xu, G. Shi, C. y Yang, L. (2016). *Predicting students performance in educational data mining*. *Proceedings 2015 International Symposium on Educational Technology, ISET 2015*, 125–128. <https://doi.org/10.1109/ISET.2015.33>

Guruler, H. y Istanbulu, A. (2014). Modeling student performance in higher education using Data Mining. *Studies in Computational Intelligence*, 524, 105-124. [https://doi.org/10.1007/978-3-319-02738-8\\_4](https://doi.org/10.1007/978-3-319-02738-8_4)

Guzmán, C. y Durán, D. (2009). *Deserción estudiantil en la educación superior colombiana*. Ministerio de Educación Nacional, Republica de Colombia. [http://www.mineducacion.gov.co/sistemasdeinformacion/1735/articulos-254702\\_libro\\_desercion.pdf](http://www.mineducacion.gov.co/sistemasdeinformacion/1735/articulos-254702_libro_desercion.pdf)

Hackeling, G. (2014). *Mastering Machine learning with Scikit-Learn: Learn to implement and evaluate machine learning solutions with Scikit-Learn*. Packt Publishing Ltd. <https://www.smallake.kr/wp-content/uploads/2017/03/Mastering-Machine-Learning-with-scikit-learn.pdf>

Hamsa, H., Indiradevi, S. y Kizhakkethottam, J. J. (2016). Student academic performance prediction model using Decision Tree and Fuzzy Genetic Algorithm. *Procedia Technology*, 25, 326–332. <https://doi.org/10.1016/j.protcy.2016.08.114>

Hasbun, T., Araya, A. y Villalon, J. (2016). *Extracurricular activities as dropout prediction factors in higher education using decision trees*. 16th International Conference on Advanced Learning Technologies Extracurricular, 242-244. <https://doi.org/10.1109/ICALT.2016.66>

Hassan, S., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S. y Herrera, F. (2019). Virtual learning environment to predict withdrawal by leveraging deep learning. *International Journal of Intelligent Systems*, 34(8), 1935-1952. <https://doi.org/10.1002/int.22129>

- Heredia, D., Amaya, Y. y Barrientos, E. (2015). Student dropout predictive model using data mining techniques. *IEEE Latin America Transactions*, 13(9), 3127-3134. <https://doi.org/10.1109/TLA.2015.7350068>
- Heredia, D.M., Amaya, Y. y Barrientos, E. U. (2015). Student dropout predictive model using data mining techniques. *IEEE Latin America Transactions*, 13(9), 3127-3134. <https://doi.org/10.1109/TLA.2015.7350068>
- Hernández-Blanco, A., Herrera-Flores, B., Tomás, D. y Navarro-Colorado, B. (2019). A systematic review of deep learning approaches to educational data mining. *Complexity*, 2019, 1-22. <https://doi.org/10.1155/2019/1306039>
- Hernandez Gonzalez, A. G., Melendez Armenta, R. A., Morales Rosales, L. A., Garcia Barrientos, A., Tecpanecatli Xihuitl, J. L. y Algreto, I. (2016). Comparative study of algorithms to predict the desertion in the students at the ITSM-Mexico. *IEEE Latin America Transactions*, 14(11), 4573-4578, <https://doi.org/10.1109/TLA.2016.7795831>
- Hidayah, I., Adji, T. B., Setiawan, N. A. y Maharani, K. (2016). *Work in progress: application of unsupervised learning method toward student's metacognition assessment*. IEEE Global Engineering Education Conference, EDUCON, 1175-1178. <https://doi.org/10.1109/EDUCON.2016.7474704>
- Higuchi, K. (2021). KHCODER. 2021.
- Hu, Y.-H., Lo, C.-L. y Shih, S.-P. (2014). Developing early warning systems to predict students' online learning performance. *Computers in Human Behavior*, 36, 469-478. <https://doi.org/10.1016/j.chb.2014.04.002>
- Hua Leong, F. y Lindsay, M. (2016). Modeling engagement of programming students using unsupervised machine learning technique. *GSTF Journal on Computing (JOC)*, 6(1), 1-6. [https://doi.org/10.5176/2251-3043\\_6.1.114](https://doi.org/10.5176/2251-3043_6.1.114)
- Huang, S. y Fang, N. (2011). *Work in progress: prediction of students' academic performance in an introductory engineering course*. Proceedings - Frontiers in Education Conference, FIE, 11-13. <https://doi.org/10.1109/FIE.2011.6142729>
- Huang, S. y Fang, N. (2013). Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. *Computers & Education*, 61, 133-145. <https://doi.org/10.1016/j.compedu.2012.08.015>

- Huddleston, S. H. y Brown, G. G. (2018). Machine Learning. *Inform's Analytics Body of Knowledge Machine Learning* (pp. 231-274). <https://doi.org/10.1002/9781119505914.ch7>
- Hung, J.-L., Rice, K., Kepka, J. y Yang, J. (2020). Improving predictive power through deep learning analysis of K-12 online student behaviors and discussion board content. *Information Discovery and Delivery*, 48(4), 199-212. <https://doi.org/10.1108/IDD-02-2020-0019>
- Hussain, M., Zhu, W., Zhang, W., Abidi, S. M. R. y Ali, S. (2019). Using Machine Learning to Predict Student Difficulties From Learning Session Data. *Artificial Intelligence Review*, 52(1), 381-407. <https://doi.org/10.1007/s10462-018-9620-8>
- Hussain, S., Gaftandzhieva, S., Maniruzzaman, M., Doneva, R. y Muhsin, Z. F. (2020). Regression Analysis of Student Academic Performance Using Deep learning. *Education and Information Technologies*, 26, 783-798. <https://doi.org/10.1007/s10639-020-10241-0>
- Hussain, S., Muhsion, Z. F., Salal, Y. K., Theodoru, P., Kurtoğlu, F. y Hazarika, G. C. (2019). Prediction Model on Student Performance Based on Internal Assessment using Deep learning. *International Journal of Emerging Technologies in Learning (IJET)*, 14(8), 4-22. <https://doi.org/10.3991/ijet.v14i08.10001>
- Iam-On, N. y Boongoen, T. (2017). Generating descriptive model for student dropout: a review of clustering approach. *Human-Centric Computing and Information Sciences*, 7(1), 1-24. <https://doi.org/10.1186/s13673-016-0083-0>
- IArtificial.Net. (2018). *Las 7 Fases del Proceso de Machine Learning*. <https://www.iartificial.net/tag/proceso-machine-learning/>
- Iatrellis, O. Savvas, I. Fitsilis, P. y Gerogiannis, V. (2020). A two-phase machine learning approach for predicting student outcomes. *Education and Information Technologies*, 26, 69-88. <https://doi.org/10.1007/s10639-020-10260-x>
- IBM. (2020). ¿Qué es Machine Learning? <https://www.ibm.com/mx-es/analytics/machine-learning#:~:text=Machine%20learning%20es%20una%20forma,m%C3%A1s%20precisos%20basados%20en%20datos.>
- IBM (2020). *IBM SPSS Statics 20*. <https://www.ibm.com/support/pages/downloading-ibm-spss-statistics-20>

- Imtiyaz, M. y Singhal, A. (2016). *Evaluating the quality of teaching in higher education institutes using clustering approach and set pair analysis*. Proceedings on 2015 1st International Conference on Next Generation Computing Technologies, NGCT 2015, September, 588-592. <https://doi.org/10.1109/NGCT.2015.7375189>
- InteractiveChaos. (2013). *Reducción de dimensionalidad*. <https://interactivechaos.com/es/manual/tutorial-de-machine-learning/reduccion-de-dimensionalidad>
- Interoperability, W. (2018). *WEKA The workbench for Machine Learning*.
- Islam, H. y Haque, M. (2012). An Approach of Improving Student's Academic Performance by using K-Means Clustering Algorithm and Decision Tree. *International Journal of Advanced Computer Science and Applications*, 3(8), 146-149. <https://doi.org/10.14569/ijacsa.2012.030824>
- Istvan, R. y Lasagna, V. (2019). Sistema informático para la detección temprana de deserción estudiantil universitaria. *Innovación y Desarrollo Tecnológico y Social*, 1(2), 1-15. <https://doi.org/10.24215/26838559e006>
- Jalota, C. y Agrawal, R. (2019). *Analysis of educational data mining using classification*. Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Perspectives and Prospects, COMITCon 2019, 243-247. <https://doi.org/10.1109/COMITCon.2019.8862214>
- Jayne, C., Lanitis, A. y Christodoulou, C. (2011). Neural network methods for one-to-many multi-valued mapping problems. *Neural Computing and Applications*, 20(6), 775-785. <https://doi.org/10.1007/s00521-010-0483-4>
- Jha, N., Ghergulescu, I. y Moldovan, A.-N. (2019). *OULAD MOOC dropout and result prediction using ensemble, deep learning and regression techniques*. Proceedings of the 11th International Conference on Computer Supported Education, 154-164. <https://doi.org/10.5220/0007767901540164>
- Jia, J. y Mareboyana, M. (2013). *Machine learning algorithms and predictive models for undergraduate student retention*. Transactions on Engineering Technologies: Special Issue of the World Congress on Engineering and Computer Science 2013, I, 315-329. [https://doi.org/10.1007/978-94-017-9115-1\\_24](https://doi.org/10.1007/978-94-017-9115-1_24)

- Jin, R. F., Wan, J. X., Gu, S. Y., Sun, P., Zhang, Z. Bin, Jin, X. P. y Xia, Z. L. (2011). Association Analysis: Basic Concepts and Algorithms. *Chinese Journal of Industrial Hygiene and Occupational Diseases*, 29(7), 481-486.
- Jin, X. y Han, J. (2017). K-Medoids Clustering. *Encyclopedia of Machine Learning and Data Mining*. [https://doi.org/10.1007/978-1-4899-7687-1\\_432](https://doi.org/10.1007/978-1-4899-7687-1_432)
- Johnson, F. P. y Robinson, D. M. (2003). Latent Dirichlet Allocation. *The Art Bulletin*, 12(4), 421-422, <https://doi.org/10.2307/3050792>
- Johnson, L., Adams, S., Estrada, V. y Freeman, A. (2015). NMC horizon report: 2015 Higher Education Edition. <http://cdn.nmc.org/media/2015-nmc-horizon-report-HE-ES.pdf>
- Jolliffe, I., Cox, T. F., y Cox, M. A. A. (2010). Multidimensional scaling. *Technometrics*, 38(4), 403-404. <https://doi.org/10.2307/1271315>
- Joshika, P. y Rajeshwari. (2019). Students' performance analysis using simple K-Means. *Journal of the Gujarat Research Society*, 21(14), 990-995. <http://www.gujaratresearchsociety.in/index.php/JGRS/article/view/1915>
- Karnad, A., Yadappanavar, S. y Sunitha Hiremath, P. G. (2017). *Evaluation and validation of problem solving and thinking skills based on student academic performance*. 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, 642-646. <https://doi.org/10.1109/RTEICT.2017.8256676>
- Katare, A. y Dubey, S. (2017). A comparative study of classification algorithms in EDM using 2 Level Classification for predicting student's performance. *International Journal of Computer Applications*, 165(9), 35-40. <https://doi.org/10.5120/ijca2017914023>
- Kaunang, F. J. y Rotikan, R. (2018). *Students' academic performance prediction using data mining*. Proceedings 3rd International Conference on Informatics and Computing, ICIC 2018, 1-5. <https://doi.org/10.1109/IAC.2018.8780547>
- Khalid, M., Pal, N. y Arora, K. (2014). Clustering of image data using K-Means and Fuzzy K-Means. *International Journal of Advanced Computer Science and Applications*, 5(7), 160-163. <https://doi.org/10.14569/ijacsa.2014.050724>
- Khan, A. y Ghosh, S. K. (2020). Student performance analysis and prediction in classroom learning: A review of educational data mining

- studies. *Education and Information Technologies. Education and Information Technologies*, 26, 205-240. <https://doi.org/10.1007/s10639-020-10230-3>
- Khan, I. A., y Choi, J. T. (2014). An application of educational data mining (EDM) technique for scholarship prediction. *International Journal of Software Engineering and Its Applications*, 8(12), 31-42. <https://doi.org/10.14257/ijseia.2014.8.12.03>
- Kim, B.-H., Vizitei, E. y Ganapathi, V. (2018). GritNet: Student performance prediction with deep learning. *arXiv:1804.07405*. <https://doi.org/10.48550/arXiv.1804.07405>
- Kim, B. H., Vizitei, E. y Ganapathi, V. (2018). GritNet 2: Real-Time student performance prediction with domain adaptation. *arXiv:1809.06686*. <https://arxiv.org/pdf/1809.06686.pdf>
- Kitchenham, B. (2004). *Procedures for performing systematic reviews*. Keele University Technical Report.
- Kitchenham, B. A. y Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering version 2.3. [https://www.elsevier.com/\\_\\_data/promis\\_misc/525444systematicreviewsguide.pdf](https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf)
- Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J. y Linkman, S. (2009). Systematic literature reviews in software engineering: A systematic literature review. *Information and Software Technology*, 51(1), 7-15. <https://doi.org/10.1016/j.infsof.2008.09.009>
- Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37, 52-65. <https://doi.org/10.1016/j.neunet.2012.09.018>
- Kulcsar, T., Koncz, P., Balaton, M., Nagy, L. Y Abonyi, J. (2014). Statistical process control based energy monitoring of chemical processes. *Computer Aided Chemical Engineering*, 33(3), 397-402. <https://doi.org/10.1016/B978-0-444-63456-6.50067-3>
- Kumar, V. Krishna, A. Neelakanteswara, P. y Basha, C. (2020). *Advanced prediction of performance of a student in a university using machine learning techniques*. Proceedings of the International Conference on Electronics and Sustainable Communication Systems, ICESC 2020, Icesc, 121-126. <https://doi.org/10.1109/ICESC48915.2020.9155557>
- Lan, Y., Zhou, D., Zhang, H. y Lai, S. (2017). Development of early warning models. En W. Yang (Ed.), *Early Warning for Infectious Disease*



- Outbreak: Theory and Practice* (pp. 35-74). <https://doi.org/10.1016/B978-0-12-812343-0.00003-5>
- Lee, M., Baylor, A. L., Journal, S., Lee, M. y Baylor, A. L. (2006). *Designing metacognitive maps for web-based learning*. *International Forum of Educational Technology & Society*, 9(1), 344-348. <https://www.jstor.org/stable/jeductechsoci.9.1.344>
- Li, M. y Shi, R. (2015). *Mining the relation between dorm arrangement and student performance*. Proceedings IEEE International Conference on Big Data, IEEE Big Data 2015, 2344-2347. <https://doi.org/10.1109/BigData.2015.7364025>
- Liang, J., Yang, J., Wu, Y., Li, C. y Zheng, L. (2016). *Big data application in education: dropout prediction in edx MOOCs*. Proceedings IEEE 2nd International Conference on Multimedia Big Data, BigMM 2016, 440-443. <https://doi.org/10.1109/BigMM.2016.70>
- Liu, S. y D'Aquin, M. (2017). *Unsupervised learning for understanding student achievement in a distance learning setting*. IEEE Global Engineering Education Conference, EDUCON, April, 1373-1377. <https://doi.org/10.1109/EDUCON.2017.7943026>
- Loja, C. (2019). *Aplicación de técnicas de minería de datos en el contexto del rendimiento académico en la Universidad de Cuenca* [Tesis de Pregrado, Universidad de Cuenca]. <http://dspace.ucuenca.edu.ec/handle/123456789/33486>
- Lykourantzou, I., Giannoukos, I., Mpardis, G., Nikolopoulos, V. y Loumos, V. (2009). Early and dynamic student achievement prediction in e-learning courses using neural networks. *Journal of the American Society for Information Science and Technology*, 60(2), 372-380. <https://doi.org/10.1002/asi.20970>
- Ma, X., Yang, Y. y Zhou, Z. (2018). *Using machine learning algorithm to predict student pass rates in online education*. Proceedings of the 3rd International Conference on Multimedia Systems and Signal Processing - ICMSSP '18, 156-161, <https://doi.org/10.1145/3220162.3220188>
- Mahaweerawa, A., Nilnumpetch, C. y Kraipeerapun, P. (2020). Applying stacked generalization with the difference of truth and falsity data to predict student's performance. 2020 5th International Conference on Computer and Communication Systems, ICCCS 2020, 148-152. <https://doi.org/10.1109/ICCCS49078.2020.9118567>

- Márquez, C., Romero, C. y Ventura, S. (2012). Predicción del fracaso escolar mediante técnicas de minería de datos. *IEEE-Rita*, 7(3), 109-117.
- Marrero, H. y Espino, M. (1988). Evaluación comparativa del poder predictor de las aptitudes sobre notas escolares y pruebas objetivas. *Revista de Educación*, 287, 97-112. <https://www.educacionyfp.gob.es/revista-de-educacion/numeros-revista-educacion/numeros-anteriores/1988/re287/re287-05.html>
- Martinez-Rodriguez, R. A., Alvarez-Xochihua, O., Mejia Victoria, O. D., Jordan Aramburo, A. y Gonzalez Fraga, J. A. (2019). Use of machine learning to measure the influence of behavioral and personality factors on academic performance of higher education students. *IEEE Latin America Transactions*, 17(4), 633–641. <https://doi.org/10.1109/TLA.2019.8891928>
- Martins, L. C. B., Carvalho, R. N., Carvalho, R. S., Victorino, M. C. y Holanda, M. (2017). *Early prediction of college attrition using data mining*. Proceedings 16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017, 2017-Decem, 1075-1078. <https://doi.org/10.1109/ICMLA.2017.000-6>
- Masabanda, J. y Zapata, C. (2019). *Modelo basado en minería de datos para determinar factores de deserción estudiantil en la facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi* [Tesis de Pregrado, Universidad Técnica De Cotopaxi]. <http://repositorio.utc.edu.ec/handle/27000/5328>
- MathWorks. (2018). Matlab. 2016.
- Maya, P., Aguilar, J., Zamora, R. y Barron, M. (2018). Predictive model design applying data mining to identify causes of dropout in university students. *Strategy, Technology, & Society*, 7(2), 11-39.
- Menacho Chiok, C. H. (2017). Predicción del rendimiento académico aplicando técnicas de minería de datos. *Anales Científicos*, 78(1), 26-33. <https://doi.org/10.21704/ac.v78i1.811>
- Mengash, H. A. (2020). Using data mining techniques to predict student performance to support decision making in university admission systems. *IEEE Access*, 8, 55462-55470: <https://doi.org/10.1109/ACCESS.2020.2981905>
- Merchán, S. y Duarte, J. (2016). Analysis of Data Mining Techniques for Constructing a Predictive Model for Academic Performance. *IEEE*



*Latin America transactions*, 14(6), 2783-2788. <https://doi.org/10.1109/TLA.2016.7555255>

Microsoft, C. (2018). Definir y resolver un problema con Solver.

Microsoft, C. (2020). Microsoft Excel.

Miguéis, V. L., Freitas, A., Garcia, P. J. V. y Silva, A. (2018). Early segmentation of students according to their academic performance: a predictive modelling approach. *Decision Support Systems*, 115, 36–51. <https://doi.org/10.1016/j.dss.2018.09.001>

Miller, A. H. (2018). Using unsupervised machine learning to model tax practice learning theory. *International Journal of Engineering & Technology*, 7(4), 109. <https://doi.org/10.14419/ijet.v7i2.4.13019>

Ministerio de Educación. (2019a). *Sistema Nacional de información de la educación superior*.

Ministerio de Educación. (2019b). *Sistema para la prevención de la deserción de la educación superior: Estadísticas de Deserción*.

Miranda, M. A. y Guzmán, J. (2017). Análisis de la deserción de estudiantes universitarios usando técnicas de minería de datos. *Formación Universitaria*, 10(3), 61-68. <https://doi.org/10.4067/S0718-50062017000300007>

Mohammadi, M., Dawodi, M., Tomohisa, W. y Ahmadi, N. (2019). *Comparative study of supervised learning algorithms for student performance prediction*. 1st International Conference on Artificial Intelligence in Information and Communication, ICAIIC 2019, 124-127. <https://doi.org/10.1109/ICAIIIC.2019.8669085>

Monroy, F. y Pina, F. H. (2014). Factores que influyen en los enfoques de aprendizaje universitario. Una revisión sistemática. *Educación XXI*, 17(2), 105-124. <https://doi.org/10.5944/educxx1.17.2.11481>

Moss, A., Schluntz, R. y Buhr, P. A. (2018). C: Adding modern programming language features to C. *Software: Practice and Experience*, 48(12), 2111-2146. <https://doi.org/10.1002/spe.2624>

Moubayed, A., Injadat, M., Shami, A. y Lutfiyya, H. (2020). Student engagement level in an e-learning environment: Clustering using K-Means. *American Journal of Distance Education*, 34(2), 137-156. <https://doi.org/10.1080/08923647.2020.1696140>

- Mourdi, Y., Sadgal, M., El Kabtane, H. y Berrada Fathi, W. (2019). A machine learning-based methodology to predict learners' dropout, success or failure in MOOCs. *International Journal of Web Information Systems*, 15(5), 489-509. <https://doi.org/10.1108/IJWIS-11-2018-0080>
- Pavithra, A. y Dhanaraj, S. (2019). Prediction accuracy on academic performance of students using different data mining algorithms with influencing factors. *International Journal of Scientific Research in Computer Science Applications and Management Studies*, 7(5), 1-7.
- Mycosystem información inteligente. (2019). RAPIDMINER. 2019.
- Nagy, M. y Molontay, R. (2018). *Predicting dropout in higher education based on secondary school performance*. 2018 IEEE 22nd International Conference on Intelligent Engineering Systems (INES), 389-394: <https://doi.org/10.1109/INES.2018.8523888>
- Najdi, L. y Brahim, E.-R. (2016). Use of unsupervised clustering to characterize graduate students profiles based on educational outcomes. *International Journal of Computer Techniques*, 3(2), 68-75. <http://www.ijctjournal.org/Volume3/Issue2/IJCT-V3I2P9.pdf>
- Najdi, L. y Er-Raha, B. (2016). *Implementing cluster analysis tool for the identification of students typologies*. Colloquium in Information Science and Technology, CIST, 0, 575-580. <https://doi.org/10.1109/CIST.2016.7804852>
- Nájera, A. B. U., de la Calleja, J. y Medina, M. A. (2017). Associating students and teachers for tutoring in higher education using clustering and data mining. *Computer Applications in Engineering Education*, 25(5), 823-832: <https://doi.org/10.1002/cae.21839>
- Namdeo, J. y Jayakumar, N. (2014). Predicting students performance using data mining technique with rough set theory concepts. *International Journal of Advance Research in Computer Science and Management Studies*, 2(2), 367-373. <https://doi.org/10.1002/cae.21839>
- Naqa, I. El y Murphy, M. J. (2015). What is machine learning? Machine learning in radiation oncology. En I. Naqa, R. Li, Murphy, M. (Eds), *Machine Learning in Radiation Oncology* (pp. 3-11). <https://doi.org/10.1007/978-3-319-18305-3>
- Nguyen, P. y Vo, C. (2019). *Early in-trouble student identification based on temporal educational data clustering*. Proceedings 2019 International Conference on Information Technology, ICIT 2019, 313-318. <https://doi.org/10.1109/ICIT48102.2019.00062>

- Nieto, Y., Gacía, V., Montenegro, C., González, C. y González, R. (2019). Usage of machine learning for strategic decision making at higher educational institutions. *IEEE Access*, 7, 75007-75017. <https://doi.org/10.1109/ACCESS.2019.2919343>
- Noel, M., Ayán, R., Ángel, M. y Díaz, R. (2011). Indicadores de rendimiento de estudiantes universitarios: calificaciones versus créditos acumulados. *Revista de Educación*, 355, 467-492. [http://www.revistaeducacion.educacion.es/re355/re355\\_20.pdf](http://www.revistaeducacion.educacion.es/re355/re355_20.pdf)
- Nuankaew, W. y Thongkam, J. (2020). *Improving student academic performance prediction models using feature selection*. 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 392-395. <https://doi.org/10.1109/ECTI-CON49241.2020.9158286>
- Nurhuda, A., y Rosita, D. (2017). *Prediction student graduation on time using artificial neural network on data mining students STMIK Widya Cipta Dharma Samarinda*. Proceedings 2017 International Conference on E-Commerce, E-Business and E-Government - ICEEG 2017, 86-89. <https://doi.org/10.1145/3108421.3108431>
- Ocando, M. G., Certad, N., Alvarado, S. y Terrones, Á. (2017). *Autonomous 2D SLAM and 3D mapping of an environment using a single 2D LIDAR and ROS*. Proceedings 2017 LARS 14th Latin American Robotics Symposium. <https://doi.org/10.1109/SBR-LARS-R.2017.8215333>
- Okubo, F., Yamashita, T., Shimada, A., y Ogata, H. (2017). *A neural network approach for students' performance prediction*. Proceedings of the Seventh International Learning Analytics & Knowledge Conference, 598-599. <https://doi.org/10.1145/3027385.3029479>
- Oladejo, M. (2010). *A Path-Analytic Study of Socio-Psychological Variables and Academic Performance of Distance Learners in Nigerian Universities* [Tesis Doctoral, University of Lagos]. <https://doi.org/10.13140/RG.2.2.19443.73762>
- Olalekan, A. M., Egwuiche, O. S. y Olatunji, S. O. (2020). *Performance evaluation of machine learning techniques for prediction of graduating students in tertiary institution*. 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), 1-7. <https://doi.org/10.1109/ICMCECS47690.2020.240888>
- Oracle. (2020). *¿Qué es la tecnología Java y para qué la necesito?* [https://www.java.com/es/download/help/whatis\\_java.html](https://www.java.com/es/download/help/whatis_java.html)

- Osmanbegovic, E. y Suljic, M. (2012). Data mining approach for predicting student performance. *Journal of Economics and Business*, 10(1), 3-12. <http://ef.untz.ba/images/Casopis/Paper1Osmanbegovic.pdf>
- Otálora, F. (2019). *Modelo para la identificación de patrones de desempeño académico estudiantil para fortalecer el acompañamiento académico en la Universidad Nacional de Colombia* [Tesis de Maestría, Universidad Nacional de Colombia]. <https://repositorio.unal.edu.co/handle/unal/77758?show=full>
- Oyededeji, A. O., Salami, A. M., Folorunsho, O. y Abolade, O. R. (2020). Analysis and Prediction of Student Academic Performance Using Machine Learning. *JITCE (Journal of Information Technology and Computer Engineering)*, 4(1), 10-15. <https://doi.org/10.25077/jitce.4.01.10-15.2020>
- Oyelade, O. J., Oladipupo, O. O. y Obagbuwa, I. C. (2010). Application of k Means Clustering algorithm for prediction of Students Academic Performance. *Journal of Computer Science*, 7, 292-295. <https://doi.org/10.48550/arXiv.1002.2425>
- Özkan, Y., Koçoğlu, F. Ö. y Erol, Ç. S. (2018). *Prediction of student performance by deep learning algorithm*. 7th International Conference On “Innovations In Learning For The Future”: Digital Tranformation in Education (FL2018). <https://doi.org/136708962>
- Page, M., Gaviria, J.-L. y Gómez Bueno, C. (1990). *Hacia un modelo causal del rendimiento académico*. Centro de Investigación, Documentación y Evaluación (C. I. D. E.). <https://www.researchgate.net/publication/39127951>
- Paliwal, M. y Kumar, U. A. (2009). A study of academic performance of business school graduates using neural network and statistical techniques. *Expert Systems with Applications*, 36(4), 7865-7872. <https://doi.org/10.1016/j.eswa.2008.11.003>
- Pandey, U. (2011). Data Mining : A prediction of performer or underperformer using classification. *Journal of Computer Science and Technology*, 2(2), 686-690. <https://doi.org/10.48550/arXiv.1104.4163>
- Patil, A. P., Ganesan, K. y Kanavalli, A. (2017). *Effective deep learning model to predict student grade point averages*. 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 1–6. <https://doi.org/10.1109/ICCIC.2017.8524317>

- Patil, R., Salunke, S., Kalbhor, M. y Lomte, R. (2018). *Prediction system for student performance using data mining classification*. Proceedings 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018, 2018–2021. <https://doi.org/10.1109/ICCUBEA.2018.8697770>
- Pavithra, A. S. y Dhanaraj, M. S. (2019). Prediction Accuracy on Academic Performance of Students Using Different Data Mining Algorithms with Influencing Factors. *International Journal of Scientific Research & Management Studies*, 7(5), 1-7.
- Poddar, S. y Mukhopadhyay, A. (2020). EXCLUVIS: A MATLAB GUI software for comparative study of clustering and visualization of gene expression data. *arXiv:2008.08193*. <https://doi.org/10.48550/arXiv.2008.08193>
- Polo, D. M., Caballero, L. P. y Gómez, E. M. (2015). Comparación de Redes Neuronales aplicadas a la predicción de Series de Tiempo. *Prospectiva*, 13(2), 88-95. <https://doi.org/10.15665/rp.v13i2.491>
- Ponce, J. C. y Margain, L. (2016). *Aplicación de minería de datos para la identificación de factores de riesgo asociados a la muerte fetal*. VIII Congreso Internaonal en Ciencias Computacionales - CICOMP 2016.
- Porto, A. y Gresia, L. Di. (2005). *Rendimiento de estudiantes universitarios y sus determinantes*. [http://sedici.unlp.edu.ar/bitstream/handle/10915/54674/Documento\\_completo\\_.pdf-PDFA.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/54674/Documento_completo_.pdf-PDFA.pdf?sequence=1)
- Prenkaj, B., Velardi, P., Stilo, G., Distanto, D. y Faralli, S. (2020). A survey of machine learning approaches for student dropout prediction in online courses. *ACM Computing Surveys*, 53(3), 1-34. <https://doi.org/10.1145/3388792>
- Qiu, L., Liu, Y., Hu, Q. y Liu, Y. (2019). Student dropout prediction in massive open online courses by convolutional neural networks. *Soft Computing*, 23(20), 10287-10301. <https://doi.org/10.1007/s00500-018-3581-3>
- Rakotomalala, R. (2004). *Tanagra Tool Kit*.
- Ramadass, S. y Abhraham, A. (2010). Application of Kohonan SOM in prediction. *Communications in Computer and Information Science*, 101, 313-318. [https://doi.org/10.1007/978-3-642-15766-0\\_46](https://doi.org/10.1007/978-3-642-15766-0_46)
- Ramadass, S. y Abraham, A. (2012). Unsupervised control paradigm for performance evaluation. *International Journal of Computer Applications*, 44(20), 27-31. <https://doi.org/10.5120/6380-8850>

- Ramesh, V., Parkavi, P. y Ramar, K. (2013). Predicting Student Performance: A Statistical and Data Mining Approach. *International Journal of Computer Applications*, 63(8), 35-39. <https://research.ijcaonline.org/volume63/number8/pxc3885242.pdf>
- Rana, S. y Garg, R. (2016). *Application of hierarchical clustering algorithm to evaluate students performance of an institute*. Proceedings 2nd International Conference on Computational Intelligence and Communication Technology, CICT 2016, 692-697. <https://doi.org/10.1109/CICT.2016.143>
- Rana, S. y Garg, R. (2017). Prediction of students performance of an institute using ClassificationViaClustering and ClassificationViaRegression. *Advances in Intelligent Systems and Computing*, 508, 333–343. [https://doi.org/10.1007/978-981-10-2750-5\\_35](https://doi.org/10.1007/978-981-10-2750-5_35)
- Rastrollo-Guerrero, J. L., Gómez-Pulido, J. A. y Durán-Domínguez, A. (2020a). Analyzing and Predicting Students' Performance by Means of Machine Learning: A Review. *Applied Sciences*, 10(3), 1042. <https://doi.org/10.3390/app10031042>
- Reyes, N., Morales, J., Moya, J., Teran, C. y Rodriguez, D. N. (2018). Modelo para predecir el rendimiento académico basado en redes neuronales y analítica de aprendizaje. *Revista Ibérica de Sistemas e Tecnologías de Informação*, 17, 258-266. <https://www.proquest.com/scholarly-journals/modelo-para-predecir-el-rendimiento-academico/docview/2195126580/se-2>
- Reyes, P. T. y Villegas, C. M. (2018). An Empirical Comparison of EM and K-Means Algorithms for Binning Metagenomics Datasets. *Revista Chilena de Ingeniería*, 26, 20-27. <https://doi.org/10.4067/S0718-33052018000500020>
- Rico, A., Gaytán, N. y Sánchez, D. (2019). Construcción e implementación de un modelo para predecir el rendimiento académico de estudiantes universitarios mediante el algoritmo Naive Bayes. *Diálogos sobre Educación*, 19, 1-18. <https://doi.org/10.32870/dse.v0i19.509>
- Rico, A. y Sánchez, D. (2018). Diseño de un modelo para automatizar la predicción del rendimiento académico en estudiantes del IPN / Design of a model to automate the prediction of academic performance in students of IPN. *RIDE Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 8(16), 246–266. <https://doi.org/10.23913/ride.v8i16.340>



- Rivas, A., González-Briones, A., Hernández, G., Prieto, J. y Chamoso, P. (2021). Artificial neural network analysis of the academic performance of students in virtual learning environments. *Neurocomputing*, 423, 713-720. <https://doi.org/10.1016/j.neucom.2020.02.125>
- Rodríguez, M. y Ruíz, M. (2011). Indicadores de rendimiento de estudiantes universitarios: calificaciones versus créditos acumulados. *Revista de Educación*, 355, 467-492. <https://sede.educacion.gob.es/publiventa/indicadores-de-rendimiento-de-estudiantes-universitarios-calificaciones-versus-creditos-acumulados/investigacion-educativa/22890>
- Rodriguez, M., Ruiz, M., Ángel, M. y Díaz, R. (2011). Indicadores de rendimiento de estudiantes universitarios: calificaciones versus créditos acumulados. *Revista de Educación* 355, 467-492. [http://www.revistaeducacion.educacion.es/re355/re355\\_20.pdf](http://www.revistaeducacion.educacion.es/re355/re355_20.pdf)
- Rojas, L. (2013). Actualidades investigativas en educación. *Revista Electrónica Actualidades Investigativas en Educación*, 13(1), 17-25. <https://revistas.ucr.ac.cr/index.php/aie/article/view/11707/18183>
- Rosenberg, K. M. (2014). CTsim. 2014.
- Ruby, J., & David, K. (2015). Analysis of Influencing Factors in Predicting Students Performance Using MLP -A Comparative Study. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO Certified Organization)*, 3297(2), 1085-1092. <https://doi.org/10.15680/ijircce.2015.0302070>
- Sabitha, A. S. y Mehrotra, D. (2012). *User centric retrieval of learning objects in LMS*. Proceedings of the 2012 3rd International Conference on Computer and Communication Technology, ICCCT 2012, 14-19. <https://doi.org/10.1109/ICCCT.2012.13>
- Sajjadi, S., Shapiro, B., Mckinlay, C., Sarkisyan, A., Shubin, C. y Osoba, E. (2018). *Finding bottlenecks: Predicting student attrition with unsupervised classifier*. 2017 Intelligent Systems Conference, IntelliSys, 1166-1172. <https://doi.org/10.1109/IntelliSys.2017.8324279>
- Salcedo, A. (2010). Desertion in Colombian universities. *Revista Academia y Virtualidad*, 3(1), 50-60. [http://www.alfaguia.org/alfaguia/files/1319043663\\_03.pdf](http://www.alfaguia.org/alfaguia/files/1319043663_03.pdf)

- Salgado Gomez, D. F. y Velasco Franco, F. E. (2020). *Machine learning como aproximación para establecer el rendimiento académico en estudiantes de educación superior*. Universidad Distrital Francisco José de Caldas.
- Sangam, R. S. y Om, H. (2015). The K-Modes Algorithm with entropy based similarity coefficient. *Procedia Computer Science*, 50, 93-98. <https://doi.org/10.1016/j.procs.2015.04.066>
- Santana, J. S. y Farfán, E. M. (2014). *El Arte de programar en R: Un lenguaje para la estadística*. Instituto Mexicano de Tecnología del Agua. [https://cran.r-project.org/doc/contrib/Santana\\_El\\_arte\\_de\\_programar\\_en\\_R.pdf](https://cran.r-project.org/doc/contrib/Santana_El_arte_de_programar_en_R.pdf)
- Santoso, L. W. y Yulia. (2019). The analysis of student performance using data mining. In S. Bhatia, S. Tiwari, K. Mishra, M. Trivedi (Eds.), *Advances in computer communication and computational sciences. Advances in intelligent systems and computing*. [https://doi.org/10.1007/978-981-13-6861-5\\_48](https://doi.org/10.1007/978-981-13-6861-5_48)
- Schalk, P. D., Wick, D. P., Turner, P. R. y Ramsdell, M. W. (2011). *Predictive assessment of student performance for early strategic guidance*. 2011 Frontiers in Education Conference (FIE). <https://doi.org/10.1109/FIE.2011.6143086>
- Schultz, B. K., Evans, S. W. y Zewelanji, N. S. (2013). Special topic. *School Psychology Review*, 38(1), 221-227. [https://doi.org/10.5005/jp/books/12635\\_13](https://doi.org/10.5005/jp/books/12635_13)
- Sekeroglu, B., Dimililer, K. y Tuncal, K. (2019a). Artificial intelligence in education: application in student performance evaluation. *Revista Dilemas Contemporáneos*, 7(1), 1-21. <https://doi.org/10.46377/dilemas.v28i1.1594>
- Sekeroglu, B., Dimililer, K. y Tuncal, K. (2019b). *Student performance prediction and classification using machine learning algorithms*. Proceedings of the 2019 8th International Conference on Educational and Information Technology, 7–11, <https://doi.org/10.1145/3318396.3318419>
- Shah, M. B., Kaistha, M. y Gupta, Y. (2019). *Student performance assessment and prediction system using machine learning*. 4th International Conference on Information Systems and Computer Networks (ISCON), 386–390. <https://doi.org/10.1109/ISCON47742.2019.9036250>
- Shahiri, A. M., Husain, W. y Rashid, N. A. (2015). A review on predicting student's performance using data mining techniques. *Procedia Computer Science*, 72, 414–422. <https://doi.org/10.1016/J.PROCS.2015.12.157>



- Shetty, A. Pasha, A. Singh, A. Shreyas, N. y Hande, A. (2019). Comparative study of multiple machine learning algorithms for students' performance data for job placement in university. *International Journal of Engineering and Advanced Technology (IJEAT)*, 8(5), 182-187. [https://www.academia.edu/41180094/Comparative\\_Study\\_of\\_Multiple\\_Machine\\_Learning\\_Algorithms\\_for\\_Students\\_Performance\\_Data\\_for\\_Job\\_Placement\\_in\\_University](https://www.academia.edu/41180094/Comparative_Study_of_Multiple_Machine_Learning_Algorithms_for_Students_Performance_Data_for_Job_Placement_in_University)
- Shidhaganti, G. y Patil, A. P. (2019). *Unsupervised techniques to develop a social networking platform in an educational institute*. Proceedings - IEEE 10th International Conference on Technology for Education, 246-247. <https://doi.org/10.1109/T4E.2019.00057>
- Shih, B., Koedinger, K. R. y Scheines, R. (2010). *Unsupervised discovery of student learning tactics. educational data mining*. 3rd International Conference on Educational Data Mining, 201-210. [http://pact.cs.cmu.edu/pubs/Shih,%20Koedinger,%20Scheines-edm2010\\_.pdf](http://pact.cs.cmu.edu/pubs/Shih,%20Koedinger,%20Scheines-edm2010_.pdf)
- Shukla, M. y Malviya, A. K. (2019). *Modified classification and prediction model for improving accuracy of student placement prediction*. Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE) 2019. <https://doi.org/10.2139/ssrn.3351006>
- Singhal, S. y Jena, M. (2013). A study on WEKA tool for data pre-processing, Classification and Clustering. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 2(6), 250-253. <https://www.ijitee.org/portfolio-item/f0843052613/>
- Sisovic, S., Matetic, M. y Bakaric, M. (2015). Mining student data to assess the impact of MOODLE activities and prior knowledge on programming course success. *ACM International Conference Proceeding Series*, 1008, 366-373. <https://doi.org/10.1145/2812428.2812459>
- Sobnath, D., Isiaq, S., Rehman, I. y Nasralla, M. (2020). *Using machine learning advances to unravel patterns in subject areas and performances of university students with special educational needs and disabilities (MALSEND): A conceptual approach*. 4th International Congress on Information and Communication Technology (ICICT 2019) (Vol. 1027, pp. 509-517). (Advances in Intelligent Systems and Computing (AISC)). Springer Singapore. [https://doi.org/10.1007/978-981-32-9343-4\\_41](https://doi.org/10.1007/978-981-32-9343-4_41)
- Sokkhey, P. y Okazaki, T. (2019). *Comparative study of prediction models on high school student performance in mathematics*. 2019 34th International Technical Conference on Circuits/Systems, Computers and

- Communications (ITC-CSCC), 1–4. <https://doi.org/10.1109/ITC-CSCC.2019.8793331>
- Sotomonte, J., Rodríguez, C., Montenegro, C., Gaona, P. y Castellanos, J. (2016). Hacia la construcción de un modelo predictivo de deserción académica basado en técnicas de minería de datos. *Revista Científica*, 26(3), 37-52. <https://doi.org/10.14483/23448350.11089>
- IBM. (2018). SPSS Software <https://www.ibm.com/co-es/products/spss-statistics>
- Srinivas, S. y Rajendran, S. (2019). Topic-based knowledge mining of on-line student reviews for strategic planning in universities. *Computers and Industrial Engineering*, 128, 974-984. <https://doi.org/10.1016/j.cie.2018.06.034>
- Sukhbaatar, O., Usagawa, T. y Choimaa, L. (2019). An artificial neural network based early prediction of failure-prone students in blended learning course. *International Journal of Emerging Technologies in Learning (IJET)*, 14(19), 77. <https://doi.org/10.3991/ijet.v14i19.10366>
- Sun, D., Mao, Y., Du, J., Xu, P., Zheng, Q. y Sun, H. (2019). *Deep learning for dropout prediction in MOOCs*. 8th International Conference on Educational Innovation through Technology (EITT), 87–90. <https://doi.org/10.1109/EITT.2019.00025>
- Syms, C. (2018). Principal components analysis. *Encyclopedia of Ecology*, 566–573. <https://doi.org/10.1016/B978-0-12-409548-9.11152-2>
- Tanuar, E., Heryadi, Y., Lukas, Abbas, B. S. y Gaol, F. L. (2018). *Using machine learning techniques to earlier predict student's performance*. 2018 Indonesian Association for Pattern Recognition International Conference (INAPR), 85-89. <https://doi.org/10.1109/INAPR.2018.8626856>
- The PHP Group. (2021). PHP Script. 2001-2020.
- Tibay, J. P., Ambat, S. C. y Lagman, A. C. (2019). *School-based management performance efficiency modeling and profiling using data envelopment analysis and K-Means clustering algorithm*. 2019 IEEE 4th International Conference on Computer and Communication Systems, ICCCS 2019, 149-153. <https://doi.org/10.1109/CCOMS.2019.8821732>
- Tilson, L. V. Excell, P. S. y Green, R. J. (1988). *A generalisation of the Fuzzy C-Means clustering algorithm*. IGARSS '88 Symposium, Edinburgh, 1783-1784. <https://doi.org/10.1109/igarss.1988.569600>

- Timarán, R., Calderón, A. y Jiménez, J. (2013). Descubrimiento de perfiles de deserción estudiantil con técnicas de minería de datos. *Revista Vínculos*, 10(1), 373-383. <https://doi.org/10.14483/2322939X.4687>
- Timaran, R. y Jiménez, J. (2011). *Detección de patrones de deserción estudiantil en programas de pregrado de instituciones de educación superior con CRISP-DM*. Congreso Iberoamericano de Ciencia, Tecnología, Innovación y Educación, 758. <https://docplayer.es/867817-Deteccion-de-patrones-de-desercion-estudiantil-en-programas-de-pregrado-de-instituciones-de-educacion-superior-con-crisp-dm.html>
- Tomás, J., Expósito, M. y Sempere, S. (2014). Determinantes del rendimiento académico en los estudiantes de grado. Un estudio en administración y dirección de empresas. *Revista de Investigación Educativa*, 32(2), 379-392. <https://doi.org/dx.doi.org/10.6018/rie.32.2.177581>
- Tomasevic, N., Gvozdenovic, N. y Vranes, S. (2020). An overview and comparison of supervised data mining techniques for student exam performance prediction. *Computers & Education*, 143, 103676. <https://doi.org/10.1016/j.compedu.2019.103676>
- Tourón, J. (1984). *Factores del rendimiento académico en la universidad*. Universidad de Navarra.
- Tourón, J. (1985). La predicción del rendimiento académico: Procedimientos y resultados. *Revista Española de Pedagogía*, 1(25), 168–182. <http://dadun.unav.edu/handle/10171/18774>
- Trandafil, E. Allkoçi, A. Kajo, E. y Xhuvani, A. (2012). *Discovery and evaluation of student's profiles with machine learning*. ACM International Conference Proceeding Series, 174–179. <https://doi.org/10.1145/2371316.2371350>
- Tsai, S.-C., Chen, C.-H., Shiao, Y.-T., Ciou, J.-S. y Wu, T.-N. (2020). Precision education with statistical learning and deep learning: a case study in Taiwan. *International Journal of Educational Technology in Higher Education*, 17(1), 12. <https://doi.org/10.1186/s41239-020-00186-2>
- Universidad de Princeton. (2015). *The Fuzzy-K-Means Procedure*. [https://www.cs.princeton.edu/courses/archive/fall08/cos436/Duda/C/fk\\_means.htm](https://www.cs.princeton.edu/courses/archive/fall08/cos436/Duda/C/fk_means.htm)
- Universidad Internacional de Valencia. (2016). *Lenguaje SQL, historia y conceptos básicos*. <https://www.universidadviu.com/co/actualidad/nuestros-expertos/lenguaje-sql-historia-y-conceptos-basicos>

- Universidad Nacional Agraria La Molina. (2017). *Descripción de la metodología de análisis de cluster con algoritmo Fuzzy C-Means*. <https://hdl.handle.net/20.500.12996/3240>
- Universidad Nacional de Colombia, (2019). MySQL. 2019.
- Urteaga, I., Siri, L. y Garófalo, G. (2020). Predicción temprana de deserción mediante aprendizaje automático en cursos profesionales en línea. *RIED. Revista Iberoamericana de Educación a Distancia*, 23(2), 147. <https://doi.org/10.5944/ried.23.2.26356>
- Usman, O. L. y Adenubi, A. O. (2013). Artificial Neural Network (ANN) Model for Predicting Students' Academic Performance. *Journal of Science and Information Technology*, 1(2), 23-37.
- Valero, S., Salvador, A. y García, M. (2010). *Minería de datos: predicción de la deserción escolar mediante el algoritmo de árboles de decisión y el algoritmo de los k vecinos más cercanos*. Recursos Digitales Para La Educación y La Cultura. [http://fcaenlinea.unam.mx/anexos/1566/1566\\_u6\\_act1b.pdf](http://fcaenlinea.unam.mx/anexos/1566/1566_u6_act1b.pdf)
- Veena, N. y Guruprasad, S. (2017). Comparative Analysis of Classification Algorithms for Student Performance. *IJSTE - International Journal of Science Technology & Engineering*, 4(2), 81-85. <http://www.ijste.org/articles/IJSTE4I2035.pdf>
- Vega, J. (2019). *Modelo de pronóstico de rendimiento académico de alumnos en los cursos del programa de estudios básicos de la Universidad Ricardo Palma usando algoritmos de Machine Learning* [Tesis de Maestría, Universidad Ricardo Palma]. <https://hdl.handle.net/20.500.14138/2914>
- Viloria, A., García Guliany, J., Niebles Núñez, W., Palma, H. H. y Niebles Núñez, L. (2020). Data Mining Applied in School Dropout Prediction. *Journal of Physics*, 1432, 12092. <https://doi.org/10.1088/1742-6596/1432/1/012092>
- Virtual, C. C. (2021). Factores de aprendizaje.
- Vital, T. P., Lakshmi, B. G., Rekha, H. S. y DhanaLakshmi, M. (2019). Student Performance Analysis with Using Statistical and Cluster Studies. *Soft Computing in Data Analytics*, 758, 743-757. [https://doi.org/10.1007/978-981-13-0514-6\\_71](https://doi.org/10.1007/978-981-13-0514-6_71)
- Waheed, H., Hassan, S.-U., Aljohani, N. R., Hardman, J., Alelyani, S. y Nawaz, R. (2020). Predicting academic performance of students

- from VLE Big Data using deep learning models. *Computers in Human Behavior*, 104, 106189. <https://doi.org/10.1016/j.chb.2019.106189>
- Wei, W., Han, J., Kong, J. y Xia, H. (2016). *Prediction of the scholarship using comprehensive development*. Proceedings - 4th International Conference on Enterprise Systems: Advances in Enterprise Systems, ES 2016, 183-188. <https://doi.org/10.1109/ES.2016.30>
- Weka 3 (2018). *Data Mining with Open Source Machine Learning Software in Java*. [https://www.cs.waikato.ac.nz/~ml/weka/tips\\_and\\_tricks\\_old.html](https://www.cs.waikato.ac.nz/~ml/weka/tips_and_tricks_old.html)
- KNIME. (2019). *What's New in KNIME Analytics Platform 4.0 and KNIME Server 4.9*. <https://www.knime.com/whats-new-in-knime-40>
- Widyaningsih, Y., Fitriani, N. y Sarwinda, D. (2019). *A semi-supervised learning approach for predicting student's performance: first-year students case study*. 2019 12th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, 291-295. <https://doi.org/10.1109/ICTS.2019.8850950>
- Wikipedia. (2019). Perceptrón. <https://es.wikipedia.org/wiki/Perceptrón>
- Will, N. N., Kemczinski, A. y Parpinelli, R. (2019). *Deep learning para Previsão do Desempenho do Estudante: Um Mapeamento Sistemático da Literatura*. Anais Do XXX Simpósio Brasileiro de Informática Na Educação (SBIE 2019), 1798. <https://doi.org/10.5753/cbie.sbie.2019.1798>
- Wu, N., Zhang, L., Gao, Y., Zhang, M., Sun, X. y Feng, J. (2019). *CLMS-Net: dropout prediction in MOOCs with deep learning*. ACM TURC '19: Proceedings of the ACM Turing Celebration Conference – China, 1–6. <https://doi.org/10.1145/3321408.3322848>
- Xiangyu Song, Li, J., Sun, S., Yin, H., Dawson, P. y Doss, R. (2020). SEPN: A Sequential Engagement Based Academic Performance Prediction Model. *IEEE Intelligent Systems*, 36(1), 46-53. <https://doi.org/10.1109/MIS.2020.3006961>
- Xing, W. y Du, D. (2019). Dropout Prediction in MOOCs: Using Deep learning for Personalized Intervention. *Journal of Educational Computing Research*, 57(3), 547-570. <https://doi.org/10.1177/0735633118757015>
- Xu, S. (2010). The Expectation Maximization Algorithm. *Heredity*, 105(5), 483–494. <https://doi.org/10.1038/hdy.2009.180>
- Xu, X., Wang, J., Peng, H. y Wu, R. (2019). Prediction of academic performance associated with internet usage behaviors using Machine

- Learning algorithms. *Computers in Human Behavior*, 98, 166-173. <https://doi.org/10.1016/j.chb.2019.04.015>
- Yamasari, Yuni. Nugroho, Supeno M.S. Sukajaya, I. N. y Purnomo, M. H. (2017). *Features extraction to improve performance of clustering process on student achievement*. 20th International Computer Science and Engineering Conference: Smart Ubiquitous Computing and Knowledge, ICSEC 2016, 2–6. <https://doi.org/10.1109/ICSEC.2016.7859946>
- Yang, Y. y Chen, K. (2011). Temporal data clustering via weighted clustering ensemble with different representations. *IEEE Transactions on Knowledge and Data Engineering*, 23(2), 307-320. <https://doi.org/10.1109/TKDE.2010.112>
- Yasir, M., Erna, A., Tungadi, E. y Syamsuddin, I. (2017). *Determining single tuition fee of higher education in indonesia: a comparative analysis of data mining classification algorithms*. Proceedings of 2017 4th International Conference on New Media Studies, 113-117. <https://doi.org/10.1109/CONMEDIA.2017.8266041>
- Yunita, A., Santoso, H. B. y Hasibuan, Z. A. (2019). *Deep learning for predicting students' academic performance*. 2019 4th International Conference on Informatics and Computing (ICIC), 1-6. <https://doi.org/10.1109/ICIC47613.2019.8985721>
- Zacharis, Z. N. (2016). Predicting student academic performance in blended learning using artificial neural networks. *International Journal of Artificial Intelligence & Applications*, 7(5), 17-29. <https://doi.org/10.5121/ijaia.2016.7502>
- Zambrano Matamala, C., Rojas Diaz, D., Carvajal Cuello, M., & Acuña Leiva, G. (2011). Análisis de rendimiento académico estudiantil usando data warehouse y redes neuronales. *Ingeniare. Revista chilena de ingeniería*, 19(3), 369-381. <https://www.redalyc.org/pdf/772/77221486007.pdf>
- Zambrano, P. (2019). *Análisis de retención de estudiantes en la Carrera de Negocios Internacionales de la Facultad de Ciencias Económicas y Administrativas de la UCSG, durante los dos primeros años de la carrera, mediante técnica de Minería de Datos* [Tesis de Pregrado, Universidad Católica de Santiago de Guayaquil]. <http://repositorio.ucsg.edu.ec/handle/3317/12618>
- Zarate, A. (2019). *Reconocimiento de patrones de deserción de estudiantes universitarios basados en modelos de clasificación*. [Tesis de Pregrado, Universidad



Nacional de San Agustín de Arequipa]. <http://repositorio.unsa.edu.pe/handle/UNSA/9419>

- Zhang, J., Mo, Y., Chen, C. y He, X. (2020). *Neural Attentive knowledge tracing model for student performance prediction*. 2020 IEEE International Conference on Knowledge Graph (ICKG), 641-648. <https://doi.org/10.1109/ICKG50248.2020.00096>
- Zhang, N., Biswas, G. y Dong, Y. (2017). *Characterizing students' learning behaviors using unsupervised learning methods*. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10331 LNAI, 430-441. [https://doi.org/10.1007/978-3-319-61425-0\\_36](https://doi.org/10.1007/978-3-319-61425-0_36)
- Zhang, T., Yin, C. y Pan, L. (2017). *Improved clustering and association rules mining for university student course scores*. Proceedings of the 2017 12th International Conference on Intelligent Systems and Knowledge Engineering, 1-6. <https://doi.org/10.1109/ISKE.2017.8258808>
- Zhang, X.-D. (2017). *A matrix algebra approach to artificial intelligence*. Springer Nature Singapore.
- Zhang, Y., Oussena, S., Clark, T. y Kim, H. (2010). *Use data mining to improve student retention in higher education. A case study*. In ICIES 2010 (Ed.), 12th International Conference on Enterprise Information Systems, 190-197. <https://pdfs.semanticscholar.org/abf6/57446266610f579cf-761d447929d4c7f4f38.pdf>
- Zulfiker, M. S., Kabir, N., Amin, A., Chakraborty, P. y Mahfujur, M. (2020). Predicting students' performance of the private universities of Bangladesh using machine learning approaches. *International Journal of Advanced Computer Science and Applications*, 11(3). <https://doi.org/10.14569/IJACSA.2020.0110383>





# **Autores**

## **Leonardo Emiro Contreras Bravo**

Ingeniero mecánico, máster en Ingeniería (materiales y procesos de manufactura), PhD(c) en Ingeniería. Docente de planta, Universidad Distrital Francisco José de Caldas. Director, grupo de investigación en Diseño, Modelamiento y Simulación, DIMSI. Áreas de conocimiento: diseño, manufactura, análisis de datos, aprendizaje automático.

## **Giovanny Mauricio Tarazona Bermúdez**

Ingeniero industrial, máster en Diseño y Gestión de proyectos tecnológicos. Doctor en Sistemas y servicios informáticos para internet. Docente de planta, Universidad Distrital Francisco José de Caldas. Director, grupo de investigación en Comercio Electrónico Colombiano, GICOECOL. Áreas de conocimiento: innovación estratégica, comercio electrónico.

## **Angie Paola Alemán Cardona**

Ingeniera industrial, de la Universidad Distrital Francisco José de Caldas, con más de 3 años de experiencia en el área de Tecnología. Actualmente, lidera el soporte de Generación de demanda para las unidades de Automatización y Data & AI de IBM en Latinoamérica. Ha alcanzado certificaciones en metodología Agile, Design Thinking, Transformación Digital y Seguridad Informática.

## **Jhonatan Estiven Ruiz Vanegas**

Ingeniero Industrial, Universidad Distrital Francisco José de Caldas con énfasis en *Data analysis*, productividad y eficiencia de procesos logísticos. Actualmente, miembro del Ministerio de Ciencias, Tecnología e Innovación de Colombia, dentro del equipo de Logística Inversa de los centros de distribución nacional de la compañía SODIMAC Colombia S. A.

## **Brandon Herney Hernández Arcila**

Técnico en sistemas, ingeniero industrial, Universidad Distrital Francisco José de Caldas, con múltiples especialidades en el sector automotor. Actualmente, gerencia un taller automotor, con *hobby* en diseño para impresión 3D.

## **Daniela Garzón Trujillo**

Ingeniera industrial, Universidad Distrital Francisco José de Caldas con profundización en metodologías Lean y planeación de la demanda en procesos productivos de consumo masivo y retail. Actualmente, se desempeña como *supply and demand planner senior*, responsable del abastecimiento de todas las tiendas Ópticas Lafam en el país.

## **Óscar Eduardo Sánchez Gómez**

Ingeniero industrial, Universidad Distrital Francisco José de Caldas. Analista comercial del grupo Coquecol, con experiencia en procesos como *sales and operation planning*. Actualmente, hace parte del proceso comercial de la organización para implementar las herramientas tecnológicas y mejorar la experiencia del cliente e inteligencia de mercados, mediante proyectos tecnológicos.

## **Marisol Ángel Cifuentes**

Tecnóloga industrial e ingeniera de Producción, Universidad Distrital Francisco José de Caldas. Técnica en programación de software (SENA). Ha trabajado en el área de mejoramiento continuo y calidad de diferentes empresas, ayudando a mantener sus sistemas de gestión. También ha trabajado como gestora de procesos y ha liderado el proceso de calidad y *testing* para una *start app*.

Este libro se terminó  
de imprimir en abril de  
2023 en la Editorial UD  
Bogotá, Colombia