Facultad de Ingeniería
Universidad de la República

Estudio de la viabilidad de la aplicación de técnicas de aprendizaje automático a la educación

Informe de Proyecto de Grado presentado al Tribunal Evaluador como requisito de graduación de la carrera Ingeniería en Computación

Documentación del trabajo realizada por:

Alejandro Añón
Emiliano Torrano

Tutor: Gustavo Guimerans
Tutora: Mónica Wodzislawski

Montevideo, Uruguay

Resumen

En la actualidad existe un crecimiento considerable en el uso de sistemas de aprendizaje a distancia, en donde se recolecta una gran cantidad de información. Además, el incremento en el poder de cálculo de las computadoras generó un contexto muy propicio para la aplicación de técnicas de aprendizaje automático. En este contexto, surge el interés de investigar si es viable la aplicación de estas técnicas en el ámbito educativo, en particular con los datos de un sistema de aprendizaje a distancia.

En primera instancia se estudia el estado del arte de la aplicación de técnicas de aprendizaje automático a la educación. Allí se concluye que existen varias aplicaciones posibles —entre las que se destacan la predicción del rendimiento académico de los estudiantes y la sugerencia de contenido o ejercicios que maximicen el aprendizaje— que pueden aportar valor a los distintos actores de la educación.

Se busca estudiar en mayor profundidad la aplicación de técnicas de aprendizaje automático para predecir el rendimiento académico de los estudiantes con el fin de generar alertas de forma temprana a los profesores para que puedan tomar acciones a tiempo. Las acciones que se puedan tomar ante una alerta de pérdida de un curso dependen fuertemente del contexto, como la asignatura y el estudiante, por lo que queda por fuera de este proyecto.

Luego de definido el foco del proyecto, se toma como caso de estudio el Centro de Ensayos de Software (CES). En este centro se dictan carreras con la particularidad de ser completamente en línea, sobre la plataforma Moodle, lo cual permite que se dispongan de datos de todas las actividades que realizan los estudiantes en el sistema informático. Sobre estos datos se aplican distintas técnicas de aprendizaje automático para predecir si un estudiante aprobará o no un curso y también para predecir qué nota obtendrá al finalizar el curso.

Se detallan los pasos seguidos para obtener las predicciones con el objetivo de que sean de utilidad para el CES así como también que sea posible generalizarlo a otros centros educativos. Se tiene como objetivo que la solución sea lo más genérica posible y agnóstica al centro educativo que se utiliza para el análisis, aunque esto no siempre es posible. Se describen los pasos que solo aplican al CES para que puedan ser adaptados a otra realidad en caso de ser de interés.

Por último, se desarrolla una plataforma web para facilitar el proceso realizado y sea utilizado por el CES. La misma realiza todos los pasos detallados previamente y ofrece los resultados en una interfaz gráfica para que se pueda tomar acción sobre las alertas allí generadas. Las transformaciones realizadas en los datos y los parámetros utilizados en los modelos predictivos son configurables para poder ser aplicados a otros cursos o carreras.

Índice general

Índice de cuadros

Índice de figuras

Glosario

Análisis de Discriminante Lineal Es una generalización del discriminante lineal de Fisher, un método utilizado en estadística, reconocimiento de patrones y aprendizaje de máquinas para encontrar una combinación lineal de rasgos que caracterizan o separan dos o más clases de objetos o eventos.

Aprendizaje simbólico Es la capacidad humana de ir más allá de lo literal y representar el significado de determinadas acciones, emociones y/o palabras. Se trata de realizar interpretaciones y conductas dotadas de significado, rompiendo con los códigos básicos establecidos.

Árboles de decisión Aprendizaje basado en árboles de decisión utiliza un árbol de decisión como un modelo predictivo que mapea observaciones sobre un artículo a conclusiones sobre el valor objetivo del artículo. Es uno de los enfoques de modelado predictivo utilizadas en estadísticas, minería de datos y aprendizaje automático. Los modelos de árbol, donde la variable de destino puede tomar un conjunto finito de valores se denominan árboles de clasificación. En estas estructuras de árbol, las hojas representan etiquetas de clase y las ramas representan las conjunciones de características que conducen a esas etiquetas de clase. Los árboles de decisión, donde la variable de destino puede tomar valores continuos (por lo general números reales) se llaman árboles de regresión.

Artificial Neural Network (ANN) Modelo inspirado en el cerebro compuesto de capas (al menos una de las cuales está oculta) que consisten en unidades conectadas simples o neuronas, llamadas neuronas artificiales, conectadas entre sí para transmitirse señales. La información de entrada atraviesa la red neuronal (donde se somete a diversas operaciones) produciendo unos valores de salida.

Atributo (feature) Variable de entrada que se usa para realizar predicciones. Los Atributos son las propiedades individuales que se pueden medir de un fenómeno que se observa. La elección de atributos informativos, discriminatorios e independientes es un paso crucial para la eficacia de los algoritmos de Machine Learning.

Backpropagation (propagación del error hacia atrás) Algoritmo principal para realizar descenso de gradientes en redes neuronales. Primero, los valores de resultado de cada nodo se calculan (se almacenan en caché) y se propagan hacia adelante. Después, el derivado

parcial del error con respecto a cada parámetro se calcula y se propaga hacia atrás a través del gráfico.

Baseline Modelo simple o heurístico que se usa como punto de partida para comparar la eficacia del desempeño de un modelo. Un modelo de referencia ayuda a los programadores de modelos a cuantificar el rendimiento mínimo esperado en un problema en particular.

Business Intelligence Se denomina inteligencia empresarial, inteligencia de negocios o BI (del inglés business intelligence), al conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitectura técnicas, los cuales están enfocados a la administración y creación de conocimiento sobre el medio, a través del análisis de los datos existentes en una organización o empresa. El término inteligencia empresarial se refiere al uso de datos en una empresa para facilitar la toma de decisiones. Abarca la comprensión del funcionamiento actual de la empresa, bien como la anticipación de acontecimientos futuros, con el objetivo de ofrecer conocimientos para respaldar las decisiones empresariales.

Algoritmos de clasificación Se usan cuando el resultado deseado es una etiqueta discreta. En otras palabras, son útiles cuando la respuesta a su pregunta sobre su empresa cae dentro de un conjunto finito de resultados posibles. Muchos casos de uso, como determinar si un correo electrónico es correo no deseado o no, solo tienen dos resultados posibles. Esto se llama clasificación binaria. La clasificación multicategoría captura todo lo demás, y es útil para la segmentación del cliente, la categorización de imágenes y audio y el análisis de texto para optimizar el sentimiento del cliente.

Clasificador Lineal En el campo del aprendizaje automático, el objetivo del aprendizaje supervisado es usar las características de un objeto para identificar a qué clase (o grupo) pertenece. Un clasificador lineal logra esto tomando una decisión de clasificación basada en el valor de una combinación lineal de sus características. Las características de un objeto son típicamente presentadas en un vector llamado vector de características.

Clasificador No Lineal Este tipo de clasificadores utilizan funciones no lineales para resolver problemas que los clasificadores lineales no son capaces, debido a que los datos no son linealmente separables.

Decision Tree (DT) Es un modelo de predicción utilizado en diversos ámbitos que van desde la inteligencia artificial hasta la Economía. Dado un conjunto de datos se fabrican diagramas de construcciones lógicas, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren de forma sucesiva, para la resolución de un problema.

Educational Data Mining Es la utilización de la minería de datos en el ámbito de la enseñanaza. También conocida como Descubrimiento de Conocimiento en Bases de datos (sus siglas en inglés son KDD, por Knowledge Discovery in Databases), es el campo que

nos permite descubrir información nueva y potencialmente útil de big data o grandes cantidades de datos.

Impedance Mismach Conjunto de dificultades conceptuales y técnicas que se presentan generalmente cuando una base de datos relacional es servida por una o más aplicaciones escritas en un lenguaje de programación orientado a objetos dado que los objetos o clases deben ser mapeados a tablas de la base de datos.

Internationalized Resource Identifier (IRI) Estándar de protocolo de internet que extiende el conjunto de caracteres ASCII del protocolo Uniform Resource Identifier (URI) con el fin de identificar recursos.

K-Nearest Neighbors Es un algoritmo de Machine Learning que consiste en realizar predicciones sobre una clase en base a la clase a la que pertenecen los puntos vecinos más cercanos al que intentamos predecir.

Knowledge Discovery in Databases (KDD) Es el proceso de identificar patrones válidos, novedosos, potencialmente útiles y comprensibles a partir de los datos.

Learning Analytics Es la medición, recopilación, análisis e informe de datos sobre los alumnos y sus contextos, con el fin de comprender y optimizar el aprendizaje y los entornos en los que se produce. Un campo estrechamente relacionado es la Minería de datos educativa.

Learning Management System (LMS) Es un software instalado en un servidor web que se emplea para administrar, distribuir y controlar las actividades de formación no presencial (o aprendizaje electrónico) de una institución u organización. Permitiendo un trabajo de forma asíncrona entre los participantes.

Learning Record Store (LRS) Es un contenedor de datos que sirve como repositorio para registros generados en actividades de aprendizaje creadas con el estándar Tin Can API.

Machine Learning Es el campo de las ciencias de la computación y una rama de la inteligencia artificial, cuyo objetivo es desarrollar técnicas que permitan que las computadoras aprendan. De forma más concreta, se trata de crear programas capaces de generalizar comportamientos a partir de una información suministrada en forma de ejemplos.

Mapas auto-organizados de Kohonen Pertenece a la categoría de redes no supervisadas, la diferencia con otras redes, es que las neuronas que representan patrones parecidos aparecen juntas en el espacio salida, este espacio puede ser unidimensional, una línea, bidimensional, un plano o N-dimensional. Es el propio diseñador de la red el que establece el espacio de salida que tendrá la red.

Massive Online Open Courses (MOOC) Son cursos online masivos y abiertos, ofrecen formación en línea. Están diseñados para ser impartidos a un gran número de alumnos a la vez.

Moodle Es una herramienta de gestión de aprendizaje (LMS), o más concretamente de Learning Content Management System (LCMS), de distribución libre, escrita en PHP. Está concebida para ayudar a los educadores a crear comunidades de aprendizaje en línea, Moodle es usada en blended learning, educación a distancia, clase invertida y diversos proyectos de e-learning en escuelas, universidades, oficinas y otros sectores.

Naïve Bayes Es una técnica de clasificación y predicción que construye modelos que predicen la probabilidad de posibles resultados. Utiliza datos históricos para encontrar asociaciones y relaciones y hacer predicciones.

Red de Hopfield Es una forma de red neuronal artificial recurrente inventada por John Hopfield. Las redes de Hopfield se usan como sistemas de Memoria asociativa con unidades binarias. Están diseñadas para converger a un mínimo local, pero la convergencia a uno de los patrones almacenados no está garantizada.

Red neuronal multicapa con alimentación hacia adelante Es una red neuronal artificial donde las conexiones entre las unidades no forman un ciclo. En esta red, la información se mueve en una única dirección: adelante. De los nodos de entrada, a través de los nodos escondidos (si los hay) hacia los nodos de salida.

Regresión Son útiles para predecir productos que son continuos y por lo tanto el resultado se representa mediante una cantidad que puede determinarse de manera flexible en función de las entradas del modelo en lugar de limitarse a un conjunto de posibles etiquetas. Los problemas de regresión con entradas ordenadas por tiempo se denominan problemas de pronóstico de series temporales, como el pronóstico ARIMA, que permite a los científicos de datos explicar los patrones estacionales en las ventas, evaluar el impacto de las nuevas campañas de marketing, etc.

Selección de atributos (Feature selection) Es el proceso de seleccionar un conjunto de feature (variables) relevantes para la construcción de un modelo predictivo.

Society for Learning Analytics Research (SoLAR) Es una red interdisciplinaria de liderazgo de investigadores internacionales que están explorando el rol e impacto de las técnicas de análisis en la enseñanza. Ha estado activa en la organización de la conferencia internacional Learning Analytics & Knowledge (LAK).

Support Vector Machines (SVM) Es un algoritmo de aprendizaje supervisado que se puede emplear para clasificación binaria o regresión. Las máquinas de vectores de soporte son muy populares en aplicaciones como el procesamiento del lenguaje natural, el habla, el reconocimiento de imágenes y la visión artificial. Una máquina de vectores de soporte construye un hiperplano óptimo en forma de superficie de decisión, de modo que el margen de separación entre las dos clases en los datos se amplía al máximo. Los vectores

de soporte hacen referencia a un pequeño subconjunto de las observaciones de entrenamiento que se utilizan como soporte para la ubicación óptima de la superficie de decisión. Las máquinas de vectores de soporte pertenecen a una clase de algoritmos de Machine Learning denominados métodos kernel y también se conocen como máquinas kernel.

Web Analytics Es un conjunto de técnicas relacionadas con el análisis de datos relativos al tráfico en un sitio web con el objetivo de entender su tráfico como punto de partida para optimizar diversos aspectos del mismo.

Capítulo 1

Introducción

1.1. Motivación

En la actualidad se utilizan entornos virtuales como apoyo en la enseñanza mediante los cuales se genera cada vez más datos. La Facultad de Ingeniería de la Universidad de la República tiene el Entrono Virtual de Aprendizaje, conocido por su sigla EVA, basado en la plataforma Moodle. El EVA contiene los datos de las actividades que los alumnos realizan en la facultad. Incluye no solo a cuáles asignaturas está inscrito un estudiante, sino que también almacena los resultados de las asignaturas, de las pruebas realizadas e incluso datos de comportamiento como participación en los foros o los materiales descargados por los estudiantes.

Al mismo tiempo, se han desarrollado herramientas para el procesamiento de datos, lo cual genera nuevas oportunidades en el área educativa. La motivación del proyecto es utilizar estas herramientas para analizar los datos generados por los sistemas educativos para aportar valor a la educación. En particular interesa su aplicación para lograr una menor deserción, al detectar de forma temprana a aquellos estudiantes que tienen una alta probabilidad de abandonar la carrera, así como también de perder un examen. De lograr esto, se podría actuar temprana y eficientemente para minimizar la deserción.

1.2. Objetivos

El principal objetivo de este proyecto es estudiar la viabilidad de la aplicación de técnicas de aprendizaje automático en la educación, con el fin de mejorar la calidad de la enseñanza. En particular, se busca estudiar si es posible predecir el rendimiento académico de un estudiante al cursar una asignatura. De ser posible esta predicción, se podría brindar apoyo particular a los estudiantes con mayor riesgo de reprobar la asignatura con el objetivo de mejorar la calidad de la educación, aumentar la tasa de aprobación de los cursos y mejorar la eficiencia con la que se asignan los recursos educativos.

Se intenta analizar los datos recolectados de una carrera académica para estudiar si se puede obtener información valiosa que le permita a los docentes tomar mejores decisiones e impactar

de forma positiva en el aprendizaje de los estudiantes.

Como objetivo secundario se plantea el desarrollo de un sistema de información que brinde apoyo a los alumnos y docentes a través del análisis de datos. Los datos pueden incluir la información brindada por lo estudiantes, todas las interacciones de los estudiantes con la plataforma de aprendizaje virtual, los resultados de las evaluaciones, entre otros. Además se podría brindar a los docentes una herramienta de análisis estadísticos, para obtener una mejor visión en tiempo real de cada estudiante, con el objetivo de poder actuar de forma proactiva y poder brindarle apoyo a los estudiantes según sus necesidades. Estas herramientas se deberían diseñar configurables para reflejar cambios en los métodos de enseñanza y eventuales ajustes según los comportamientos y resultados obtenidos de cada caso.

Se toma como caso de estudio los cursos de Introducción al Testing y de Introducción al Testing de Performance realizados en el Centro de Ensayos de Software (CES) desde el 2011 hasta el 2018. Se analizará la base de datos con la información de estudiantes. En caso de no ser posible aplicar técnicas de Machine Learning se investigarán otras opciones y se registrarán las necesidades para cada caso.

1.3. Resultados esperados

- Estudio del estado del arte de la aplicación de técnicas de aprendizaje automático a la educación.
- Relevamiento de los datos almacenados en las bases de datos del EVA y de Bedelías.
- Análisis de los datos obtenidos para validar la utilidad de la aplicación de técnicas de aprendizaje automático.
- Análisis y diseño de requerimientos según las necesidades del CES, teniendo en cuenta que se pueda generalizar a otros escenarios.
- Diseño e implementación de un sistema que automatice parte del análisis y ayude en la visualización de los resultados.

1.4. Organización del documento

En el capítulo 2 se realiza una revisión del estado del arte de los principales conceptos relacionados con el proyecto como Machine Learning, Learning Analytics y Educational Data Mining. Además, se detallan algunos ejemplos de usos de sistemas de información educativos y del análisis de datos educativos aplicados.

En el capítulo 3 se detalla la metodología utilizada para realizar los análisis y determinar la viabilidad de la aplicación de técnicas de aprendizaje automático a la educación.

En los capítulos 4,5 y 6 se documenta el análisis y procesamiento realizado sobre los conjuntos de datos del CES. Se incluye la extración y el preprocesamiento de los datos además de

los algoritmos aplicados, tanto de clasificación como de regresión, y se muestran los resultados obtenidos.

En el capítulo 7 se describe la solución propuesta para la implementación de un software de análisis de datos educativos. Se presenta un diseño de un sistema que busca automatizar y simplificar la aplicación de técnicas de Machine Learning a la educación, que contempla entre otros, tareas de recolección de datos, aplicación de algoritmos y presentación de los resultados.

En el capítulo 8 y 9 se plantean las conclusiones y el trabajo a futuro.

Capítulo 2

Estado del arte

2.1. Introducción

Existe una gran variedad de problemas a los cuales se enfrentan los sistemas educativos en la actualidad para poder brindar una educación de calidad a toda la sociedad. Un estudio realizado en 2012 en la Facultad de Ingeniería de la Universidad de la República muestra que la deserción estudiantes, definida como aquellos estudiantes que no han participado en ninguna actividad académica en los 2 últimos años de alguna carrera de ingeniería, alcanza a un 48,2 % de los estudiantes.[4] Este no es un problema únicamente del Uruguay, sino que ha sido documentado en varias partes del planeta, alcanzando el 25 % de los estudiantes que abandonan en el primer año de la carrera hasta el 75 % al cabo de dos años.[5] A su vez, se ha mostrado que un rol activo por parte de los profesores y poder tener una evaluación del rendimiento académico de forma continua tienen un impacto importante para reducir la deserción y mejorar la calidad de la educación en general.[5] [44]

Es de interés poder detectar tempranamente a los estudiantes que están en riesgo de perder un curso o de abandonar, para poder accionar sobre dichos estudiantes y lograr mejorar la educación tanto en aspectos cualitativos como cuantitativos. Es un problema que aumenta su complejidad con la cantidad de estudiantes, ya que se hace más difícil encontrar a los estudiantes con estas problemáticas. Este es un desafío que no es particular de la educación, sino que ha impactado con distintos grados a muchos ámbitos de la industria y la academia, por lo que se han desarrollado nuevas técnicas para resolver este tipo de problemas.

En los últimos años hubo una explosión en la cantidad de datos generados por los dispositivos que la humanidad utiliza directa o indirectamente. Según algunas estimaciones la humanidad produce 2.5 exabytes de datos por día, principalmente generados por usuarios a través de interacciones en plataformas informáticas, como por ejemplo las redes sociales, y por información recolectada por sensores. Esto ha provocado que sea muy difícil obtener información valiosa de estos datos, ya que es una tarea imposible que una persona, o incluso un equipo de personas, procesar en un tiempo razonable dichos datos. Incluso es difícil manejar ese volumen utilizando las bases de datos tradicionales para realizar el análisis.

En conjunto con el aumento de la cantidad de datos generados, la capacidad de cómputo

también ha crecido de forma exponencial, de acuerdo a la Ley de Moore, lo que ha hecho posible utilizar la tecnología para obtener información valiosa de un gran volumen de datos. Tomando en cuenta estos dos cambios importantes, se han comenzado a aplicar técnicas para utilizar una combinación de la enorme capacidad de cómputo disponible con la inmensa cantidad de datos, para poder obtener el mayor provecho y utilidad posible.

Hay varios problemas sobre el procesamiento de estos datos, más allá de su gran volumen. Una característica fundamental suele ser que los datos no tienen una estructura definida y consistente a lo largo del tiempo. Por otro lado, existe una gran heterogeneidad en los datos, al provenir de distintas fuentes. Por ejemplo, un correo electrónico puede tener datos estructurados, como el remitente y la fecha de envío, y tener datos no estructurados, como imágenes adjuntas y el cuerpo del correo. Además, muchas veces hay inconsistencia en los datos y es difícil encontrar y modelar la relación entre ellos. Esto ha provocado que las técnicas tradicionales de análisis de datos tengan que ser rediseñadas por completo.

Debido a esta gran variabilidad en los datos, las bases de datos relacionales, que han sido tan buenas para el manejo de datos estructurados, no han conformado en su capacidad para almacenar y administrar los datos no estructurados. Como reacción a esto se han desarrollado otro tipo de bases de datos, denominadas NoSQL, que buscan obtener mejores resultados en el tratamiento de estos grandes volúmenes de datos no estructurados.

2.2. Movimiento NoSQL

A principios del siglo Google, que ya tenía que lidiar con las problemáticas del gran volumen y velocidad de cambio de los datos al tener que indexar una enorme cantidad de sitios web que cambian todo el tiempo y va creciendo en cantidad.

Se publicaron artículos que comenzaron a establecer las bases de algunos conceptos que se utilizan hoy en día para afrontar estos problemas. En 2003 publicó su sistema de manejo de archivos (GFS, Google File System), en 2004 el algoritmo de procesamiento distribuído MapReduce y en 2006 su base de datos distribuida llamada BigTable. Estas publicaciones, en conjunto con otros avances, maduraron en la comunidad Open Source bajo el proyecto Apache Hadoop.

Otro enfoque utilizado fue el de las bases de datos documentales que guardan datos serializados generalmente en forma de JSON. Se han hecho muy populares como alternativa a las bases de datos tradicionales principalmente debido a que solucionan de forma bastante amigable el impedance mismatch que existe entre las bases de datos relacionales y los lenguajes orientados a objetos. Además, tienen un esquema muy flexible que hace que no haya que hacer grandes cambios a la base de datos cuando se modifica la aplicación que la utiliza.

Estas nuevas bases de datos no han reemplazado a las bases de datos relacionales, que se siguen utilizando en muchísimas organizaciones, pero han facilitado el manejo de grandes cantidades de datos.

Además del cambio en las herramientas utilizadas y en las técnicas para obtener información

útil a partir de los datos, también se han visto modificadas por la explosión en la cantidad de datos a manejar. Una de las técnicas utilizadas es Data Mining, la cual es una disciplina que tiene como objetivo descubrir información no trivial como patrones, a partir del estudio de datos heterogéneos que permita inferir o explicar el comportamiento de los mismos en un determinado contexto.

En conjunto con Data Mining, es muy común utilizar Machine Learning el cual se enfoca en realizar modelos predictivos a partir de los datos. Estas técnicas están siendo muy utilizadas en varios ámbitos por organizaciones y empresas, en un área denominada Bussiness Inteligence, para obtener información valiosa y poder mejorar la toma de decisiones.

2.3. Machine Learning

Machine Learning es una rama de la ciencia de la computación que fue definida por Arthur Samuel en 1959 como el campo de estudio que busca que las computadoras tengan la habilidad de aprender sin ser explícitamente programadas[2]. Una definición más formal establece que un programa de computadoras se dice que aprende de la experiencia E con respeto a una clase de tareas T y con una medida de performance P, si su performance en la tarea T, medida por P, mejora con la experiencia E.

Los algoritmos de Machine Learning comenzaron a ser desarrollados en las décadas de 1950 y 1960 con el objetivo de modelar y analizar grandes volúmenes de datos. Desde esa época se distinguen tres ramas de Machine Learning: Artificial Neural Network (ANN), los métodos estadísticos y el análisis de discriminante. Con el paso del tiempo, las tres ramas de Machine Learning desarrollaron métodos más avanzados. Los métodos estadísticos desarrollaron algoritmos como K-Nearest Neighbors (KNN), análisis de discriminante y clasificadores Bayesianos. El aprendizaje simbólico desarrolló técnicas de árbol de decisión y programas de inducción lógica, mientras que las Artificial Neural Network (ANN) desarrollaron redes neuronales multicapa con alimentación hacia adelante (multilayered feedforward neural network) y propagación del error hacia atrás (backpropagation), los Mapas auto-organizados de Kohonen y las Red de Hopfield como memoria asociativa (Hopfield associative memory).[29]

La mayoría de estos algoritmos fueron desarrollados hace muchos años, pero es hace poco tiempo que Machine Learning ha despertado un interés en la industria en general. Esto se debe principalmente a la gran cantidad de datos que manejan las industrias hoy en día y al aumento del poder de cómputo de los procesadores.

El enfoque de las técnicas de Machine Learning se basa en desarrollar algoritmos que aprendan de un conjunto de ejemplos que especifiquen para cierta entrada cuál debe ser la salida y puedan generalizar estos resultados, para que cuando reciban una entrada nueva, producir la salida correcta.

Algunos de los casos de uso más comunes incluyen al reconocimiento de patrones, como identificar rostros de personas, expresiones faciales, poder interpretar imágenes para detectar objetos se encuentran en ella, el reconocimiento de voz, entre otras.

Otra de las aplicaciones involucra la detección de anomalías, utilizado por ejemplo en la prevención de fraude para detectar transacciones fraudulentas, el monitoreo de redes de computadoras para detectar ataques informáticos y la detección de spam. El tercer gran campo de aplicación tiene que ver con las predicciones, incluyendo a los sistemas de recomendación que predicen los gustos de los consumidores y la predicción del clima[4],[5].

Hoy en día, los algoritmos de Machine Learning se dividen en tres grandes sectores: Supervised Learning, Unsupervised Learning y Reinforcement Learning.

2.3.1. Supervised Learning

Los algoritmos de Supervised Learning son aquellos que reciben un conjunto de datos y se conoce cual es la salida correcta. El algoritmo debe aprender la relación entre esas entradas y salidas conocidas y generalizar dicho resultado para cuando reciba una nueva entrada, poder predecir cuál será la salida.

Normalmente se subdividen en dos categorías, la regresión y la clasificación. En la regresión el algoritmo intenta predecir los resultados con una función continua, mientras que en la clasificación lo hace con una función discreta.

Un ejemplo de un caso de uso de clasificación, es la detección de Spam. El algoritmo recibe un conjunto de emails de los cuales un subconjunto está marcado como spam, y luego cuando se recibe un email nuevo, el algoritmo debe decidir si es spam o no en base a lo que aprendió previamente. Algunos de los algoritmos de clasificación son los clasificador lineal y clasificador no lineal, support vector machines (SVM), KNN, árbol de decisión y Artificial Neural Network (ANN).

Por otro lado, un ejemplo de regresión es el problema de predecir el valor de una vivienda en el mercado, donde el objetivo es determinar el valor exacto.

2.3.2. Unsupervised Learning

Una segunda categoría es la de los algoritmos de Unsupervised Learning. En estos algoritmos el conjunto de datos recibido no tiene una salida o etiqueta correspondiente, y depende del algoritmo agrupar los datos de acuerdo a los atributos de los mismos. El tipo de algoritmos más utilizado es el de clustering, en el que se busca agrupar los datos en conjuntos, denominados clusters, de forma tal que los elementos de un cluster sean más parecidos entre ellos que con los elementos de otro cluster, de acuerdo a determinado criterio.[7]

2.3.3. Redes neuronales

Las redes neuronales son algoritmos que intentan replicar las redes neuronales que existen en los cerebros biológicos. Los primeros pasos en esta área se pueden rastrear hasta 1943, cuando se comenzaron a desarrollar modelos de redes neuronales basados en el conocimiento de

neurología. Estas redes neuronales eran binarias con el objetivo de realizar simples funciones lógicas.

En general, las redes neuronales se definen por cuatro parámetros: el tipo de la neurona, la arquitectura de la red, el algoritmo de aprendizaje y el algoritmo por el cual la información es extraída de la red.

Luego de un período de estancamiento en el avance de las redes neuronales, se desarrolló el método de aprendizaje de back-propagation en 1974, que es una de las aplicaciones de redes neuronales más utilizadas hoy en día

En el ámbito educativo también se ha visto un fenómeno similar en el aumento de la cantidad de datos, generada principalmente por las plataformas de aprendizaje virtual y sistemas de manejo del aprendizaje, como por ejemplo la plataforma Moodle. Es por esto que se comenzaron a investigar el uso de Data Mining y Machine Learning aplicados a la educación para analizar el comportamiento de estudiantes y poder mejorar el rendimiento de los mismos, por ejemplo brindando recomendaciones, mejorando los procesos de enseñanza, entre otros. Debido a esto, se han desarrollado las áreas conocidas como Educational Data Mining y Learning Analytics, las cuales aplican estas técnicas a la educación.

2.4. Learning Analytics

2.4.1. Definición

Learning Analytics tiene sus orígenes en varios sectores como Business Intelligence, Web Analytics, Educational Data Mining y sistemas de recomendación. En 2010 se realiza la primer conferencia internacional de Learning Analytics and Knowledge con el fin de vincular la ciencia de la computación y áreas de Psicología, Sociología, entre otras para fomentar el objetivo en común en favor de las necesidades de los distintos actores. En 2011 se funda la Society for Learning Analytics Research (Society for Learning Analytics Research (SoLAR)), para supervisar las conferencias, desarrollar y fomentar un programa de investigación de Learning Analytics. Ésta sociedad define a Learning Analytics como:

"la medición, recolección, análisis y reporte de datos sobre estudiantes y su contexto, con el propósito de entender y optimizar el proceso de aprendizaje y el ambiente en que éste ocurre."[1]

Se enfoca en recolectar y analizar datos de distintas fuentes para obtener información sobre qué cosas funcionan y cuáles no con respecto a la enseñanza y el aprendizaje. Este análisis busca a las instituciones educativas mejorar su calidad de enseñanza, al actuar sobre los resultados del análisis

2.4.2. Objetivos

Uno de sus objetivos es hacer visible a los usuarios la información contenida en bases de datos educativas, para poder tener un juicio más informado y poder tomar decisiones más acertadas. Se busca que exista una acción posterior a los resultados y que se trace un plan de estudios, personalización y adaptación, así como predicción e intervención sobre las capacidades, con el objetivo de mejorarlas.

Es un campo que ha tenido un desarrollo importante en los últimos tiempos debido a varios factores. En primer lugar, la introducción de plataformas de aprendizaje virtual y sistemas de manejo de aprendizaje, como por ejemplo Moodle, le han brindado a las instituciones educativas de una enorme cantidad de datos. Estos datos pueden ser interacciones que tiene el estudiante con el sistema, datos personales e información académica. En segundo lugar, ha habido un incremento en el aprendizaje en línea que trajo como consecuencia nuevos desafíos. Los estudiantes pueden sentirse desorientados o perder motivación al no tener contacto directo con los docentes y para los profesores es muy difícil identificar a los estudiantes que están teniendo dificultades, así como también evaluar la calidad del aprendizaje de estudiantes individuales en cursos masivos; por lo tanto, es necesario resolver estos desafíos para optimizar el aprendizaje en línea.

Hoy en día se realizan cada vez más cursos a distancia, como los dictados mediante las plataformas MOOC, los cuales generan una gran cantidad de información acerca de los usuarios, tanto de sus datos personales, como de sus interacciones, participación en foros, material consultado, tiempo de respuesta en una prueba, entre otros.

El término MOOC proviene de la sigla en inglés Massive Online Open Courses (Cursos abiertos masivos) y denomina a los cursos dictados a distancia a través de internet para cualquier persona y con un gran número de participantes.

Dispone además de materiales (videos, archivos de lectura), actividades (cuestionarios, evaluaciones), y foros para estudiantes y profesores

2.4.3. Aplicaciones

Algunas de las aplicaciones de Learning Analytics han sido el desarrollo de sistemas para detectar estudiantes en riesgo de abandonar el estudio y aumentar los ratios de retención sobre la inscripción en los organismos educativos (Arnold et al. 2012); herramientas para sugerir opciones a tomar en una carrera (Bramucci & Gaston, 2012); el análisis de las estructuras de las currículas de Universidades (Méndez et al. 2014) y la predicción del rendimiento académico (Antunes, 2010; Essa & Ayad, 2012a; Essa & Ayad, 2012b; Romero & Ventura, 2013).

Debido a la gran variedad de los campos involucrados existen diferentes enfoques, perspectivas y corrientes de investigación, por lo cual el hecho de encontrar puntos en común que favorezca la comunicación entre las partes involucradas, así como también establecer los distintos objetivos particulares y generar otros que sean comunes; ha sido y es un gran motivo de investigación.

Como también sucede en otras áreas que utilizan grandes volúmenes de datos, surge la necesidad de analizarlos y el cuestionamiento de cómo sacar el mayor provecho de los mismos para contribuir o dar apoyo a los involucrados en la enseñanza. Ésta disciplina pone foco principalmente en tres puntos de vista distintos, el alumno, el profesor, y los que están encargados de la administración o gestión.

Normalmente hay tres tipos de datos, los cuñantitativos (cantidad de asistencia a clase, cantidad de intentos de las pruebas, etc.), los cualitativos (notas de evaluación, encuesta del nivel de satisfacción con escala del 1 al 5, etc.) y los datos de interacciones sociales de los cuales se pueden crear grafos de red y los que permitan un análisis de calidad para medir la calidad de aportes e interacciones.[14][23]

2.5. Metodología

Un modelo de referencia de Learning Analytics es el que se muestra en la Figura 2.1, que contempla lo mencionado anteriormente basado en cuatro dimensiones.

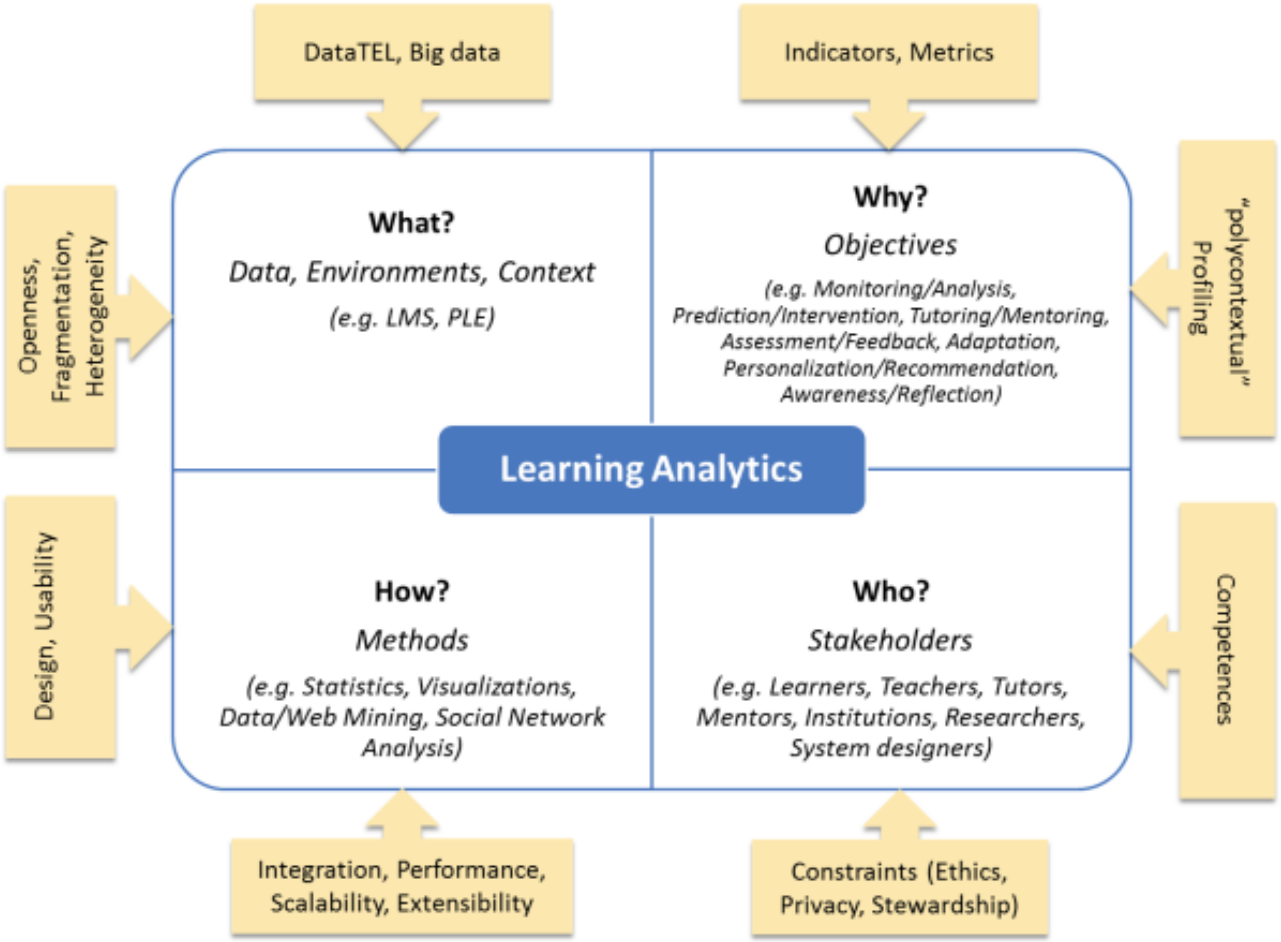


Figura 2.1: Modelo de referencia para un proyecto de Learning Analytics

2.5.1. Datos, entorno y contexto

El '¿Qué?' involucra a los datos, al entorno y al contexto. Su origen pueden ser sistemas centralizados, como por ejemplo los Learning Management System (LMS), que almacenan información de la actividad de sus usuarios, o de sistemas distribuidos en donde los datos se generan utilizando diferentes medios, en un espacio, tiempo y formato distinto, tanto en contextos de educación formal como también informal. Una gran cantidad de datos es producida por los usuarios provenientes de sistemas distribuidos, denominados entornos de aprendizaje personal.

Cabe destacar el desafío que presenta agrupar e integrar datos de múltiples fuentes heterogéneas, con distintos formatos, para obtener un conjunto de datos que refleje las actividades del estudiante en todos sus ámbitos. Además de la agrupación de los datos, hay que prestar atención a la calidad de los datos para que sean relevantes para el análisis, consistentes para tener información similar de todos los estudiantes y confiable para que los datos sean certeros y aporten valor a los análisis.

2.5.2. Actores

El '¿Quién?' se centra en los actores de la aplicación de Learning Analytics. Estos pueden ser simplemente los estudiantes y profesores, pero también puede, por ejemplo, incluir a investigadores o directores de las organizaciones educativas. En este cuadrante el desafío más grande es poder ofrecer un resultado de valor a los distintos actores, teniendo en cuenta que cada uno tiene distintos intereses, perspectivas y expectativas sobre los resultados de la aplicación Learning Analytics.

Otro desafío consiste en mantener la privacidad de los datos, principalmente de los estudiantes, ya que los datos pueden contener información valiosa y personal que su dueño puede no querer hacer pública. Para proteger este derecho a la privacidad, últimamente se han creado y modificado muchas leyes en varios países y organizaciones internacionales para lidiar con este asunto. A su vez, se ha planteado la necesidad de poder compartir los datos recolectados por distintas organizaciones para permitir a terceros utilizar dichos datos y poder tomar mejores decisiones.

Esto ha generado el movimiento de Open Data, para incentivar a las organizaciones a compartir sus datos y tiene particular interés en el área científica ya que los datos generados en un experimento pueden colaborar en el avance de otras áreas. Cabe destacar la problemática que surge al querer compartir datos que contengan información personal ya que existen leyes que la protegen.

Para poder proteger los datos personales y a la vez poder compartirlos, los datos deben ser anonimizados previo a compartirlos. Esto significa remover o encriptar cualquier tipo de dato que pueda ser vinculado a una persona específica, por ejemplo, en Uruguay existe una Ley de Protección de Datos Personales [10] por lo que se deben quitar la cédula de identidad y nombre completo, entre otros.[26]

2.5.3. Objetivos

El '¿Por qué?' se refiere a los objetivos del estudio de Learning Analytics, los cuales pueden ser diferentes para cada uno de los distintos actores del sistema. Uno de estos objetivos puede ser el monitoreo de las actividades de los estudiantes, para generar reportes para que los profesores y los directores de las instituciones educativas puedan tomar mejores decisiones. Este monitoreo y su análisis, pueden ayudar a los profesores a mejorar sus métodos de enseñanza, teniendo un impacto directo en el aprendizaje de los estudiantes.

Otro objetivo suele ser la predicción del rendimiento de los estudiantes. Esto permite intervenir proactivamente para sugerir acciones a tomar cuando se logra predecir que un estudiante tendrá dificultades para aprobar un curso por ejemplo, o que está por abandonar los estudios.

Además existe un objetivo que normalmente se plantea que consiste en personalizar los métodos de enseñanza que mejor se adecuan a cada estudiante. Por ejemplo, un modelo puede recomendar un orden distinto de los temas para aprender, en base al conocimiento de cada estudiante, o un enfoque diferente de acuerdo a la facilidad de aprendizaje y preferencia por parte de cada estudiante de los distintos métodos de enseñanza o recursos disponibles.

Estos son algunos de los objetivos que se pueden plantear para llevar a cabo un estudio de este tipo aunque no son los únicos. Cabe destacar que un desafío que se presenta a la hora de plantear los objetivos, es la forma de medir el resultado esperado, más allá de basarse solo en el resultado final del curso.

Por otro lado, para desarrollar algunos puntos como el de recomendaciones, es fundamental sortear el problema de crear un perfil de cada estudiante para poder ofrecer recomendaciones adecuadamente.

2.5.4. Técnicas y métodos

Por último, el '¿Cómo?' aplica a las diferentes técnicas utilizadas en Learning Analytics para obtener información a partir de los datos educativos. Una de las técnicas más utilizadas es la estadística. La mayoría de los LMS ofrecen reportes con estadísticas básicas de las interacciones de los estudiantes con el sistema, como el tiempo que están conectados, la frecuencia con la que participan en los foros, entre otros. Se puede obtener, por ejemplo, información de cada estudiante en relación a sus compañeros.

Uno de los desafíos de los métodos estadísticos se encuentra en la dificultad de interpretar los reportes, para lo cual una posible solución es mostrar visualmente la información, normalmente en forma de gráficas. Aquí el desafío consiste en definir cuál tipo de gráfica es el más adecuado para alcanzar el objetivo planteado.

Otra de las herramientas más utilizadas es el análisis de las redes sociales. Normalmente se representa mediante un grafo a los estudiantes y sus interacciones, para poder extraer información, un ejemplo del mismo se puede ver en la Figura 2.2. Los nodos representan a los estudiantes y las aristas a una interacción entre dos estudiantes, como por ejemplo la pregunta y respuesta en un foro de discusión. A partir de estos grafos se pueden realizar diversos análisis

como por ejemplo al relacionarlo con los niveles de aprendizaje alcanzados de cada participante. Luego, de las interacciones registradas se pueden buscar patrones que expliquen la incidencia de las mismas en los resultados.

Por otro lado se podría medir la calidad de los aportes de usuarios en los foros, mediante valoraciones o votos que pueden realizar los demás usuarios y analizar cómo influyen los mismos en el aprendizaje tanto para el emisor como para los receptores de los aportes.

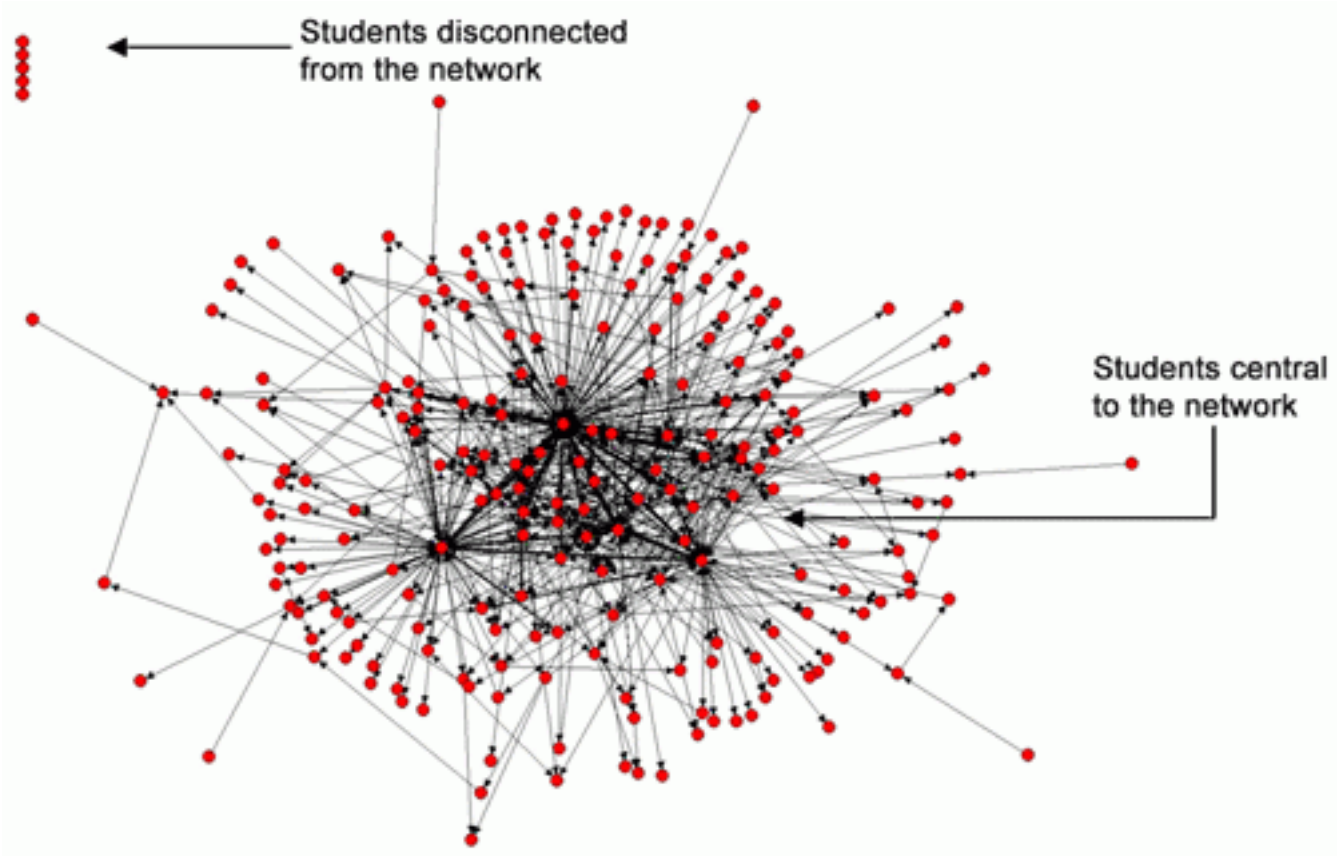


Figura 2.2: Grafo de interacciones entre estudiantes en redes sociales

2.6. Educational Data Mining

Data Mining, también llamado Knowledge Discovery in Databases (KDD), es la disciplina encargada de la extracción de patrones y de información de grande volúmenes de datos.(Klosgen & Zytkow, 2002). Una organización puede haber recolectado muchos datos y, sin embargo, no haber obtenido ningún valor real. Para obtener ese valor, hay que extraer información de ese conjunto de datos. El objetivo de Data Mining se puede resumir como obtener información a partir de grandes volúmenes de datos.

No es un proceso trivial, ya que los datos pueden no ser homogéneos, lo cual significa que no todos deben tener las mismas características. Además, estos pueden ser desestructurados en forma de imágenes, texto, audio, video, entre otros, lo que dificulta su tratamiento. Por último,

el enorme volumen y la gran velocidad a la que éstos son generados y recolectados agrega una complejidad adicional al análisis.

En el siglo XXI hubo un incremento en el uso de las plataformas de aprendizaje virtual lo que provocó una explosión en la cantidad de datos sobre el aprendizaje de cada estudiante y la creación del campo de estudio de estos datos, Educational Data Mining. Se centra en desarrollar métodos para explorar los datos que se obtienen de contextos educativos, y en aplicar dichos métodos para comprender mejor a los estudiantes y sus métodos de aprendizaje.[11] Desde un punto de vista técnico, el objetivo de Educational Data Mining es analizar los datos educativos para resolver los desafíos de la educación y comprender el contexto en el que los estudiantes aprenden.[18]

Educational Data Mining es la disciplina que se ocupa de desarrollar métodos para explorar los datos que surgen del ambiente educativo y usar los métodos para comprender mejor a los estudiantes y al ambiente en el que éstos aprenden. Las técnicas de Educational Data Mining provienen de varios campos como el propio Data Mining, Machine Learning, estadística, visualización de datos, entre otras.

La información que se obtiene de la aplicación de las técnicas de Educational Data Mining puede ser de gran utilidad para los profesores para tener una devolución más temprana que las pruebas finales del estado del aprendizaje de cada estudiante. Tener esta información puede permitir evaluar la propia estructura del curso dictado buscando optimizar el mismo para que sea más efectivo. También se puede clasificar a los estudiantes en grupos para que el profesor pueda asistir a cada grupo según sus necesidades y preferencias puntuales.

Por otro lado, la información puede ser utilizada directamente por los estudiantes. El resultado del análisis de los datos puede ser la recomendación de pasos a seguir para el estudiante, como material a estudiar o ejercicios a realizar, buscando mejorar el aprendizaje del estudiante.

En la enseñanza tradicional, los profesores pueden obtener información de la calidad del aprendizaje de los estudiantes en base a la interacción personal con los mismos. Es una fuente de información que en la enseñanza a distancia en gran medida se pierde. Hay varios estudios de Educational Data Mining aplicado a la educación presencial o tradicional, como [Sanjeev and Zytkow (1995)] que se enfocó en el análisis de las razones de los estudiantes para inscribirse a distintas asignaturas, o el trabajo de [Ma et al. (2000)] en el que buscan identificar a los estudiantes con dificultades para recomendarle cursos de nivelación.

Cabe destacar que el dominio de análisis, los datos, el proceso y los objetivos de Learning Analytics y Educational Data Mining es muy similar. Sin embargo, las técnicas utilizadas en ambos campos son muy distintas. En Educational Data Mining se utilizan herramientas típicas de Data Mining como clustering, clasificación y association rule mining para apoyar tanto a profesores como a estudiantes a analizar el proceso de aprendizaje. Las técnicas de Clustering son aquellas que intentan encontrar y agrupar datos que tengan similitudes en lo que se conoce como un clúster. En el caso de la educación podría ser identificar estudiantes con características similares para reforzar algunas áreas del conocimiento en particular.[16]

Más allá de las dificultades técnicas de tratar con tantos datos, hay otros desafíos que hay

que tener en cuenta. En primer lugar se encuentra la privacidad de los datos, ya que los mismos pueden contener información valiosa y personal que su dueño puede no querer hacer pública. Para proteger este derecho a la privacidad, últimamente se han creado y modificado muchas leyes en varios países y organizaciones internacionales para lidiar con este asunto.

En segundo lugar, al poder extraer información importante de los datos, muchas veces se ha planteado la necesidad de compartir la información recolectada por distintas organizaciones para permitir el acceso a terceros y poder tomar mejores decisiones.

Esto ha generado el movimiento de Open Data, para incentivar a las organizaciones a compartir sus datos y tiene particular interés en el área científica ya que los datos generados en un experimento pueden colaborar en el avance de otras áreas.

Cabe destacar la problemática que surge al querer compartir datos que contengan información personal, protegida por ley. Para poder proteger los datos personales y a la vez poder compartirlos, los datos deben ser anonimizados previo a compartirlos. Esto significa remover o encriptar cualquier tipo de datos que pueda vincular a una persona específica con los datos, por ejemplo, en Uruguay puede significar quitar la cédula de identidad y nombre completo, entre otros, antes de compartir datos.[26]

Una manera de intentar evaluar el aspecto cualitativo de la enseñanza de los estudiantes en el ámbito digital es a través de las interacciones del estudiante con las plataformas de enseñanza. Ejemplos de esto son los resultados de pruebas realizadas en línea, pero también otro tipo de información como el tiempo que le toma al estudiante responder las preguntas o si cambia de decisión mientra se encuentra realizando las pruebas. También hay otro tipo de interacciones como la calidad y cantidad de interacciones en foros de discusión, el tiempo que mira los videos de los cursos, entre otros.

Una de las maneras más comunes para registrar este tipo de interacciones es con los logs que muestran todos los accesos de los estudiantes a las plataformas. Entre los distintos tipos de logs, se destacan los correspondientes al cliente y al servidor, descritos a continuación. Por un lado, los logs del servidor contienen información acerca del recurso accedido, fecha de acceso y cuál fue la respuesta. Los formatos más utilizados son CLF y ELF. Por otro lado se encuentran los logs del cliente, en general uno por estudiante o sesión web, que contienen información de la interacción de un usuario con la plataforma. Usualmente son implementados con el lenguaje de programación Javascript, utilizando las cookies de los navegadores.

El uso de logs para registrar las interacciones de los usuarios tiene limitaciones ya que no registra los datos relacionados a las actividades del estudiante y su contexto. Son más eficientes para reconocer una computadora específica pero no tanto para reconocer una persona. Además, limitaciones técnicas de los navegadores, como el uso de caché, pueden provocar que no se registre información en los logs de los servidores y así evitar su análisis.

Se han propuesto varias alternativas al uso de logs para resolver los inconvenientes anteriormente mencionados en la forma de estándares para los sistemas de aprendizajes.

2.7. Estándares de sistemas de aprendizaje

En sus inicios, los sistemas de gestión de aprendizaje utilizaban formatos propietarios para distribuir contenidos. Esto provocaba varios problemas, debido a la heterogeneidad entre los contextos y tecnologías. Algunos de los problemas eran la imposibilidad de reutilizar los recursos creados para un sistema en otro, la interoperabilidad entre distintas herramientas, la durabilidad en el tiempo ante cambios de versiones, entre otros.[43]

2.7.1. Historia

Para solucionar estos problemas en 1987 el Aviation Industry Computer-Based Training Committee (AICC) crea un estándar, llamado de la misma forma que la organización, con el objetivo de abordar los problemas mencionados.

2.7.2. SCORM

Sharable Content Object Reference Model (SCORM) es un conjunto de estándares técnicos para software educativo, desarrollado por ADL,[39] que establece las interfaces que tienen que implementar las plataformas de aprendizaje para poder comunicarse y ser interoperables. En particular, establece cómo los LMS se pueden comunicar entre ellos, para mantener un estándar a través de toda la industria. Se basa en crear unidades de material educativo que puedan ser compartidas a través de diferentes sistemas, llamados Sharable Content Objects (SCO). Además, define cómo crear los SCOs para que puedan ser reutilizados por otros sistemas. Las últimas palabras de la sigla, Reference Model, se refieren a que no es un estándar nuevo, sino que reconoce a los estándares anteriores que resolvieron partes de los problemas planteados e indica cómo utilizarlos de forma conjunta.

El principal beneficio de SCORM es la interoperabilidad y es el estándar de facto de la industria en ésta área. Al generar contenido educativo es deseable que éste pueda ser interpretado y almacenado en diferentes LMS. Además, al crear un nuevo LMS interesa poder incorporar contenido de otros LMS para lo cual SCORM brinda facilidades de integración.

2.7.3. Experience API

Uno de los problemas que se comenzaron a detectar con SCORM, es que este sólo permite registrar las actividades formales de los estudiantes que típicamente ocurren en los LMS, como las pruebas. No se almacena información sobre otras actividades directamente relacionadas con el aprendizaje como la asistencia y participación en clase, el estudio en grupos de estudiantes, la lectura de material complementario, entre otros.

El modelo de aprendizaje 70/20/10 establece que el 70 % del aprendizaje se da en el aprendizaje por experiencia, el 20 % en la interacción con otros estudiantes y profesores, y el 10 % a través de la lectura de artículos, libros y el aprendizaje en clase. Más allá de que este modelo ha sido cuestionado en el último tiempo debido a que fue planteado en la década del 80 y se

duda de la veracidad de esos porcentajes en el contexto actual con la proliferación de cursos online y otras herramientas, hay un consenso en que hay una gran cantidad de ámbitos de aprendizaje que se manifiestan por fuera de la educación formal. Es en ese contexto en donde SCORM es particularmente ineficiente, por lo que se ha desarrollado un nuevo estándar que permita registrar no solo el aprendizaje formal sino también todas las actividades realizadas y tener una visión más integral del aprendizaje.

Experience API (xAPI) es un estándar que busca registrar todas las actividades realizadas por los estudiantes como la asistencia a los cursos, la lectura de material, las interacciones entre estudiantes y entre estudiantes y profesores, entre muchas otras. Busca también estandarizar un formato universal de contenido que pueda ser utilizado para registrar estas actividades por distintas fuentes de información. Su objetivo es tener una visión integral de toda la experiencia del aprendizaje del estudiante para poder comprender cómo se relaciona con su rendimiento académico.

Los registros de las actividades de xAPI se almacenan en un formato que refleja una acción realizada por un actor sobre un objeto, se puede ver un ejemplo en la Figura 2.3. Además, se pueden almacenar datos del contexto como puntajes o cualquier otro dato que sea relevante. El actor es el usuario principal de la actividad, el verbo es la acción realizada y el objeto es sobre qué se hizo la actividad. El objeto puede ser otro actor, un libro, una clase u otra actividad previa por ejemplo.

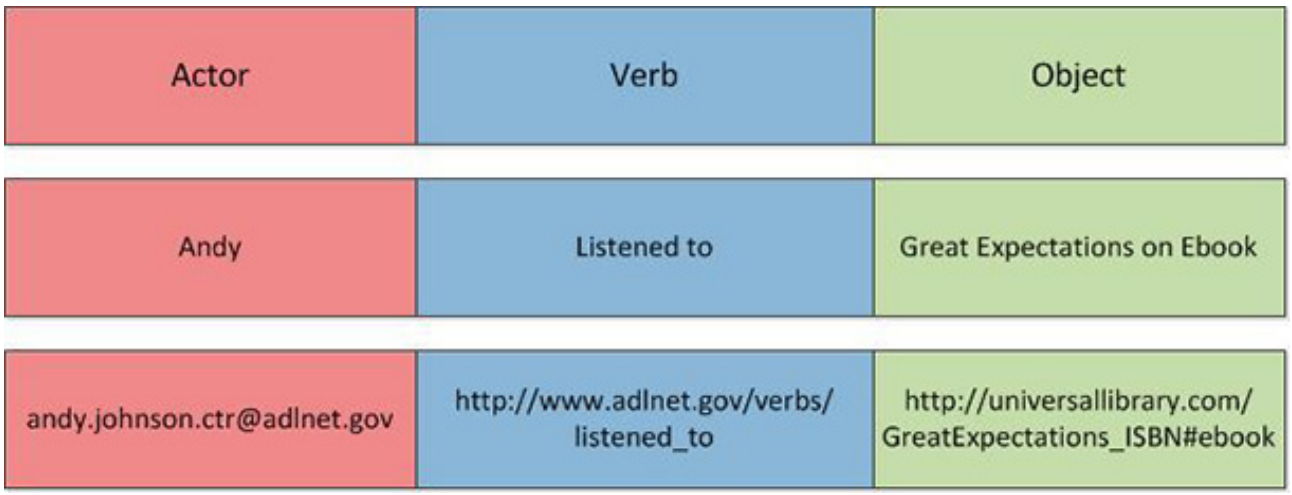


Figura 2.3: Ejemplo de una actividad registrada como oración según el estándar xAPI

El actor se identifica por un identificador único que puede ser asociado a una persona, como la cédula de identidad o una dirección de correo electrónico.
El verbo es la acción realizada durante la actividad de aprendizaje. xAPI no especifica ningún verbo en particular, sino que define cómo crear verbos para que cualquier comunidad pueda establecer verbos que sean relevantes. Los verbos son una propiedad importante para identificar las actividades por lo que su uso correcto y consistente dentro de una comunidad es muy importante. Por ejemplo, si un proveedor de actividades usa el verbo mirar y otro el verbo ver para la misma actividad relacionada con videos, puede generar confusión y bajar la efectividad

del análisis posterior. El objeto es el que cumple la función de indicar qué se hizo en la actividad. Otra denominación es que identifica el objetivo del verbo que el actor realizó.

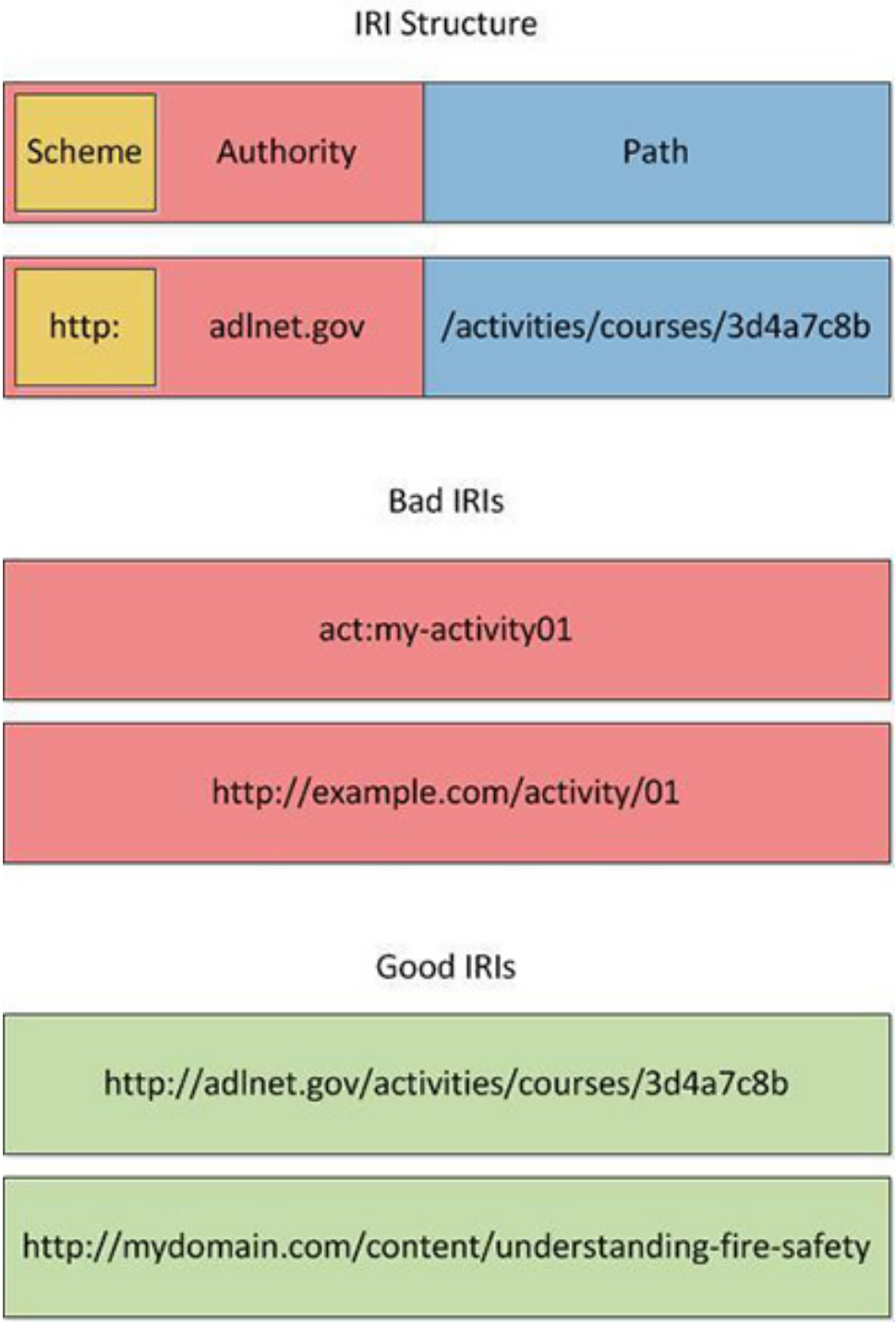


Figura 2.4: Estructura de una IRI

El tipo de objeto generalmente es una actividad pero también puede ser un actor o una experiencia. Por ejemplo, en la experiencia "María aprobó el examen de matemáticas" el objeto sería el examen de matemáticas. Un actor se utiliza para capturar las interacciones entre personas. Por ejemplo, "Juan le mandó un mensaje a María". Por último, el tipo experiencia se utiliza para reflejar acciones sobre experiencias previas, como por ejemplo, "María respondió

el mensaje que le mandó Juan".
Además de estas tres entidades, xAPI guarda propiedades sobre el contexto en que ocurre una actividad, como por ejemplo el puntaje que obtuvo un estudiante en un examen.

Para asegurarse de la unicidad de los objetos y verbos y de ser consistentes en su uso en toda la comunidad educativa (o al menos para un proyecto), se utilizan las IRI, las cuales son similares a las URLs. En la Figura 2.4 se describe su estructura.

Las actividades son guardadas en un Learning Record Store (LRS), que puede ser una parte integral de un LMS con soporte para xAPI o un sistema aparte. El LRS es responsable de la implementación de las APIs para recibir la información y de la persistencia de los datos.

2.8. Casos de estudio

2.8.1. SmartKlass

Para aplicar Learning Analytics uno de los complementos disponibles para Moodle es SmartKlass que es una aplicación desarrollada por la empresa Klass Data, descrita a continuación, que analiza el comportamiento de los estudiantes para detectar los que están más atrasados en el curso, brindar recomendaciones, entre otros.[33]

SmartKlass es una aplicación desarrollada por Klass Data, la cual es una empresa privada, especializada en el desarrollo de herramientas de Learning Analytics que permitan mejorar los procesos de enseñanza y aprendizaje, tanto en contextos de educación formal como informal.[27]

Su objetivo es mejorar significativamente el aprendizaje de estudiantes y hacer accesible a cualquier compañía o institución educativa la herramienta y el valor provisto por el análisis del aprendizaje.

Está compuesto por un módulo de backend, encargado del análisis de los datos y una interfaz de usuario responsable de la recolección y presentación de los mismos.

El backend brinda una API mediante la cual se puedan enviar los datos que el usuario crea conveniente analizar. La API está implementada según el estándar xAPI, también denominado Tin Can API, el cual establece el formato de datos a recolectar provenientes de un entorno educativo, tanto virtual como presencial, permitiendo conectar cualquier plataforma digital o dispositivo con SmartKlass para obtener información de un grupo de actividades o acerca de la experiencia que tiene una persona. Éste módulo tiene la capacidad para administrar grandes volúmenes de datos y analizarlos mediante algoritmos de Inteligencia artificial.

Luego la interfaz de usuario provee paneles de administración que muestran los resultados de los análisis mediante gráficos, tablas, etc., para los profesores, estudiantes y responsables de la institución para mejorar los procesos de aprendizaje. Hasta el momento el único módulo de recolección de datos que han implementado es la extensión para el entorno de aprendizaje Moodle descrito anteriormente.[32]

Los paneles de administración brindan métricas extraídas de buenas prácticas educativas y estadísticas. Por ejemplo puede mostrar los estudiantes que están en riesgo de quedarse atrás

en el curso o también los que presenten una falta de motivación que pueda provocar un bajo desempeño o que abandonen el curso.

Cada perfil de usuario tiene su propio panel de administración adaptado a sus necesidades y con información personalizada. La personalización busca que los involucrados focalicen su atención en dónde más se precise y así mejorar la calidad de la enseñanza y aprendizaje. El objetivo consiste en que el profesor brinde el apoyo requerido al estudiante, en los puntos que le esté costando más entender y adquirir, actuando como guía durante todo el ciclo. También debe detectar los alumnos que avancen más rápido, para poder brindarles el nivel de aprendizaje adecuado a cada uno, respetando el tiempo que le implique hacerlo.

Esta interfaz, además de mostrar los resultados, es responsable de recolectar los datos y enviarlos al backend, encriptando la identificación del estudiante garantizando así su privacidad, para su almacenamiento y posterior procesamiento con el fin de brindar recomendaciones a los usuarios.

La extensión para Moodle es gratuita y de código abierto con licencia GNU GPL versión 3 o superior. Está disponible para instalar desde el panel de extensiones de Moodle. Por otro lado el módulo de procesamiento o backend no está disponible para instalar y se desconoce su código fuente. En el caso de que la plataforma sea utilizada en la escuela se busca que los padres sean los que utilicen el panel del estudiante para brindarle al mismo los contenidos que el sistema y ellos crean más convenientes.

Los desarrolladores de SmartKlass brindan los siguientes resultados obtenidos al utilizar su plataforma.

En las pruebas iniciales que realizamos para validar el beneficio de nuestra plataforma, logramos mejorar en un 60 % los resultados de los alumnos. Klass Data está continuamente afinando la plataforma de Learning Analytics y adaptándola a diferentes entornos, de manera que en los próximos años podremos ver resultados aún mejores.[32]

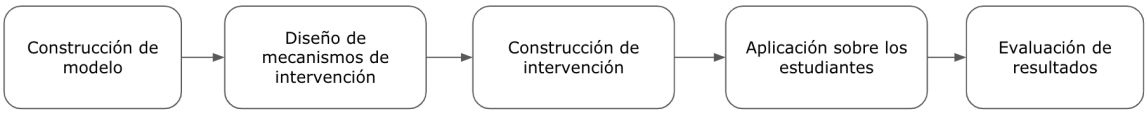
Capítulo 3

Metodología

En este capítulo se detallan los pasos que se tomaron para la generación del modelo predictivo del resultado académico de los estudiantes, así como la metodología utilizada en base al modelo de referencia de Learning Analytics detallado en la sección anterior.

3.1. Introducción

Hay varios casos de estudio en la literatura acerca de la aplicación de técnicas de Machine Learning y Data Mining a la educación. En general el foco de las investigaciones es en la personalización de la educación. Para ello, el proceso normalmente consiste en cinco etapas claramente definidas. En primer lugar, se busca utilizar el histórico de datos para construir un modelo que permita predecir el comportamiento de los individuos. Luego se diseña un mecanismo de intervención sobre los individuos con el objetivo de mejorar el comportamiento. En tercer lugar, se intenta que las predicciones realizadas del primer punto se hagan en tiempo real, para poder aplicar las intervenciones diseñadas en el segundo punto cuando todavía pueden tener un efecto en el comportamiento y rendimiento de los estudiantes. A continuación, se usa la predicción para dar apoyo a los estudiantes que lo necesiten y, por último, se evalúa si la intervención fue beneficiosa y correcta. Luego de terminar el quinto paso, es posible volver a iterar sobre esta solución con el objetivo de mejorar los resultados obtenidos. En el siguiente gráfico se muestran los distintos pasos del proceso.



Un estudio de la Universidad de Harvard es un ejemplo de un proyecto que abarcó los cinco pasos mencionados anteriormente. Allí se realizó un análisis del histórico de datos de una plataforma de MOOCs para dos cursos con más de 40.000 estudiantes, y se diseñó un modelo para predecir cuáles estudiantes no iban a culminar el curso. Luego, en el siguiente curso buscaron en tiempo real estudiantes propensos a abandonarlo y les enviaron proactivamente una encuesta por correo electrónico para consultarles sobre las razones de su falta de motivación.

Esto provocó un resultado beneficioso, al aumentar en un 1 % la cantidad de estudiantes que continuaron el curso por el solo hecho de recibir la encuesta.[45]

Este proyecto se centra en el primer paso de este proceso de cinco etapas. Para esto, se toma como referencia el modelo sobre Learning Analytics que se describió en la sección del estado del arte.

3.2. Datos, entorno y contexto

En referencia al '¿Qué?' se cuenta con las bases de datos del LMS (Moodle) y de Bedelías de la carrera de testing del Centro de Ensayo de Software (CES). Ambas se pueden combinar ya que comparten los identificadores de los estudiantes y los cursos, permitiendo que se pueda extraer información de la actividad de los estudiantes de forma sencilla. La base de datos de Bedelías contiene datos enfocados en la carrera, como los cursos, notas y algunos datos personales de los estudiantes, mientras que la base de datos del LMS contiene datos en común y además otros datos como las preguntas en los foros de discusión y de la actividad de los estudiantes. Se cuenta con datos para los cursos desde 2014 a 2017.

No se pudo obtener los datos de los logs de acceso del LMS debido a que se eliminan del sistema todos los años y no se guarda un respaldo de los mismos. Estos podrían agregar información como el tiempo en que un estudiante está realizando actividades, cuantas veces accede al LMS y muchos otros datos del comportamiento en la plataforma que tienen un valor potencial para realizar un análisis más completo del estudiante. Cabe destacar que si bien no hubo grandes dificultades para cruzar los datos entre las bases de datos de Bedelías y del LMS, agregar otras fuentes de información como los logs de acceso, podría significar un desafío complejo. Más adelante se comentará sobre una propuesta de diseño para una arquitectura de sistemas orientada a resolver este potencial problema.

3.3. Objetivos

La siguiente pregunta planteada es el '¿Por qué?' del análisis. Con ella se busca responder cuál es el objetivo del estudio. Como se discutió anteriormente, hay una gran variedad de objetivos que se plantean normalmente en estudios similares. En este caso, el objetivo es el de predecir el rendimiento académico de los estudiantes con el fin de poder detectar alumnos en riesgo, intervenir lo más tempranamente posible y poder tomar las acciones que el cuerpo docente entienda para que el estudiante mejore su rendimiento.

Para ello se busca predecir si un estudiante determinado aprobará o reprobará el curso. También se incluye la categoría de los estudiantes que abandonan el curso como caso particular de aquellos que reprueban, ya que se entiende que las causas de abandonar un curso pueden ser distintas a las de reprobar un examen. A su vez, no solo se intenta la predicción de cual de las categorías descritas anteriormente estará un estudiante, sino que se busca predecir la nota exacta de la prueba final.

3.4. Público objetivo

Por el lado de '¿Quién?' se plantea la interrogante de quiénes serán los interesados en el análisis a realizar. Esta pregunta no tiene una única respuesta ya que hay varios actores que pueden estar interesados en los resultados.

- Para los docentes el estudio es de interés para poder brindar un mejor apoyo a los estudiantes con más dificultades lo más tempranamente posible y disponer de suficiente tiempo para que éstos puedan mejorar su rendimiento.
- Para los estudiantes esta información puede ser útil para que por ejemplo puedan proactivamente dedicar más horas al estudio del material, o cambiar un método de estudio que no les esté dando buenos resultados.
- Para los directores del centro de enseñanza podrían manejar los recursos de forma más eficiente para que los estudiantes en riesgo tengan un apoyo más directo que el resto, o por ejemplo se puede colocar a los mejores profesores a dar las clases de dichos estudiantes.

3.5. Métodos y técnicas

Por último, se plantea el '¿Cómo?'. Esta pregunta hace referencia a cómo se hará el análisis y qué métodos se aplicarán. Para el alcance de este proyecto, se decidió utilizar técnicas de Data Mining para la recolección y preparación de los datos, Machine Learning para el análisis de los mismos y visualización de gráficas para analizar los resultados.

Se utilizó el lenguaje de programación Python para el tratamiento de los datos, para el entrenamiento y creación del modelo predictivo. Esta decisión se debió a la abundante documentación disponible y a la gran comunidad de desarrolladores que hay, por lo que facilita el aprendizaje del mismo y la resolución de problemas. Además, se utilizó la librería $Scikit - Learn$. Esta librería cuenta con una gran variedad de algoritmos de aprendizaje automático ya implementados así como funciones para entrenar, evaluar y validar modelos.

Para la extracción de los datos se utilizaron consultas SQL sobre las bases de datos MySQL a disposición y Microsoft Excel para visualizar los resultados de forma más amigable. El preprocesamiento de los datos se realizó utilizando Python.

En cuanto a las técnicas de aprendizaje automático, se decidió por utilizar aprendizaje supervisado ya que es el método que mejor se adapta para las necesidades del proyecto. Se tomaron dos enfoques, uno aplicando técnicas de clasificación y otro de regresión. Las mismas serán detalladas más adelante.

3.6. Proceso

A continuación se detallan los pasos que se siguieron para cumplir los objetivos de realizar los modelos predictivos.

En primer lugar se extrajeron los datos de las distintas fuentes de datos a disposición. Este paso consistió de varias etapas. Se comenzó por un estudio de la literatura académica relevante con el objetivo de identificar cuáles son los datos más importantes que conviene extraer para lograr una buena predicción. A continuación, se analizaron las bases de datos a disposición para entender que información se puede extraer de allí para luego, en base a lo relevado de la literatura, obtener los datos más importantes. Como el objetivo de las predicciones es sobre una asignatura particular, se definió el curso a analizar tomando las sugerencias de los tutores sobre la idoneidad del curso a predecir y la cantidad de datos de cada curso.

Luego se procedió a la curación de estos datos aplicando diversas técnicas de preprocesamiento de datos. El objetivo de este paso es transformar los datos extraídos en el paso anterior, para dejarlos preparados para los modelos predictivos. Busca asegurar la calidad de los datos, así como la consistencia de los mismos y que se presenten en el formato adecuado para poder ser consumidos por los algoritmos.

Luego se realizó una etapa donde se entrenan los modelos e intenta predecir los resultados. Aquí se realizaron dos proceso independientes. Por un lado, se realizó un modelo de clasificación que busca categorizar a los alumnos en dos categorías, según si el mismo aprueba o no el curso. Por otro lado, se entrenó un modelo de regresión que intenta predecir la nota exacta de un estudiante. En ambos casos, se toma como entrada los datos preprocesados del paso anterior, y se realizaron varias pruebas con distintos algoritmos y modificando los parámetros, buscando obtener el mejor resultado posible. Una vez que se validan los resultados de los modelos, se retroalimenta el proceso buscando mejorar los resultados en cada iteración.

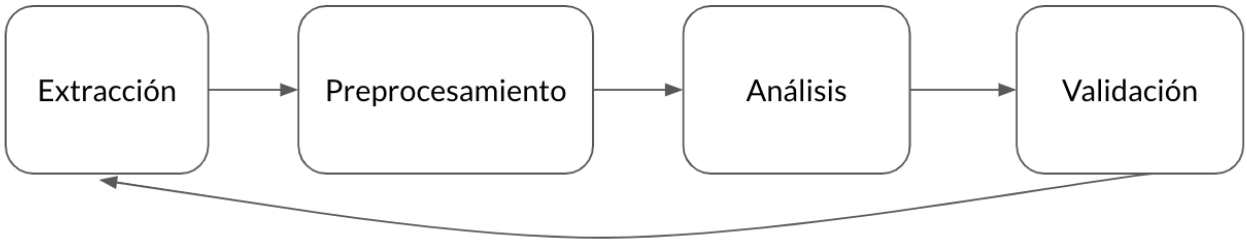


Figura 3.1: Pasos de la construcción del modelo

Capítulo 4

Extracción de datos

4.1. Introducción

Una de las primeras preguntas a la que hubo que enfrentarse fue en relación a qué datos se necesitan obtener para poder predecir el rendimiento académico de los estudiantes. Es una pregunta cuya respuesta condiciona en gran medida los resultados de cualquier análisis, ya que según la información que se disponga, diferirá significativamente los resultados del mismo. A continuación se detalla el análisis que se hizo de la literatura sobre otros proyectos que abordan una problemática similar, con el objetivo de entender la naturaleza de los datos que nos

4.2. Relevamiento de otros proyectos

No existe un consenso en la comunidad académica sobre cuales son las características que más influyen en el rendimiento de los estudiantes. Por un lado, el estudio de A.Harris sugiere que el ambiente académico y la gestión de la universidad juega un rol preponderante en el rendimiento.[35] Mientras que por otro lado, un estudio de la Universidad de Concordia, en Portland, Oregon, plantea que el rol del profesor es la variable más importante.[42]

A pesar de la falta de consenso, se observa en la literatura la existencia de varios casos de estudio, que se mencionan a continuación, en los que se obtienen resultados positivos al analizar conjuntos de estudiantes con el objetivo de predecir su rendimiento académico. Tomando como referencia estos estudios con sus respectivos conjuntos de datos, se observó que existe una gran heterogeneidad y por lo tanto es de gran interés realizar un análisis de los mismos.

En primer lugar, se analizó el conjunto de datos del estudio de Cortez publicado en 2008. El mismo cuenta con datos de 649 estudiantes de dos liceos de Portugal y se dispone del rendimiento académico en las asignaturas de matemáticas y portugués.

En este estudio se plantean varias preguntas sobre las que sería interesante aplicar técnicas de Educational Data Mining como:

- ¿quiénes son los estudiantes que toman más cursos universitarios?

- ¿quiénes son los que vuelven a tomar clases?
- ¿qué tipos de cursos se pueden ofrecer para aumentar la cantidad de estudiantes de una Universidad?
- ¿cuáles son las principales razones por las que un estudiante se cambia de curso o carrera?
- ¿es posible predecir el rendimiento académico de los estudiantes?
- ¿cuáles son los factores que afectan a este rendimiento?

Dicho estudio busca responder las dos últimas preguntas, que son también las más relevantes para este proyecto. Para ello, se realizó un análisis sobre un conjunto de datos de los liceos, de donde se seleccionó el rendimiento y la asistencia a clase, y además se recabó más información realizando cuestionarios a los estudiantes y sus padres. Se observa que los datos que se analizaron se pueden clasificar en datos demográficos (como el sexo, la edad y el lugar de residencia), datos académicos (como la asistencia a clase y el rendimiento histórico) y datos biográficos sobre el estudiante (como el consumo de alcohol y la frecuencia con la que sale con amigos).

En su análisis, Cortez concluye que es posible obtener una buena predicción si se tienen los datos del histórico del rendimiento académico de los estudiantes. En este conjunto de datos se cuenta con los resultados de dos pruebas intermedias y se muestra que estas variables son las más relevantes para la predicción académica. Para llegar a esta conclusión, a partir del conjunto de datos inicial se crean dos réplicas con la distinción de que en un caso falta uno de los resultados de las pruebas intermedias y en otro faltan los dos. Se aplican cinco clasificadores y en todos se muestra una notoria mejoría cuando el conjunto de datos contiene más datos del rendimiento histórico.[19]

Por otro lado, se realizó un análisis sobre el conjunto de datos del estudio de Elaf Abu Amrieh et al. de la Universidad de Jordania. Este conjunto de datos tiene la particularidad que dispone de atributos (features) relacionados con el comportamiento del estudiante (acceso a recursos, cantidad de intervenciones en clase, participación en foros de discusión); además de utilizar los datos demográficos (género, edad) y los datos académicos (nivel académico, rendimiento histórico). Estos datos fueron obtenidos utilizando el estándar xAPI, mencionado anteriormente, y extraído del uso del LMS llamado Kalboard 360.

En este estudio también se aplican técnicas de EDM, en particular utilizan las técnicas de clasificación Decision Tree (DT), Artificial Neural Network (ANN) y Naïve Bayes. Además se plantearon dos escenarios distintos para el análisis, uno que incluye las variables relacionadas con el comportamiento del estudiante y otro que las excluye.

En la Tabla 4.1 se muestran los resultados obtenidos por cada clasificador y por caso de estudio (con datos de comportamiento, c/DC, o sin ellos, s/DC).

Allí se observa claramente que la inclusión de los datos que tienen relación con el comportamiento del estudiante durante el curso mejora significativamente todas las métricas para los tres clasificadores.

	DT		ANN		NB	
	c/DC	s/DC	c/DC	s/DC	c/DC	s/DC
Accuracy	61.3	55.6	73.8	45.8	72.5	50.4
Recall	61.3	55.6	73.8	45.9	72.5	50.4
Precision	60.9	56.2	73.9	45.2	72.7	49.6
F1-Score	60.1	53.4	73.2	44.8	71.9	49.4

Cuadro 4.1: Se presentan los resultados de aplicar distintos clasificadores al conjunto de datos de [2] con datos de comportamiento [c/DC] y sin datos de comportamiento [s/DC]

En este estudio también se aplican técnicas de selección de atributos en las cuales se vuelve a confirmar la relevancia de los datos de comportamiento, siendo la cantidad de intervenciones en clase, la cantidad de recursos accedidos y la participación en los foros de discusión las variables más influyentes.

Por último, en el estudio de Ellis et al. se plantea que los cuestionarios en los que los estudiantes responden sobre su rendimiento académico aportan atributos relevantes y efectivos. Este estudio se realizó en una Universidad de Australia sobre 145 estudiantes de una carrera de Ingeniería. Allí se realizó un cuestionario con el objetivo de identificar el método de estudio de cada estudiante y de analizar si el mismo afecta su rendimiento académico. Se mencionan dos métodos, uno con un posible enfoque más superficial, donde el estudiante memoriza la información como hechos aislados, sin conexión con experiencias previas o con el contexto general con el objetivo central de retener datos para aprobar la evaluación; mientras que el otro es más profundo, orientado a realizar un análisis crítico de nuevas ideas o contenido, para luego integrarlo al conocimiento previo sobre el tema, favoreciendo con ello su comprensión y su retención en el largo plazo.[2]

Por último, en el estudio se concluye que los estudiantes que tienen un enfoque de estudio más profundo, se correlaciona con un mejor rendimiento académico.[22]

Como conclusión de lo estudios mencionados, por un lado de acuerdo al estudio de Amrieh et al. las variables relacionadas con el comportamiento del estudiante son las que tienen más relevancia a la hora de predecir el resultado académico. Por otro lado, en el estudio de Cortez se concluye que las variables más relevantes son el histórico del rendimiento académico (al menos dentro de un mismo curso) aunque también destaca otros factores relacionados con la actividad del estudiante (como el número de inasistencias y el apoyo extracurricular) y factores relacionados con el ámbito social (como el consumo de alcohol o la frecuencia con la que se reúne con sus amigos). El estudio de Ellis et al. sugiere que los estudiantes pueden proveer variables muy importantes al expresar cómo cree que está rindiendo académicamente, cuales son sus expectativas respecto al curso y cuál es su método de estudio (si utiliza aprendizaje superficial o profundo).

4.3. Selección del curso

Para el caso de estudio en que se basa este proyecto, se cuenta con la bases de datos del LMS (Moodle) y de Bedelías de la carrera de testing del Centro de Ensayo de Software (CES).

El primer paso fue unificar las mismas y seleccionar los datos más relevantes de acuerdo a los estudios previamente consultados. Se extrajeron datos demográficos (sexo, edad), datos académicos (resultados de las actividades intermedias de los cursos, nivel de estudio alcanzado previo a comenzar la carrera) y datos de comportamiento (cantidad de preguntas realizadas en los foros de discusión).

4.3.1. Introducción al Testing 1

Como primer instancia de análisis, se generó un conjunto de datos de 229 estudiantes de la asignatura Introducción al Testing 1 (IT1) de todos los años en que fue dictada. Para obtener los datos demográficos se utilizó la base de datos de Bedelías, mientras que para los datos de comportamiento y los académicos se utilizó además la de Moodle.

Analizando los datos obtenidos, ver Figura 4.1, se observó que el 98 % de los estudiantes aprobaron el curso. Tomando como modelo de referencia (baseline) un predictor que no tome en cuenta ningún dato y siempre predice que el estudiante va a aprobar, tendrá un 98 % de efectividad. Esto deja un margen muy pequeño para obtener un mejor resultado con un nuevo modelo por lo que se decidió cambiar el enfoque.

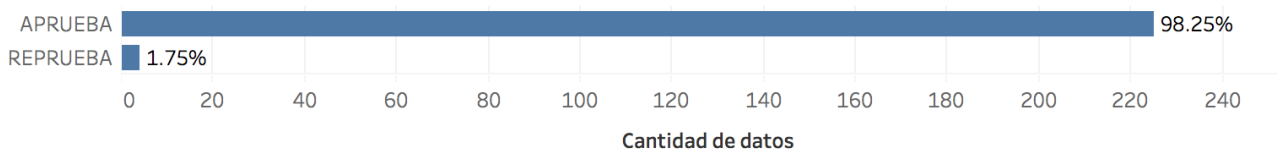


Figura 4.1: Relación entre la aprobación y reprobación de los estudiantes del curso IT1.

En lugar de intentar predecir la nota final del curso, se intentó predecir la nota de la primera instancia de la prueba final del mismo.

Este curso tiene varias entregas y una prueba final. Tanto las entregas como la prueba final pueden ser tomadas dos veces, y en algunos casos hasta una tercera vez. Esto aumenta significativamente la cantidad de estudiantes que aprueban el curso. Sin embargo, la cantidad de estudiantes que pierden la primera instancia de la prueba final es de un 85 %. Si bien éste conjunto está más balanceado que el otro, la diferencia sigue siendo considerable.

4.3.2. Introducción al Testing de Performance

Debido a esto, se optó por cambiar de curso. Por recomendación de los tutores, se procedió a analizar los datos del curso Introducción al Testing de Performance (ITP) tomando en cuenta instancias realizadas desde el año 2014 al 2017. Para la decisión se tomó en cuenta que el curso

cuenta con una gran cantidad de estudiantes registrados por período, que tiene una cantidad considerable de materias previas para utilizar como datos de entrada y además es uno de los cursos que más reprobados registra. Esto último es importante ya que por lo mencionado anteriormente acerca del curso de IT, cuanto más balanceado esté el conjunto de datos, es mayor la probabilidad de obtener un mejor modelo predictivo que el que se utiliza como baseline.

ITP es la última asignatura del Diploma Tester de Software, que es el primer diploma de los tres que otorga la carrera de testing. Este diploma se obtiene al completar el primer año de la carrera y se compone de las asignaturas Introducción al Testing, Introducción a la gestión de incidentes, Introducción al testing funcional, Documentación y reportes, Técnicas de testing funcional y por último Introducción al testing de performance.[3] Para los estudiantes que cursaron ITP, se extrajo de la base de datos las notas de todas las actividades de ITP, así como también de los cursos anteriores. Además, se extrajo la fecha nacimiento, la edad, el sexo, la nacionalidad, el nivel académico, la profesión y la empresa para la cual trabajan como datos personales.

Cabe destacar que no se utilizaron datos que puedan identificar a un usuario en particular, como la cédula de identidad, teléfono o la dirección de correo electrónico para mantener la privacidad de los estudiantes. Con los datos extraídos se generó un dataset de 185 estudiantes con 91 variables.

Al igual que en el caso del curso anterior, con el objetivo de que el conjunto de datos se encuentre lo más balanceado posible, se optó también por tomar la nota de la primera instancia de la prueba final como variable dependiente. En la Figura 4.2 se muestran los resultados.

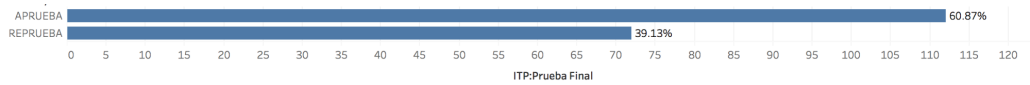


Figura 4.2: Muestra la relación de aprobación y reprobación de todos los estudiantes del curso ITP.

Se observa que el conjunto de datos está mucho más balanceado que el del otro curso, ya que el 61 % de los estudiantes aprobó la prueba final mientras que el 39 % la reprobó en su primera instancia. En un análisis más detallado, se observa en la Figura 4.3 la distribución de las notas a lo largo de la escala del 0 al 12, en la cual los estudiantes que obtienen una nota de 7 o más, aprueban el curso.

De los 184 estudiantes registrados, 112 aprobaron la primera instancia de la prueba final del curso, por lo cual tiene una tasa de aprobación de 61 %.

Debido a que es de interés predecir los estudiantes con riesgo de reprobación, aquellos estudiantes que aprobaron con la mínima nota pueden ser considerados también dentro de la población riesgosa. En este caso se observa que 86 (47 %) no tuvieron riesgo de perder el curso al obtener una nota mayor o igual a 8, mientras que los restantes 98 (53 %) estudiantes son considerados de riesgo, al perder el curso o salvar con la nota mínima aceptable.

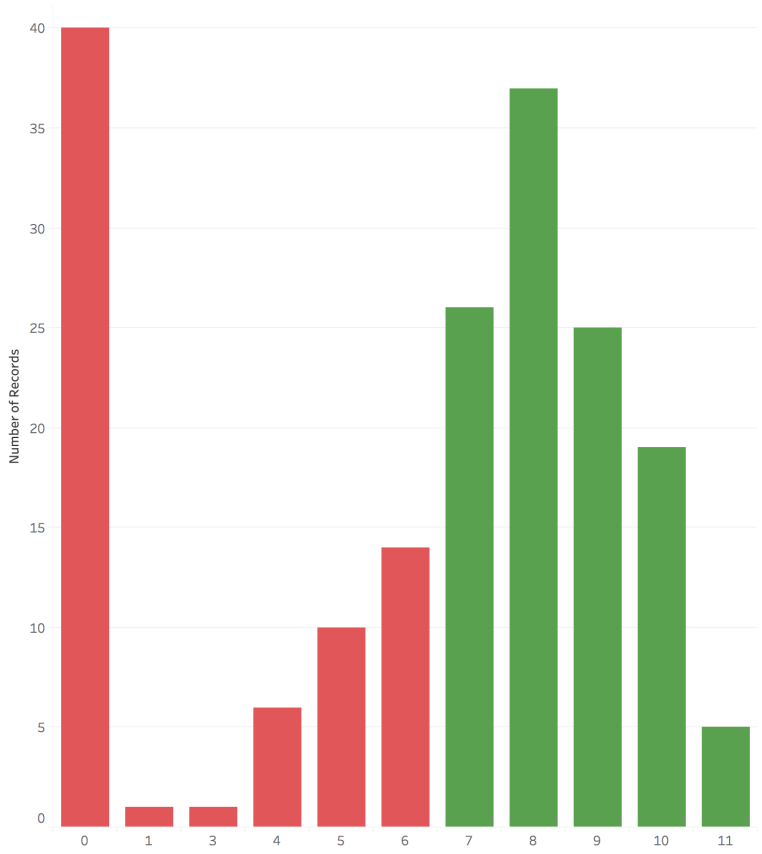


Figura 4.3: Distribución de notas de la prueba final de ITP.

4.4. Análisis exploratorio de datos

Uno de los pasos que se considera muy importante es la exploración de los datos dado que se pueden encontrar patrones, tendencias, relaciones o anomalías. En la etapa inicial mediante este análisis se generan hipótesis, lo cual sirve como base en la toma de decisiones en la etapa de preprocesamiento y en la etapa de validación del modelo. Para ésto se pueden utilizar tanto gráficas como tablas con información estadística que contribuyan a analizar los datos.

4.4.1. Análisis estadístico

En primer lugar se prestó atención a datos estadísticos de la muestra, para lo cual se crearon 2 tablas. Una de las tablas es la presente en la Figura 4.4, compuesta por la cantidad de datos no nulos y de valores distintos por atributo. Conocer la cantidad de nulos para un atributo es importante ya que cada nulo se sustituye por un valor estimado y si el número es muy grande se estaría tomando en cuenta un atributo con pocos valores reales y esto puede aumentar el error de la predicción. Por otro lado, si la cantidad de valores distintos es muy inferior a la cantidad posible según el dominio de cada atributo, seguramente no sea útil ya que puede agregar overfitting, dificultando la generalización del modelo al no tener ejemplos de la mayoría de los valores. Por estos motivos los atributos que cumplan con estas condiciones son candidatos a ser descartados. Cuando se analizaron los datos de ITP, se encontraron casos de

	count	nunique
Carrera Testing	124	2
Mes Nacimiento	124	12
Ano Nacimiento	124	33
Edad	124	32
Sexo	124	2
Ano Prueba	124	3
Mes Prueba	124	4
Dia de la semana Prueba	124	3
UY	124	2
PY	124	2
TS-M1-IT:Caracteristicas del testing	124	8
TS-M1-IT:Desarrollo de Software	124	6
TS-M1-IT:El software	124	4
TS-M1-IT:El testing	124	8
TS-M1-IT:Estrategias	124	6
TS-M1-IT:Tipos de prueba	124	5
TS-M1-IT:Prueba final	124	9
TS-M2-ITF:Testing exploratorio	124	7
TS-M2-ITF:Prueba final	124	6
TS-M2-ITF:Pensar casos de prueba	124	7

Figura 4.4: Tabla con contadores de datos y de valores únicos por fila

los mencionados, y los resultados mejoraron considerablemente por lo cual se implementó un método para automatizar el descarte de los mismos. Luego, en la Figura 4.5 se muestra otra

	Carrera Testing	Mes Nacimiento	Ano Nacimiento	Edad	Sexo	Ano Prueba	Mes Prueba	Dia de la semana Prueba	UY	PY	...	TS-M2-IGI:Definiendo un flujo de incidentes:n_entregas
count	124.000000	124.000000	124.000000	124.000000	124.000000	124.000000	124.000000	124.000000	124.000000	124.000000	...	124.000000
mean	0.701613	6.750000	1986.032258	29.064516	0.346774	2015.096774	10.467742	3.129032	0.983871	0.016129	...	0.951613
std	0.459406	3.565017	8.527274	8.465849	0.477874	0.726047	1.698567	0.754043	0.126483	0.126483	...	0.553699
min	0.000000	1.000000	1962.000000	17.000000	0.000000	2014.000000	5.000000	2.000000	0.000000	0.000000	...	0.000000
25%	0.000000	3.750000	1981.750000	22.000000	0.000000	2015.000000	10.000000	3.000000	1.000000	0.000000	...	1.000000
50%	1.000000	7.000000	1988.000000	28.000000	0.000000	2015.000000	11.000000	3.000000	1.000000	0.000000	...	1.000000
75%	1.000000	10.000000	1993.000000	33.000000	1.000000	2016.000000	11.000000	4.000000	1.000000	0.000000	...	1.000000
max	1.000000	12.000000	1999.000000	52.000000	1.000000	2016.000000	12.000000	4.000000	1.000000	1.000000	...	2.000000

Figura 4.5: Tabla con estadísticas de algunos atributos de ITP.

tabla con datos estadísticos de los atributos. Esta información brindada ayuda a comprender mejor las tendencias de la muestra y analizar si existe algún indicador con valores extraños con respecto al dominio de cada atributo. Por ejemplo para cada uno se muestra su desviación, lo cual si es muy elevada es un indicador de que pueden existir outliers provocados por errores en la muestra.

4.4.2. Análisis univariable

Una alternativa gráfica a la tabla anterior es la llamada boxplot o diagrama de caja como el de la Figura 4.6. Este tipo de diagramas facilitan la visualización de tendencias, siguiendo el

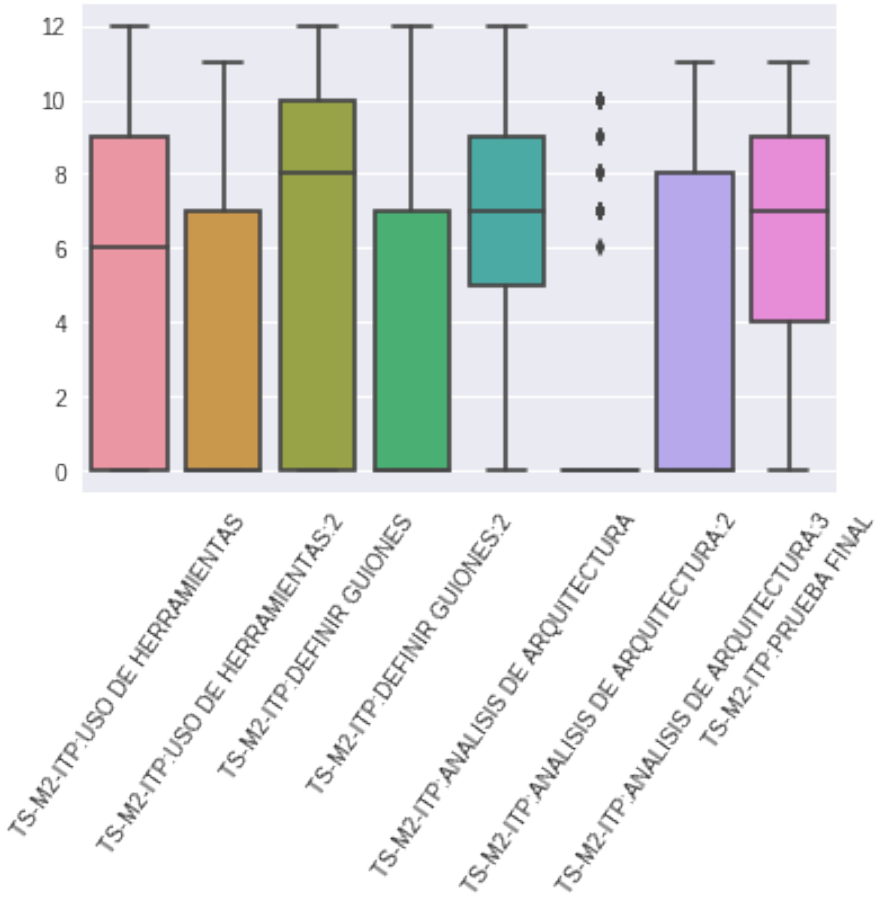


Figura 4.6: Gráfica de cajas con las entregas de actividades de ITP.

concepto anterior y por lo tanto también se pueden detectar outliers. En este caso se observaron las notas de las actividades de ITP. Sólo una actividad se destaca como fuera de lo normal, ya que tiene muy pocos datos. Esto se debe a que es una reentrega y puede pasar que la mayoría aprobó la actividad en la primer entrega y por lo tanto no necesitar reentregar. Lo mismo sucede en otros cursos, inclusive con una tercer reentrega. Esto sirvió como indicador de que en caso de tomar en cuenta esos atributos, era necesario sustituir una gran cantidad de valores nulos. Dado esto en la sección 6.2.1 se menciona que para solucionar este problema, se unificaron las entregas en una sola, y se generaron otros atributos nuevos indicando cantidad de entregas e insuficientes para poder distinguir los distintos casos existentes.

También se utilizaron gráficas tradicionales de distribución (ver Figura 4.7) y de histograma (ver Figura 4.8). Se eliminaron los estudiantes con nota 0, ya que son los que no se presentaron y no están incluídos en el caso de estudio ya que el objetivo es predecir el rendimiento de un estudiante en la prueba, por lo tanto al no presentarse no aporta información acerca del mismo. En caso de ser de interés predecir si se va a presentar o no, se debería generar un modelo aparte

para abordarlo.

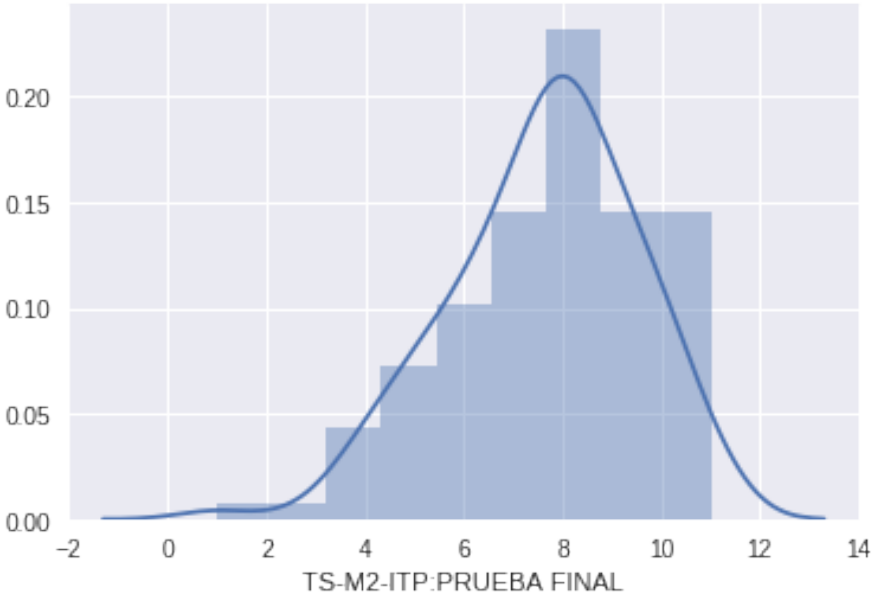


Figura 4.7: Gráfica de densidad de notas de ITP

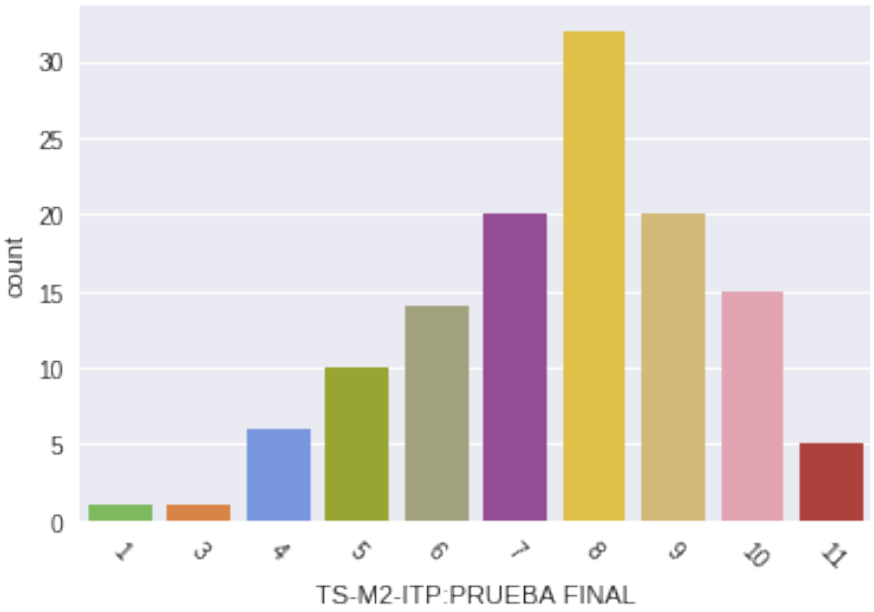


Figura 4.8: Histograma de las notas de alumnos que se presentaron (distinto a 0) a la prueba final de ITP por color.

La gráfica de distribución también incluye la de histograma para tener una mejor visión con la vista combinada. Uno de los aspectos clave a prestar atención es si la distribución de las notas se asemeja a una distribución normal, que en este caso se cumple. Esto se traduce a que se recomiendan utilizar algoritmos paramétricos y que es muy probable que la relación entre las variables independientes y la dependiente (nota de la prueba final) se pueda aproximar por

una regresión. Se destaca además, que la mayoría de los estudiantes obtuvieron una nota de 8 y que hay notas que no tienen registros.

4.4.3. Análisis bivariable - correlaciones

Otro aspecto importante es evaluar si existen correlaciones fuertes entre los atributos o con la variable dependiente como se pueden ver en la Figura 4.9 y la Figura 4.10.

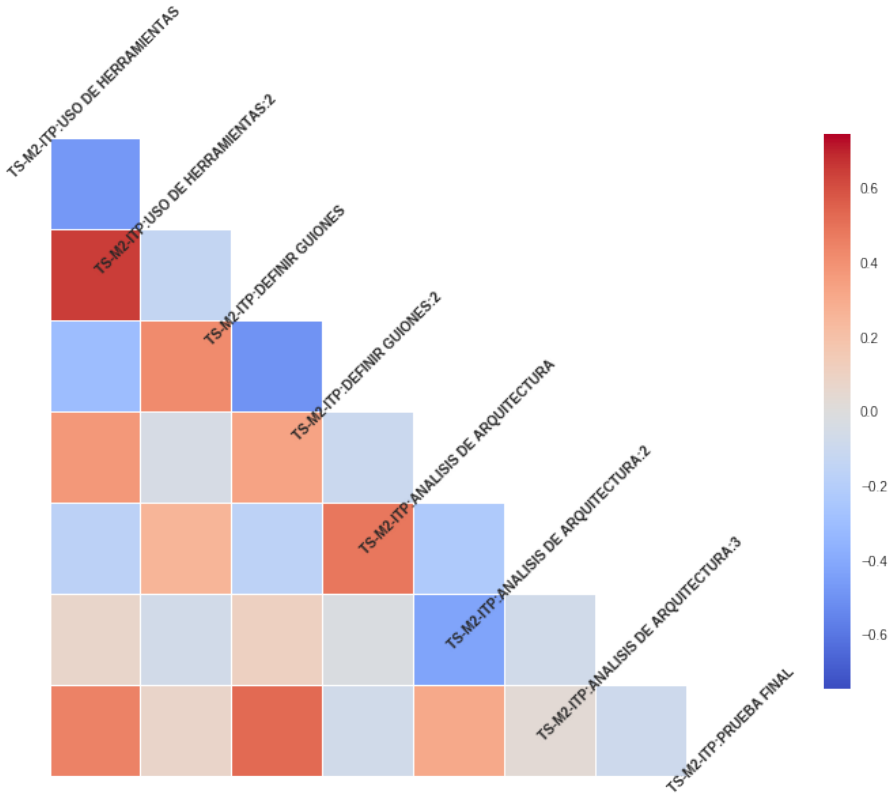


Figura 4.9: Correlación entre entregas de actividades de ITP y la nota final

En la primera se pueden apreciar mejor ya que sólo se visualizan las actividades de ITP, mientras que en la segunda se incluyen datos personales y de materias previas. Se pueden ver correlaciones fuertes, tanto positivas (color bordó), como negativas (color azul).

Para el caso de correlaciones entre atributos, en cada par hay uno de los atributos que es candidato a ser descartado ya que la variación de uno queda bien explicada por el otro. Para el caso de atributos correlacionados con la variable dependiente, son candidatos a que permanezcan ya que deberían explicar o incidir fuertemente en la variación de la misma.

4.4.4. Análisis Multivariable

Por último se procedió a analizar el conjunto entero de atributos. Dado que gráficamente, existe una limitación en cuanto a la cantidad de dimensiones, se utilizó la técnica de reducción de la dimensión con el algoritmo Principal Component Analysis (PCA) obteniendo gráficas de 2 y 3 dimensiones (ver Figura 4.11 y Figura 4.12 respectivamente). Se puede apreciar la ausencia

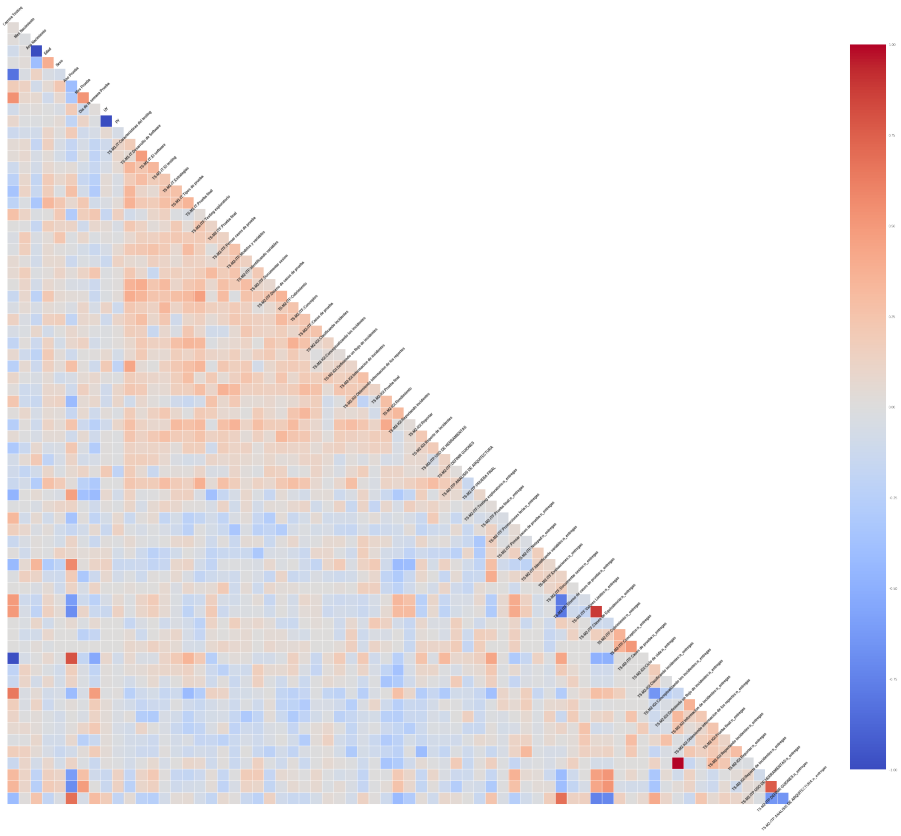


Figura 4.10: Correlación entre los atributos de ITP.

de agrupaciones de datos, lo cual es un indicador de que un modelo de clustering seguramente no sea el apropiado.

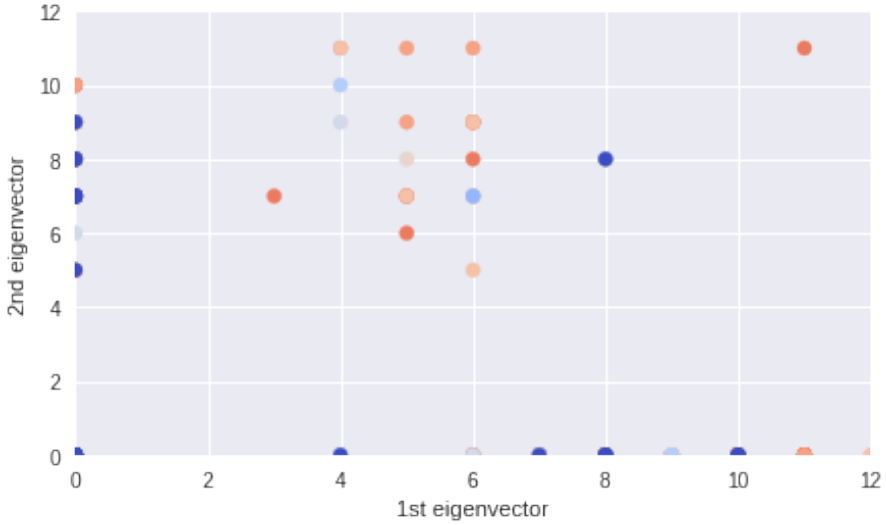


Figura 4.11: Análisis de PCA con los primeros 2 vectores.

Figura 4.12: Análisis de PCA con los primeros 3 vectores.

Capítulo 5

Preprocesamiento de datos

Luego de la extracción de datos, se procedió a realizar el preprocesamiento de los mismos. Este paso es es fundamental para que los datos se encuentren en un formato adecuado para los algoritmos de Machine Learning. Este proceso se separó en varias etapas claramente definidas, descritas a continuación.

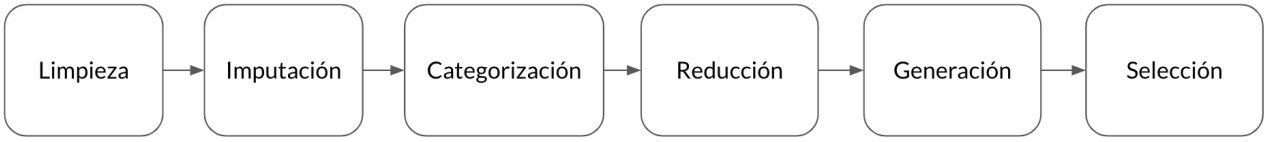


Figura 5.1: Pipeline de preprocesamiento de datos.

5.1. Limpieza

En primer lugar, se hizo una limpieza de los datos. De los 184 estudiantes que tomaron la prueba final de ITP, hay 33 estudiantes de los que no se tienen datos de los cursos ni actividades anteriores por lo que parece inviable poder predecir la nota final cuando prácticamente es el único dato disponible para poder realizar una predicción.

Debido a esto, los registros de estudiantes que le faltaban la mayoría de los datos fueron eliminados, resultando que el conjunto a analizar contara con 151 estudiantes. Esto ocurre porque hay estudiantes que se inscriben para tomar el curso ITP de forma individual sin participar de la carrera ni de otra asignatura ya que el CES brinda al estudiante la opción de realizar cursos independientes o un conjunto de los mismos.

Por otro lado, se detectaron alumnos con nota de prueba final 0 (cero) que indica que no se presentaron, lo cual se considera como inválido para los modelos que buscan predecir la nota final o su aprobación ya que no realizaron la prueba. Eliminando estos casos la cantidad de registros disminuyó a 125.

5.2. Imputación

Para algunos pocos de estos estudiantes no se cuenta con el dato de la fecha de nacimiento mientras que en otros casos la fecha era evidentemente errónea ya que el estudiante tendría 3 años al momento de tomar la prueba. Como estos valores no reflejan que el estudiante no tenga fecha de nacimiento sino que es un valor faltante de forma aleatoria podemos imputarlos (sustituir por otro valor).

Para solucionar el inconveniente, se investigaron y verificaron alternativas utilizando el promedio, la mediana o el valor más común como sustitutos. En este caso optamos por completar los datos con el promedio ya que es la única alternativa que mantiene el valor del promedio de todo el conjunto de datos y, como se verá más adelante, se utilizará el promedio para normalizar los datos al aplicar las técnicas de regresión.

5.3. Categorización

Para los valores que no son numéricos sino categoriales, como por ejemplo la nacionalidad, se analizó cómo transformarlos a numéricos para poder ser procesados por los algoritmos de predicción. Una alternativa es utilizar la técnica Label Encoder en la que se asigna un valor numérico para cada clase. En el presente caso de estudio, se cuenta con estudiantes de Uruguay, Chile, México y Paraguay. La misma asigna por ejemplo el valor 0 a Uruguay, 1 a Chile, 2 a México y 3 a Paraguay. Sin embargo, sugiere una relación de orden en algo que claramente no tiene ningún orden preestablecido. Debido a esto, se optó por utilizar la técnica de One Hot Encoder en la que se crean tantas variables nuevas como categorías se tienen, y se asigna un valor 0 o 1 indicando que el estudiante pertenece o no a esa categoría. Como resultado de su aplicación se obtuvieron variables llamadas esUruguayo, esChileno, esMexicano y esParaguayo en la que, tomando como referencia el ejemplo de la Tabla 5.1, un estudiante paraguayo tiene valor 0 en las tres primeras variables y un 1 en el campo esParaguayo. De esta forma se logra representar numéricamente datos categóricos que no tienen una relación de orden, como se puede ver en la Tabla 5.3

Estudiante	Nacionalidad
A	Uruguayo
B	Paraguayo

Cuadro 5.1: Ejemplo con nacionalidades de dos estudiantes

5.4. Reducción y generación de datos

En muchos casos, un mismo estudiante tiene la posibilidad de tomar una prueba en más de una oportunidad si es que no obtuvo un buen resultado (ver Tabla 5.3).

Estudiante	esUruguayo	esChileno	esMexicano	esParaguayo
A	1	0	0	0
B	0	0	0	1

Cuadro 5.2: Transformación de los datos con One Hot Encoder

Para contemplar esta situación, se creó una variable nueva para cada una de las actividades que refleja cuántas veces el estudiante realizó la actividad. También se generó otra que registra la cantidad de insuficientes y una más de valor binario indicando si la actividad fue aprobada o no (ver Tabla 5.4).

La cantidad de insuficientes se crea debido a que hay casos en que los estudiantes entregan luego de la fecha y hora estipulada (con solo algunos minutos de atraso) y eso queda almacenado como nota 0 en la primer entrega y con la respectiva nota en la siguiente entrega. Lo mismo puede ocurrir entre la segunda y tercer entrega. Teniendo solo la variable de cantidad de entregas, estos casos serían iguales a otros en los cuales sí realizó una entrega, que fue evaluada y luego debió re entregar. Pero para el último ejemplo se debe registrar una entrega insuficiente, pudiendo así discriminarlos con los casos anteriores de nota 0.

Por otro lado, la columna de aprobación fue generada para indicarle al modelo si la actividad fue aprobada o no, dado que la nota por sí sola carece de este dato y es importante que sea tenido en cuenta.

Estudiante	Prueba	Prueba (2^{o} instancia)	Prueba (3^{o} instancia)
A	12	-	-
B	4	8	-
C	0	3	9
D	2	5	6

Cuadro 5.3: Transformación de los datos con One Hot Encoder

Estudiante	Prueba	Prueba_n_entregas	Prueba_n_insuficientes	Prueba_aprobado
A	12	1	0	1
B	8	2	1	1
c	9	3	1	1
D	6	3	3	0

Cuadro 5.4: Ejemplo de agregar las variables cantidad de entregas, insuficientes y de aprobación

Además, se realizó una normalización de los datos correspondientes a las variables independientes, restando el promedio a cada una, dado que existen estimadores que presentan errores si los mismos no se encuentran normalizados ya que ponderan a una variable por sobre otra por el mero hecho de tener mayor valor. Por ejemplo, el año de nacimiento puede tomar una importancia mucho mayor que una nota, ya que los valores se encuentran en una escala en el

orden de los miles y no en unidades como los de la nota. Esta transformación se puede llevar a cabo ya que lo interesante para analizar de los datos es la desviación de cada variable y no el valor exacto. Un ejemplo de la misma ocurre al aplicarla en una distribución normal obteniendo otra de media μ en el valor 0 y desviación estándar σ a 1. La normalización de los datos se realizó para todas las variables excepto para la variable dependiente (la cual se busca predecir).

En lo que respecta a la variable dependiente, se tomaron dos enfoques. Por un lado, se intentó predecir la aprobación o no de un curso o una prueba. Para ello se modificó la nota de aprobación en una variable binaria, donde 0 indica que el estudiante reprobó el curso y 1 que lo aprobó. Este enfoque permite aplicar técnicas de clasificación sobre los estudiantes.

Por otro lado, se intentó predecir la nota exacta de los estudiantes, lo cual permite utilizar técnicas de regresión, y para este caso no se necesitó modificar en absoluto las variables.

5.5. Selección de atributos

Para la selección de atributos se utilizaron las clases SelectKBest, RFE (Recursive Feature Elimination), RFECV del módulo feature_selection y PCA del módulo decomposition. Para los distintos análisis se probaron todas las técnicas y se obtuvieron mejores resultados con SelectKBest en todos los casos.

SelectKBest se basa en un algoritmo que mide la relación entre la variable dependiente y cada una de las independientes, eligiendo un conjunto de tamaño 'k' con los atributos de mayor relación con la variable que se busca predecir.

Por otro lado RFE utiliza un algoritmo que le asigna un valor a los atributos que cuantifica su importancia con respecto a la variable dependiente y realiza una recursión que en cada paso elimina un conjunto 'n' de atributos con menor valor (importancia).

RFECV se basa en una validación cruzada y el algoritmo RFE para elegir el conjunto óptimo de atributos.

PCA en cambio realiza una reducción de dimensión lineal utilizando el algoritmo de SVD (descomposición en valores singulares).

5.6. Validación

Uno de los problemas que surgieron fue evaluar si la predicción realizada era aceptable o no. Para poder determinar si las predicciones son útiles y confiables hay que determinar cómo medir los resultados de la predicción. Además, es una buena práctica establecer cuales son los resultados mínimos aceptables para saber cuando se ha alcanzado un resultado válido o cuando hay que seguir iterando la solución. A continuación se documentan las formas de validación que fueron utilizadas para la clasificación y la regresión.

5.6.1. Validación de la clasificación

Para la clasificación binaria, en la que se busca predecir si un estudiante pertenece al conjunto de estudiantes de riesgo o no, se tomó como base las métricas de un clasificador que para todo estudiante predice el resultado más probable, sin considerar otros datos del estudiante (sólo la nota final del conjunto de test). Esto nos da un valor mínimo que el algoritmo predictivo debe mejorar para agregar valor al usuario final.

Para poder analizar mejor los resultados, se definen las siguientes categorías de los resultados de acuerdo a si el clasificador predijo correctamente o no el resultado.[20]

- True Positives (TP): cantidad de estudiantes que aprobaron el curso que el clasificador predijo que iban a aprobar.
- False Positives (FP): cantidad de estudiantes que reprobaron el curso que el clasificador predijo que iban a aprobar.
- True Negatives (TN): cantidad de estudiantes que reprobaron el curso que el clasificador predijo que iban a reprobar.
- False Negatives (FN): cantidad de estudiantes que aprobaron el curso que el clasificador predijo que iban a reprobar.

En primer lugar, para el caso de la clasificación, se calculó el número de estudiantes que aprueban el curso de Introducción al Testing de Performance. De los 184 estudiantes que realizaron este curso se observa que 112 lo aprobaron, representando a un 60 % del total. Si se predice que absolutamente todos los estudiantes aprueban el curso, para el 60 % de los casos este predictor dará un resultado correcto, mientras que para todos los estudiantes que reprobaron dará un resultado incorrecto. Si se define la exactitud de un algoritmo como la cantidad de ejemplos que clasifica correctamente, este predictor tendrá una exactitud de 60 %, ya que clasifica correctamente 112 estudiantes de los 184 que cursaron la materia.

En la Tabla 5.5 se muestra cómo se distribuyen los resultados del clasificador en las clases previamente definidas.

Con estos valores se puede calcular la exactitud del algoritmo con la siguiente fórmula.

$$Exactitud = \frac{TP + TN}{TP + FP + TN + FN} = \frac{112 + 0}{112 + 72 + 0 + 0} = 0{,}6087$$

		Estudiante	
		Aprueba	Reprueba
Clasificador	Aprueba	TP = 112	FP = 72
	Reprueba	FN = 0	TN = 0

Cuadro 5.5: Matriz de confusión del clasificador

Sin embargo, a la hora de determinar la performance de un modelo, generalmente la exactitud no es la mejor métrica. Un ejemplo claro para ilustrar este punto, es el caso de la predicción del tratamiento de enfermedades. En el caso de ejemplo, se desea predecir a quienes se debe administrar una vacuna que puede salvarle la vida a un individuo enfermo y no tiene efectos adversos en un individuo sano. En este caso, es mucho más problemático tener un falso negativo que un falso positivo, ya que un falso negativo implica que un individuo enfermo no recibirá la vacuna, mientras que un falso positivo solo implica que se utilizará una vacuna de más. Si solo se toma en cuenta la cantidad de casos que se aciertan sin prestar atención en los que no, se pueden estar obteniendo muchos falsos negativos, provocando que una gran cantidad de pacientes no reciban medicación.

Otras métricas comúnmente utilizadas son la precisión y el recall.

Para el caso de estudio, la precisión brinda información acerca de la proporción de estudiantes correctamente clasificados como aprobado, respecto a los clasificados como aprobados. Para el predictor base podemos calcularlo con la siguiente fórmula:

$$Precision = \frac{TP}{TP + FP} = \frac{112}{112 + 72} = 0{,}6087$$

Debido a que este clasificador siempre clasifica todos los elementos como aprobados, la precisión es igual a la exactitud, aunque como se observa en las fórmulas, esto ocurre solo para este caso en particular.

Por otro lado, el recall busca responder la pregunta de dado todos los estudiantes que aprobaron, cuántos de ellos el clasificador predijo que iban a aprobar. Se puede calcular matemáticamente con la siguiente fórmula:

$$Recall = \frac{TP}{TP + FN} = \frac{112}{112 + 0} = 1$$

Como se observa, el clasificador que se tomó como base tiene una precisión de 0.61 y un recall de 1.0. Idealmente, un clasificador perfecto tendrá tanto para la precisión como para recall un valor de 1.0. El desafío que se plantea es de obtener un balance entre precisión y recall, haciendo que ambos sean lo más alto posible, o sea, lo más cercano a 1.

Se plantea la interrogante de cuál variable es más importante, y su magnitud, en el balance mencionado. Se observa que no hay una única respuesta a esta pregunta pero lo que puede ayudar a tomar una decisión es evaluar para qué se quiere hacer la predicción. En este caso, si la predicción se realiza para identificar prematuramente a estudiantes con riesgo de perder el curso, se puede argumentar que la variable más importante es el recall, ya que al aumentar se asegura

que si un estudiante está en riesgo, el algoritmo lo detectará y se podrán tomar las acciones necesarias para mejorar su rendimiento. Sin embargo, si se toma la decisión de solo aumentar el recall sin considerar la precisión, nos enfrentamos al caso en que todos los estudiantes sean catalogados como riesgosos y se intente tomar acciones para mejorar el rendimiento de todos.

En el escenario ideal esto no sería un problema, pero también se busca utilizar los recursos de manera eficiente. Esto implica brindar un apoyo especial a todos los estudiantes que lo requieran y minimizar la cantidad de recursos destinados a estudiantes que no tienen un riesgo real de reprobar el curso.

Para lograr este balance, se introduce una nueva métrica que agrupa la precisión y el recall en una única variable, llamada F-score o F1 que representa la media armónica entre las dos variables. A continuación se muestra la fórmula matemática para calcular F1 con los valores del clasificador base.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} = \frac{2 \times 0{,}6087 \times 1}{0{,}6087 + 1} = 0{,}7578$$

Los algoritmos de clasificación utilizados fueron evaluados utilizando ésta métrica.

5.6.2. Validación de la regresión

En el caso de la regresión se plantea la misma problemática de encontrar una forma de comparar las distintas predicciones para poder elegir la mejor, de acuerdo a algún criterio. Hay varias métricas que sirven para evaluar una predicción.

En la Subsección 5.6.2 se define una de las métricas más sencillas, Mean Absolute Error (MAE), en la cual se suma el error absoluto de cada predicción con respecto al valor real y lo divide entre la cantidad de predicciones, para encontrar el error promedio.

$$MAE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} |y_i - \hat{y}_i|$$

En el presente caso de estudio, el error es la diferencia entre la nota que realmente obtuvo el estudiante con la nota predicha por el algoritmo. Esta métrica va desde 0 hasta infinito, siendo su valor óptimo el 0, el cual indica que la predicción es exactamente igual que el valor real.

En la Subsección 5.6.2 se define otra de las métricas comúnmente utilizadas es el Mean Squared Error (MSE) la cuál se calcula a partir del promedio del cuadrado de los errores.

$$MSE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (y_i - \hat{y}_i)^2$$

Al igual que el MAE la escala va del 0 al infinito y su mejor valor es el 0. Tiene como particularidad que penaliza los errores, incrementándose proporcionalmente al valor del error. Para los objetivos de este proyecto esta es una cualidad de interés, ya que cuanto más grande es el error, mayor es la probabilidad de que el estudiante sea catalogado en el grupo de atención incorrecto. Por ejemplo, si se dispone de un algoritmo con un error pequeño que predice una nota de 3 cuando un estudiante obtiene una nota de 2, el estudiante será correctamente catalogado en el grupo de los estudiantes que requieren atención. Sin embargo, un mayor error puede hacer que un estudiante no reciba la atención requerida teniendo potencialmente un impacto significativo en la calidad de la enseñanza. Si bien penalizar el error es una característica deseable, esta métrica tiene la desventaja de que si existe al menos un valor de error muy elevado, puede tener un gran impacto en la métrica haciendo que el algoritmo sea descartado, cuando quizás el error ocurre en un caso particular. Las métricas presentadas previamente tienen la particularidad que su valor está en la misma escala que las variables que se están prediciendo. Esto hace que sea difícil evaluar un algoritmo en base al valor de MSE o MAE, por lo que es conveniente utilizarlos como medidas de comparación entre los mismos. Para resolver esta problemática, hay una métrica que se conoce como R2-score o coeficiente de determinación, definida en la Subsección 5.6.2, que compara el valor de MSE de un algoritmo con el MSE de otro, con la particularidad de que este último siempre predice el valor promedio.

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{samples}-1}(y_i - \hat{y}_i)^2}{\sum_{i=0}^{n_{samples}-1}(y_i - \bar{y})^2}$$

De esta forma el valor siempre está entre menos infinito y 1. Si el valor es 1 significa que el MSE del algoritmo es 0, por lo que no hay ningún error. Un valor entre 0 y 1 significa que el MSE del algoritmo es menor al MSE del valor promedio, por lo que el algoritmo está prediciendo mejor que un modelo que solo toma el promedio. Si el valor es 0, significa que el algoritmo tiene exactamente el mismo error a tomar siempre el valor promedio, mientras que si el valor es negativo la predicción del algoritmo es peor.

5.6.3. Validación cruzada

Otra de las interrogantes que se plantean a la hora de comenzar a hacer un análisis predictivo es en relación a cómo obtener un resultado sin tener que esperar al próximo curso y ver si la predicción fue correcta o no. Lo que normalmente se realiza es dividir el conjunto de datos en 2 grupos, uno para entrenar al algoritmo y otro de verificación o test. Esto se realiza para probar el modelo generado a partir del conjunto de entrenamiento, utilizando otro distinto que aún no haya procesado. En este caso, se tomó el 80 % del conjunto de datos para entrenar el algoritmo y se validó con el 20 % restante. Esta decisión fue tomada debido a que son pocos datos y es una práctica común hacer esta separación.

Cabe destacar que los datos para entrenar el algoritmo se obtienen de forma aleatoria, para que el algoritmo pueda entrenar sobre todos los datos sin importar si estos estaban ordenados de alguna manera que pueda introducir alguna desviación en la predicción.

Además de separar los datos en conjuntos de entrenamiento y prueba, se realizó lo que se conoce como validación cruzada de tamaño N, en este caso se tomó N=10, qué es el número que se utiliza generalmente.

Validación cruzada es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba.

La validación cruzada de N iteraciones consiste en dividir los datos en N conjuntos y realizar N iteraciones. En cada iteración se utiliza un conjunto de test distinto y los N-1 conjuntos restantes como entrenamiento. Esta técnica evita que se tome un conjunto de entrenamiento o de validación particular que produzca un sesgo en la predicción y dificulte la generalización del modelo. El resultado final se obtiene del promedio de los resultados intermedios obtenidos en cada iteración.

Las técnicas de validación mencionadas, son útiles para evitar el fenómeno que se conoce como overfitting. Ocurre overfitting cuando el modelo es muy bueno para predecir sobre el conjunto de entrenamiento, con los datos que ya pudo procesar, pero no puede generalizar al conjunto de prueba, o sea, a datos nuevos. Si no se separa el conjunto de datos en entrenamiento y prueba, el modelo puede predecir muy bien utilizando datos conocidos pero quizás no sea capaz de generalizar con datos nuevos.

Capítulo 6

Análisis de datos

Una vez que se finalizó con el preprocesamiento de los datos, se comenzó con el análisis de los mismos. El objetivo de este paso es generar un modelo predictivo que logre obtener el mejor resultado posible de acuerdo a las métricas definidas anteriormente. Para ello se realizaron dos enfoques diferentes. En primer lugar, se utilizaron técnicas de clasificación cuyo resultado es discreto y busca predecir si un estudiante aprobará o no el curso. En segundo lugar, se hicieron pruebas con técnicas de regresión cuyo resultado intenta predecir la nota exacta que obtendrá un estudiante.

6.1. Clasificación

Hay una gran variedad de algoritmos de clasificación de los que se puede elegir. Para este caso, se utilizaron cinco clasificadores de cinco familias de estimadores distintas. Random Forest (RF), Gaussian Naive Bayes (GNB), Decision Tree (DT), Passive Agressive Classifier (PAC) y Support Vector Machine (SVM).

En principio se realizaron las predicciones sobre tres bases de datos con las mismas variables pero en una de ellas se tiene el total de alumnos que cursaron la asignatura, mientras que en la segunda se tiene solo los que se presentaron a la prueba final y en la tercera se modifica la variable dependiente para que sea 1 en el caso de que el estudiante se haya presentado al la prueba final y 0 en caso contrario. El primer conjunto de datos sería el ideal para realizar la predicción ya que cuenta con todos los estudiantes y responde a la pregunta de si el estudiante aprobará o no el curso. El segundo conjunto intenta responder la interrogante de si el estudiante aprobará o no dado que se presenta a la prueba final. Por otro lado, el tercer conjunto de datos busca responder si el estudiante se presentará o no a la prueba final.

En la Tabla 6.1 se pueden ver los resultados obtenidos utilizando la base de datos con todos los estudiantes. En la misma se aprecia que un predictor ciego que se utilizó como baseline tiene una precisión de 0.53, un recall de 1.00 y un F1-Score de 0.69. Luego de realizar las predicciones con los cinco clasificadores, se observa que Random Forest fue el que obtuvo una mejora significativa tanto en precisión como en F1-Score. SVM y Passive Aggressive tuvieron también una mejora sobre el predictor ciego, mientras que Naïve Bayes y Decision Tree (DT)

realizaron una predicción peor que el mismo.

Métrica	Baseline	RF	NB	DT	PAC	SVM
Accuracy	0.53	0.68 (+0.15)	0.59 (+0.06)	0.59 (+0.06)	0.65 (+0.12)	0.66 (+0.13)
Recall	1.00	0.89 (-0.11)	0.67 (-0.33)	0.59 (-0.41)	0.69 (-0.31)	0.85 (-0.15)
F1-Score	0.69	0.79 (+0.10)	0.62 (-0.06)	0.68 (-0.01)	0.74 (+0.05)	0.75 (+0.07)

Cuadro 6.1: Resultados de clasificación sobre el conjunto de datos con todos los estudiantes.

Luego, en la Tabla 6.2 se puede analizar que para el caso del conjunto de datos que solo tiene los estudiantes que se presentaron a la prueba final, la predicción de los algoritmos fue bastante mejor.

El mejor resultado es también del Random Forest con un 0.87 de F1-Score, representando una mejora de 0.22 sobre el predictor ciego.

Llama significativamente la atención el mal resultado de Naïve Bayes, con un F1-Score de 0.54 puntos por debajo del predictor ciego.

Métrica	Baseline	RF	NB	DT	PAC	SVM
Accuracy	0.60	0.78 (+0.18)	0.28 (-0.32)	0.69 (+0.09)	0.72 (+0.12)	0.74 (+0.14)
Recall	1.00	0.93 (-0.07)	0.07 (-0.93)	0.83 (-0.17)	0769 (-0.24)	0.97 (-0.03)
F1-Score	0.65	0.87 (+0.22)	0.12 (-0.54)	0.80 (+0.15)	0.79 (+0.14)	0.85 (+0.20)

Cuadro 6.2: Resultados de clasificación sobre el conjunto de datos sin los estudiantes que no se presentaron.

Por último para el conjunto de datos que busca predecir si un estudiante se presentará o no a la prueba, los resultados no fueron tan buenos (ver Tabla 6.3), ya que mantienen el nivel del clasificador ciego o empeoran los resultados.

Métrica	Baseline	RF	NB	DT	PAC	SVM
Accuracy	0.84	0.84 (0.00)	0.26 (-0.58)	0.73 (-0.11)	0.76 (-0.08)	0.83 (-0.01)
Recall	1.00	0.98 (-0.02)	0.15 (-0.85)	0.80 (-0.20)	0.83 (-0.17)	1.00 (0.00)
F1-Score	0.87	0.91 (+0.04)	0.20 (-0.67)	0.82 (-0.05)	0.84 (-0.023)	0.91 (+0.04)

Cuadro 6.3: Resultados de clasificación sobre el conjunto de datos para predecir si el estudiante se presentará o no a la prueba final.

Se observa que para algunos algoritmos se obtiene un mejor resultado que el resultado de un clasificador ciego, el cual se tomó como base. Por otro lado, otros algoritmos obtienen un peor resultado que el clasificador ciego. Incluso, en el caso de los algoritmos en los cuales se obtiene una mejora, la misma no es significativa.

Si se realiza una comparación con los resultados de los artículos consultados en la literatura, los resultados también son de menor calidad.

6.2. Regresión

Otro problema planteado fue la predicción de la nota final de cada estudiante. Dado que en este caso la nota final de los cursos es un valor numérico entre 1 y 12, se utilizan algoritmos de regresión. Para esto se utilizaron las siguientes implementaciones de algoritmos proporcionadas por la librería ScikitLearn:

Ridge, RidgeCV, Lasso, LassoCV, RandomForestsRegressor, MLPRegressor, ElasticNetCV.

6.2.1. Curso Introducción al Testing de Performance

El primer paso fue cargar los datos en un ipython notebook para ver qué resultados se obtenían sin alterar o procesar los mismos. Con el primer modelo generado se obtuvo un valor de R2 Score negativo, que significa que la regresión está obteniendo un resultado peor que devolver la nota promedio. Debido a esto se analizaron posibles soluciones a los distintos problemas encontrados en el conjunto de datos, detallados a continuación.

El primer problema encontrado fue que las implementaciones mencionadas no soportan valores nulos, por lo cual se realizaron distintas tareas de preprocesamiento, descritas anteriormente, para obtener un conjunto de datos adecuado. Una de ellas consistió en sustituir los valores nulos por el promedio de los demás valores. El problema de esta técnica es que añade error al modelo al tener que agregar estimaciones y una de las soluciones fue unificar las entregas dado que las re entregas tienen una gran cantidad de nulos a sustituir.

Además se detectó que habían alumnos con la mayoría de valores en nulo que afectaban negativamente al modelo y se quitaron del conjunto de datos. Esto es algo que ocurre para este curso puntual, ya que hay estudiantes que se anotan a este curso sin haber cursado el resto de la carrera, provocando que no se tengan notas en las actividades previas.

Para estas tareas de preprocesamiento se implementó un módulo Pipeline (descrito en el capítulo Solución propuesta) que dispone de las automatizaciones correspondientes a cada una. Cada clase del módulo implementa los métodos fit y transform requeridos para poder ser utilizados como paso de un Pipeline de Scikit Learn.

Otro obstáculo encontrado fue que existía una cantidad considerable de atributos redundantes y se procedió a realizar una selección. Primero se utilizaron implementaciones de Scikit Learn de técnicas de selección de atributos y luego se fue optimizando el conjunto a medida que se fue adquiriendo mayor conocimiento del dominio del problema. Para la selección se realizaron distintas pruebas utilizando las clases SelectKBest, RFE, RFECV del módulo feature_selection y PCA del módulo descomposition; y se obtuvieron mejores resultados con SelectKBest.

Además de esto se utilizó la librería pandas para obtener información descriptiva y la librería seaborn para obtener gráficas de correlación, distribución, histogramas y de caja (análisis de rango intercuartil). Esto permitió una mejor comprensión del dominio y realizar una selección manual de atributos optimizando el conjunto obtenido por el proceso automático.

El siguiente paso consistió en utilizar distintos algoritmos (mencionados al comienzo de ésta sección), optimizarlos y realizar una comparación para evaluar cuál era el que obtenía

mejores resultados. Para esto se utilizó como base una clase llamada EstimatorSelectionHelper implementada por Panagiotis Katsaroumpas para realizar una optimización de estimadores utilizando el módulo GridSearchCV y luego una comparación y selección de los mismos, la cual fue extendida para soportar la selección de atributos utilizando las implementaciones mencionadas al inicio de la sección y los módulos Pipeline y FeatureUnion para concatenar uno o más algoritmos de selección con un modelo predictivo.

Además se agregaron las siguientes funcionalidades:

- Se implementó un método de transformación genérica para poder utilizar uno o más algoritmos de transformación por ejemplo: Normalizer, StandardScaler, RobustScaler, PolynomialFeatures. También se implementó un método independiente para cada uno de éstos algoritmos.

- Validación cruzada para cada uno de los estimadores de entrada, disponible según las distintas versiones implementadas en ScikitLearn: cross_val, cross_val_predict, kfold, cross_val_repeated.

- Obtención de resultados utilizando métricas de regresión en formato de tabla para poder realizar la comparación y ver los resultados

- Obtención de los mejores parámetros y atributos para cada estimador

Utilizando esta clase se realizaron los últimos pasos de optimización, selección y comparación de estimadores obteniendo los resultados detallados a continuación.

Cuadro 6.4: Resultados de la regresión

	r2	mse	rmse	mae	evs
MLPRegressor	0.662071	1.306297	1.142934	0.956772	0.681052
Ridge	0.521713	1.848868	1.359731	1.112290	0.538169
Random Forest	0.447905	2.134180	1.460883	1.214000	0.490546
LassoCV	0.033901	3.734553	1.932499	1.613394	0.067272
ElasticNetCV	0.033493	3.736129	1.932907	1.613438	0.066839

El mejor score (R2: 0.66) se obtiene cuando se optimizan los parámetros de los estimadores y los datos no están estandarizados ni normalizados. En la Tabla 6.4 se puede ver además que el algoritmo con el cual se obtiene el mejor score es 'MLPRegressor'. Por otro lado el resultado de aplicar validación cruzada es mucho menor, R2 0.33 (ver Tabla 6.5).

Cuadro 6.5: Resultados utilizando Validación cruzada

	r2	mse	rmse	mae	evs
MLPRegressor	0.319441	2.406455	1.551275	1.206353	0.320714
Random Forest	0.315439	2.420606	1.555830	1.236160	0.315444
Ridge	0.297401	2.484390	1.576195	1.233168	0.297531
ElasticNetCV	0.296960	2.485950	1.576690	1.236537	0.296998
LassoCV	0.288210	2.516889	1.586471	1.234724	0.288263

La diferencia de resultados se debe a que la validación cruzada varía los conjuntos de verificación (test), mientras que en la optimización de parámetros se realiza una sola división del conjunto, el 80 % para entrenar y el 20 % para verificación.

Luego para intentar mejorar los resultados se decidió agregar una columna más que indique si el alumno aprobó o no el curso obteniendo los datos del resultado del clasificador implementado.

También se utilizó la librería imbalanced-learn[52] para mejorar el balance de los datos. Dado que sólo soporta estimadores de clasificación y no de regresión se utilizó solo en el clasificador del último experimento que se utiliza para predecir la aprobación.

Si bien se obtuvo una mejoría con los dos experimentos, no fue significativa y por lo tanto no fueron tenidos en cuenta para el modelo final.

Capítulo 7

Plataforma Web

Para la generación de los modelos predictivos se utilizó la librería Scikit-Learn y el entorno IPython Notebook. Ambas herramientas son muy completas y útiles para realizar análisis de datos y generar modelos. Sin embargo existen una gran cantidad de tareas en cada etapa de la generación de un modelo y posterior uso para predecir, que se deben repetir para cada conjunto nuevo de datos que se desee predecir o entrenar. Por este motivo surgió la oportunidad de generalizar y automatizar las tareas para ahorrar tiempo de análisis e implementación y además generar un módulo de procesamiento automático y parametrizado. Para lograrlo, se desarrollaron librerías de automatización y generalización de las etapas mencionadas en el Capítulo 3 - Desarrollo.

Luego se implementó una aplicación web con el fin de facilitar el uso de las mismas y prescindir de una consola. Esta permite al usuario cargar datos, procesarlos, visualizar los resultados en tablas/gráficas y descargarlos. También se dispone de un resumen del análisis que incluye los parámetros elegidos.

En la Figura 7.1, se pueden ver los componentes.

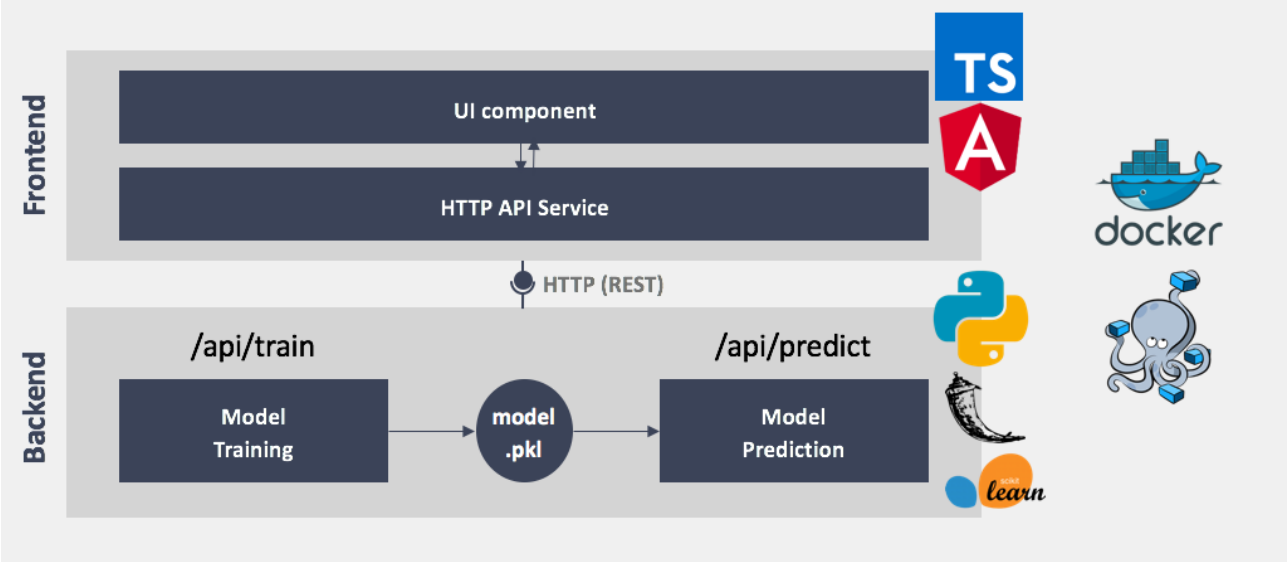


Figura 7.1: Componentes de la Aplicación Web

El sistema está compuesto por una aplicación de backend desarrollada con el framework Flask para python que consume las librerías de automatización y otra de frontend desarrollada con el framework Angular versión 8 para javascript. Para facilitar la instalación del ambiente y obtener portabilidad, se decidió automatizar el despliegue del sistema utilizando las herramientas docker y docker-compose. Como aclaración, es necesario instalar previamente docker y docker-compose para poder ejecutarlo ya que no son provistas por el sistema operativo. Esto permite definir todas las dependencias que se requieren instalar en cada archivo de imagen de docker (archivo de texto plano). Además sólo se necesita ejecutar el siguiente comando para que se comience a instalar el ambiente y queden disponibles los servicios: # docker-compose up

Para alcanzar esto se implementaron 3 imágenes de docker:

- Frontend - Web app
- Backend - Rest API service
- Redis Queue - Servicio de cola de trabajos

7.1. Librerías de automatización

7.1.1. Librería para Data Cleaning y Feature Engineering

Para el preprocesamiento de datos utilizó el módulo Pipeline de Scikit-Learn el cual permite implementar 'transformadores' o etapas de forma personalizada. Un transformador debe extender la clase base TransformerMixIn e implementar los métodos fit y transform.

Se implementaron 2 pipelines que resuelven las etapas de Data Selection, Preprocessing y Transformation. Ambos se utilizan como componentes en otro pipeline el cual es utilizado para procesar los datos y obtener una versión de los mismos lista para aplicar técnicas de Aprendizaje Automático. La razón de la creación de 2 pipelines es mantener separadas las etapas de preprocesamiento y poder ser utilizadas de forma independiente.

El pipeline encargado de Data Selection y Preprocessing dispone de las siguientes funciones:

- ColumnSelector: Selección de los atributos especificados
- CleanData: Elimina o cambia el nombre de columnas de un dataframe y reemplaza valores erróneos (fuera del dominio del atributo).
- MultiLabelEncoder: Codificación de atributos categoriales de un dataframe
- DataFrameOneHotEncoder: Codificación de atributos categoriales de un dataframe en una columna binaria por cada valor posible.
- DataFrameMapColumns: Mapeo de valores de una columna de un dataframe utilizando un objeto (dict o Series) o una función.

- AddFeatureMissingValuesCount: Agrega un atributo con la cantidad de valores faltantes de cada fila.

Por otro lado el pipeline de Transormation dispone de las siguientes funciones:

- AddDateFeatures: A partir de la fecha de nacimiento y del día de la prueba se generan atributos para el día de la semana, mes y año de la prueba; edad y año de nacimiento.
- DataFrameImputer: Reemplaza valores nulos
- AddAssessmentCounters: Genera los atributos 'n_entregas' (cantidad de entregas), 'n_insuficientes' (cantidad de insuficientes), 'insufentregas' (cociente entre los atributos anteriores), 'n_entregas_coc' (cociente entre las entregas realizadas y la cantidad máxima de entregas posibles).
- ProcessAssessments: Unifica en una sola columna las entregas tomando el valor de la entrega suficiente. Además genera las columnas 'n_entregas' (cantidad de entregas), 'n_insuficientes' (cantidad de entregas insuficientes) y 'aprobado' (valor binario de aprobación). Por último elimina filas con cantidad de valores nulos mayor a la mitad de la cantidad de actividades.
- DropCols: Elimina las columnas especificadas
- DropOneValueCols: Elimina columnas que tienen sólo un valor repetido en cada fila

7.1.2. Librería para la evaluación y optimización de estimadores y selección de atributos

Para automatizar las etapas de selección de atributos (utilizando algoritmos), comparación de estimadores y su posterior optimización, se implementó una librería.

Para esto se utilizó como base una clase llamada EstimatorSelectionHelper[57] implementada por Panagiotis Katsaroumpas para realizar una optimización de estimadores utilizando el módulo GridSearchCV y luego una comparación/selección de los mismos; la cual fue extendida para soportar la selección de atributos y la concatenación de uno o más algoritmos de selección con un modelo predictivo.

Además se agregaron las siguientes funcionalidades:

- Se implementó un método de transformación genérica para poder utilizar uno o más algoritmos de transformación por ejemplo: Normalizer, StandardScaler, RobustScaler, PolynomialFeatures. También se implementó un método independiente para cada uno de éstos algoritmos.
- Validación cruzada para cada uno de los estimadores de entrada, disponible según las distintas versiones implementadas en ScikitLearn: cross_val, cross_val_predict, kfold, cross_val_repeated.
- Obtención de resultados utilizando métricas de regresión en formato de tabla para poder realizar la comparación y ver los resultados

- Obtención de los mejores parámetros y atributos para cada estimador
- Nuevos parámetros de entrada:
- Diccionario de estimadores y parámetros de optimización utilizando un esquema más flexible y sencillo que el requerido para GridSearchCV
  Esquema: {'nombreEstimador': {'estimador': modelo, 'params': parámetros }}
- Algoritmos de selección de atributos y sus respectivos parámetros de optimización
- Lista de posibles cantidades de atributos a seleccionar (utilizado como parámetro de optimización para cada selector de atributos
- Transformadores de datos.

A continuación se describen los métodos implementados.

- transformInput: Aplicar un transformador recibido como parámetro a los datos especificados
- selectBestFeatures: Para cada modelo obtenido como parámetro, se genera un pipeline de 2 pasos, el primero compuesto por un objeto de tipo FeatureUnion con los algoritmos de selección automática de atributos pasados por parámetro y el segundo contiene el modelo.
- getEstimatorMetrics: Retorna métricas de regresión o clasificación a partir de los parámetros 'y' (variable independiente), 'y_pred' (predicción) de entrada.
- getMetrics: Retorna un Dataframe con los valores de las métricas de performance del estimador. Utiliza el método anterior, 'getEstimatorMetrics', para obtener los resultados.
- Grid: Para cada modelo se genera uno nuevo utilizando el método GridSearchCV y se utilizan los datos de entrada para entrenar utilizando el método fit.
- Grid_predict: Para cada modelo se obtiene una predicción a partir del conjunto de test recibido como parámetro y se devuelve un dataframe con los resultados utilizando el método getEstimatorMetrics descripto anteriormente.
- getBestParams: Retorna los parámetros obtenidos en la optimización de cada modelo utilizando el método grid_predict.
- getBestEstimators: Retorna para cada modelo el mejor estimador obtenido luego de la optimización de parámetros con grid_predict.
- getBestFeatures: Retorna los atributos más valorados obtenidos mediante los algoritmos de selección de atributos establecidos.
- manualCrosVal: Realiza una validación cruzada para cada modelo utilizando la clase KFold y almacena los resultados en el atributo grid_metrics.

- Cross_val: Realiza una validación cruzada para cada modelo utilizando el método cross_val_score y almacena los resultados en el atributo grid_metrics.

Con la implementación de los métodos mencionados la librería permite seleccionar un conjunto de modelos y algoritmos de selección y obtener la comparación de sus resultados (al predecir un conjunto de datos dado) y la optimización de los mismos.

Esto permite automatizar las etapas mencionadas de forma tal de agilizar el análisis de nuevos conjuntos de datos.

7.2. Backend

Con el objetivo de exponer las librerías de automatización se implementó una API REST con el framework Flask para python. El lenguaje python es el mismo que se utilizó para implementar las librerías. Se creó una carpeta llamada 'helpers' de la cual se importan las mismas.

7.2.1. API REST

Se implementaron puntos finales para brindar las funcionalidades cargar dataset, crear un modelo (entrenar un algoritmo con los datos de entrada), predecir resultados y obtener métricas de la performance del modelo.

7.2.2. Trabajos en segundo plano

Los procesos de entrenamiento y predicción, pueden demorar un tiempo considerable que exceda el tiempo límite de respuesta del servidor web. Para solucionar esto, se decidió integrar la librería de cola de trabajos Redis Queue para ejecutar las tareas en segundo plano. Como complemento, se decidió también integrar la librería flask-socketio para enviar notificaciones en tiempo real del estado de la tarea mediante websockets.

7.3. Frontend

Se implementó una aplicación web con el fin de brindar una interfaz gráfica para facilitar las tareas de análisis de datos tanto antes como después de realizar predicciones, generación de un modelo predictivo y evaluación de la performance del mismo.

7.3.1. Dashboard

Se dispone de un dashboard con indicadores en la parte superior con números y colores de fondo (verde, amarillo y rojo) para visualizar rápidamente el estado de los mismos. La correspondencia de cada color es la siguiente: verde - correcto, amarillo - alerta, rojo - mal funcionamiento. Debajo de los indicadores de muestra una tabla con el identificador del estudiante y la nota predicha para cada uno.

7.3.2. Formulario paso a paso (Wizard)

Se implementó un formulario para guiar la generación del modelo. Se cuentan con los siguientes pasos: carga de datos de entrenamiento, entrenamiento, carga de datos para predecir.

7.3.3. Performance

Otra sección disponible es la de performance para poder evaluar el modelo obtenido. Los datos que se muestran son: nombre del algoritmo utilizado, score, métricas de evaluación de desempeño y un icono para indicar gráficamente el estado. El score y las métricas no son siempre las mismas ya que varían según el tipo de algoritmo, que en la presente solución se manejan dos, clasificación y regresión. El icono que indica el estado es un sol si la performance es buena, una nube si es aceptable y lluvia en el caso contrario. La idea fue tomada de la herramienta Jenkins en la cual se utilizan los mismos iconos de utilizados para el clima indicando el estado de los últimos despliegues. Luego de esto también se muestra una tabla con el identificador de cada estudiante, la nota de la porción de los datos de entrenamiento tomados como de verificación, la nota predicha y la diferencia entre las mismas. Esto permite dar otra idea de la magnitud del error en las predicciones.

7.3.4. Resultados

Se muestra una tabla con las predicciones

Capítulo 8

Gestión del proyecto

El presente proyecto tuvo sus comienzos en el mes de abril del 2016, con la idea de los dos autores de presentar un proyecto que combinara aspectos de aprendizaje automático aplicados a un área en el cual sea de gran utilidad y se detectara una necesidad de investigación. Las áreas propuestas fueron educación y eficiencia energética. En las primeras reuniones con los profesores se decidió que el área de aplicación sea la educación y que se tomaría como caso de estudio la carrera de Testing del Centro de Ensayo de Software en donde los tutores han dictado clases. Durante el 2016 se estuvo realizando el estudio del estado del arte, para lo cuál se investigaron algunas de las actuales aplicaciones del aprendizaje automático a la educación como se detalla en el capítulo 2. A finales de ese año, se evaluaron tecnologías y herramientas para abordar el análisis de datos y la generación de modelos predictivos.

En el año 2017 se comenzó la parte práctica. En el primer semestre el trabajo consistió en la extraer los datos de 2 fuentes distintas, la base de datos de Moodle y la de bedelías. Esto se describe en los capítulos comprendidos entre el 3 y el 5. Se destaca la complejidad para extraer y entender los datos de las distintas bases de datos aportadas por el Centro de Ensayo de Software, que requirieron muchas reuniones entre los autores y los tutores para obtener un conjunto de datos apropiado sobre el cual poder realizar un análisis, así como también la falta de conocimiento de uno de los autores en las herramientas de aprendizaje automático y del área en general. Dado esto último, desde el 2016 y 2017, en paralelo a las tareas ya mencionadas, se debió agregar el estudio de las técnicas de análisis y generación de modelos predictivos con Machine Learning. Para esto se recurrió a cursos online y material extraído de la web. Sobre el segundo semestre de 2017 hasta mediados de 2018 se evaluaron y aplicaron técnicas de aprendizaje automático a los cursos del Centro de Ensayo de Software. Al principio, se tenía como objetivo predecir si un alumno va a salvar o no el curso de ITP, pero luego se decidió incorporar algo más complejo, que consistió en predecir la nota exacta del mismo. La complejidad estuvo no sólo en que se tuvieron que generar modelos de regresión, que implica mayores desafíos que los de clasificación, sino que ninguno de los 2 autores había trabajado con regresión ni con optimización de parámetros. Esto tuvo como desafío el aprendizaje del uso de algoritmos para regresión y de cómo validar los modelos, así como también la automatización de las tareas

de preprocesamiento, selección de atributos, selección del mejor estimador, optimización del modelo y validación para comparar distintos algoritmos.

En cuanto a la división de tareas, los autores decidieron repartir las tareas de generar modelos de clasificación y regresión, realizando puestas en común periódicamente sobre los resultados alcanzados.

Sobre mediados del año 2018 se comenzó con la documentación del trabajo realizado en el presente informe. Luego, a comienzos del presente año 2019, se decidió además implementar un sistema con interfaz web, para facilitar al usuario la aplicación de cada una de las etapas necesarias a los datos de estudio. Se considera uno de los aprendizajes y puntos débiles del presente proyecto el no haber realizado la redacción de la documentación en paralelo al análisis y procesamiento de datos, ya que provocó una gran demora el tener que volver a refrescar el trabajo realizado luego de haber finalizado.

Capítulo 9

Conclusiones

Del presente trabajo se extraen varias conclusiones que se describen a continuación. En primer lugar, en base a lo investigado y a los resultados obtenidos en las pruebas realizadas, se concluye que es viable la aplicación de técnicas de aprendizaje automático para construir un modelo predictivo del resultado académico de estudiantes. Esto se concluye tanto en base a la literatura consultada como por los resultados obtenidos en los estudios realizados sobre la carrera de Testing del Centro de Ensayo de Software.

9.1. Resultados obtenidos

En base a los resultados obtenidos en el análisis de clasificación y regresión se concluye que los resultados obtenidos con modelos de clasificación son más certeros para el problema planteado de predecir la aprobación de los estudiantes porque se mejoró en un valor de 0.10 el F1-Score de un clasificador de base. En el caso de regresión se obtuvieron valores de R2 score positivos, lo cual indican que son mejores que el predictor de base y valores muy bajos de MSE y RMSE. Sin embargo, al comparar las notas obtenidas de los modelos con las reales se detectó un margen de error de valor 3. Esto implica, por ejemplo, que si el modelo de regresión predice una nota de 4, la nota real del estudiante puede ser 7 (sumando 3) o 1 (restando 3) para los casos extremos. El aspecto negativo de este resultado es que existen casos en los que no se puede afirmar si el estudiante va a aprobar o no. Por otro lado se destaca la consistencia de que el margen no fue mayor en ninguna de las pruebas, lo cual en caso de que la predicción sea una nota de 10 o más, el estudiante es candidato a aprobar ya que según el margen manejado la nota real puede estar entre 7 y 12. Con el mismo razonamiento, dada una predicción de una nota de 3, el estudiante es candidato a reprobar.

9.2. Calidad de datos

Se observa que los resultados del modelo dependen fuertemente de la calidad y cantidad de los datos. Como se mostró en la sección de Obtención de Datos del Capítulo 3, diversos

estudios muestran la relevancia de diferentes datos a tener en cuenta, como el comportamiento del estudiante, el histórico del rendimiento académico o las interacciones en el ámbito social, para mejorar los resultados de las predicciones.

Al comenzar con el análisis de los datos del CES se tuvo como objetivo predecir la aprobación de los estudiantes del curso Introducción al Testing 1. Allí se encontró con que la enorme mayoría de los estudiantes aprueba el curso por lo que el conjunto de datos se encuentra muy desbalanceado. Debido a esto, sumado a la poca cantidad de datos a disposición, habían pocos estudiantes que reprobaron como para que un algoritmo pueda realizar algún tipo de generalización. De aqui se concluye que es fundamental tener a disposición un conjunto de datos balanceados.

Luego se analizó el curso de Introducción al Testing de Performance ya que tiene asignaturas previas, con notas de actividades, que fueron incluidas en el conjunto de datos. Uno de los inconvenientes encontrados fue que se cuenta con un histórico de pocos centenares de estudiantes, lo cual se entiende que dificulta el aprendizaje de los algoritmos ya que podrían detectar un patrón a partir de ciertos casos particulares, luego tomarlos como válidos para el modelo pero que no refleje una realidad que pueda generalizarse a los futuros estudiantes. Debido a esto se concluye que es muy importante tener una cantidad considerable de datos para que al aplicar métodos de aprendizaje automático se obtenga un modelo que sea lo más fiel posible a la realidad.

9.3. Desarrollo con Scikit Learn

Antes de comenzar con este proyecto el conocimiento sobre aprendizaje automático era básico. Se observa que la curva de aprendizaje tanto en lo teórico como en lo práctico para comenzar a aplicar las técnicas de aprendizaje automático, en su etapa básica o inicial, no es de una complejidad excesiva, sino que es algo bastante sencillo. Se accedió a una gran cantidad de material y cursos introductorios en internet sobre las distintas técnicas disponibles para aplicar. La herramienta utilizada para el desarrollo es una librería de Python llamada Scikit Learn. Por lo experimentado actualmente se encuentra madura y muy bien documentada, en particular en lo relacionado a las técnicas de clasificación. Si bien al tener el primer acercamiento al área de aprendizaje automático se debieron asimilar las particularidades de muchas de las técnicas aplicadas, se concluye que fue un ejercicio de valor para entender qué era lo que estaba ocurriendo y por qué los resultados no daban como se esperaba. En algunos casos no fue algo estrictamente necesario ya que la librería abstrae en gran medida la manipulación de los algoritmos y con pocas líneas de código permite obtener una predicción aceptable.

En cuanto a los aspectos negativos de Scikit Learn se destaca que no cuenta con soporte oficial para balancear conjuntos de datos, aunque hay complementos de la librería que permiten hacerlo, y que el soporte para regresión no está tan maduro como el de los algoritmos de clasificación.

9.4. Algoritmos de Machine Learning

Uno de los aspectos que nos sorprendió en cuanto a la experimentación con técnicas de Machine Learning fue que aunque se conozca el funcionamiento de los algoritmos, es muy difícil predecir antes de entrenar y obtener resultados, cuáles son los algoritmos más adecuados para el problema.

El enfoque utilizado fue probar con un conjunto de algoritmos y comparar los resultados obtenidos en cada uno para seleccionar el que devolvía el mejor score. Ésto no implica una gran cantidad de tiempo de procesamiento por lo cual se entiende que es una buena estrategia para no descartar a priori ningún algoritmo.

Se concluye que la mayor dificultad del proyecto no se encontró en la aplicación de algoritmos de aprendizaje automático sino en la obtención y preparación de los datos.

9.5. Obtención de datos y preprocesamiento

Las etapas de obtención de datos y preprocesamiento fueron de las etapas más complejas del proyecto.

Los factores que afectaron la complejidad y el tiempo dedicado fueron varios y se detallan a continuación.

9.5.1. Obtención de datos pertinentes a la nota final del curso

Al inicio, surgió el inconveniente de qué datos seleccionar que tengan valor para realizar las predicciones planteadas.

Como primer intento, se tomaron sólo las actividades del curso para predecir la nota final, pero los resultados indicaron que no eran suficientes. También se agregaron datos personales, que había poca cantidad, pero no varió mucho el resultado.

Luego se decidió tener en cuenta todo lo relacionado a los cursos previos, tanto la nota final como las actividades. Esto trajo como consecuencia disponer de un gran número de atributos, lo cual al existir muy pocos registros de estudiantes por cada curso, afectó negativamente en los resultados. Ésto se debe a que, si bien no hay una fórmula para obtener la cantidad óptima de atributos a seleccionar, teniendo en cuenta la cantidad de registros, en la práctica se observó que hay un punto de inflexión en la relación entre cantidad de atributos y cantidad de registros en la cual manejar más atributos impacta negativamente en el resultado.

9.5.2. Obtención de los atributos de mayor valor

Dado lo mencionado anteriormente el siguiente problema a resolver consistió en probar algoritmos de selección de atributos y reducción de dimensiones para mejorar la relación de cantidad de atributos y registros mencionada anteriormente.

Los algoritmos fueron de gran ayuda pero también se realizaron análisis gráficos de los datos, incluyendo la correlación entre los atributos y la nota final del curso, para luego manualmente quitar los que se detectaban como irrelevantes. También se generaron nuevos atributos a partir de la agrupación de otros ya existentes para reducir su cantidad y también revelar otras características que no quedan claras con los datos en crudo.

Capítulo 10

Trabajo a futuro

En el presente proyecto, se cumplió con el objetivo principal de desarrollar un procesador de datos educativos (Learning Analytics Processor) con el cual poder obtener nuevos datos que permitan una mejor comprensión del estado actual de cada estudiante y tomar acciones preventivas para mejorar la calidad del aprendizaje y evitar deserciones. Sin embargo, aunque el procesador de datos es el núcleo y la parte fundamental del sistema, se pueden implementar otros componentes para enriquecer cada vez más el sistema y así brindar más servicios a los usuarios que mejoren la calidad y agilidad para la toma de decisiones y ejecución de acciones lo más temprano y a tiempo posible. Un ejemplo de esto sería desarrollar aplicaciones con interfaz web y móvil para los estudiantes, profesores y otros participantes con otros cargos, brindando más facilidades y comodidades. Además se puede tener un sistema de alertas, notificaciones entre el procesador y las aplicaciones mencionadas, para brindar un servicio en tiempo real. Por otro lado, se debería implementar una API que permita el envío de datos de interacciones entre estudiantes y profesores, entre otros acerca del contexto y actividades que realiza el usuario para enriquecer el modelo predictivo y mejorar la precisión.

A continuación se propone una solución para facilitar la recolección y el análisis de los datos educativos buscando obtener los mejores resultados posibles y al mismo tiempo minimizar algunos de los problemas encontrados durante el análisis, detallados en el capítulo anterior.

Un aspecto importante a destacar es que la solución contempla que sus componentes se encuentren desacoplados y se puedan reemplazar por otros que cumplan la misma función.

Se toma como base las lecciones aprendidas en el capítulo de desarrollo, así como las mejores prácticas recomendadas en la literatura.

10.1. Recolección y almacenamiento de datos

En primer lugar se plantea un módulo para la recolección de datos con el objetivo de recopilar información relevante relacionada a los estudiantes, normalizar y estructurar dicha información en una base de datos, o Data Warehouse, para luego poder realizar los análisis predictivos sobre la misma.

10.1.1. Datos relevantes

Las bases de datos de Bedelías y Moodle utilizadas en la sección anterior, manejan datos desestructurados como logs de acceso, donde se suelen almacenar las interacciones del usuario con el sistema: los accesos a los recursos, el tiempo de permanencia en la plataforma y la interacción con recursos interactivos.

Según el estudio de Elaf Abu Amrieh et al.[54] los datos relacionados con el comportamiento de los estudiantes son muy valiosos a la hora de predecir el rendimiento académico, por lo cual es de gran importancia poder extraer datos de los logs de los usuarios.

Además existen otras investigaciones en las cuales se sugiere realizar encuestas a los estudiantes para obtener información sobre los métodos de estudio utilizados y que realicen una autoevaluación sobre su rendimiento y expectativas.

Por último, existe otra importante fuente de datos que se puede integrar, la cual se obtiene como producto del análisis de las redes sociales de los estudiantes.

10.1.2. Almacenamiento

Para normalizar las diversas fuentes de datos mencionadas se recomienda la utilización del estándar xAPI, presentado en el capítulo 2, debido a que facilita la integración y extracción de los mismos para su posterior análisis.

El estándar requiere utilizar el sistema de almacenamiento conocido como LRS, el cual es un elemento indispensable para implementar el mismo.

10.2. Sistema de alertas

Por otro lado, se plantea un componente cuyo objetivo es el procesamiento de la salida del módulo anterior y generar alertas para los usuarios.

Este módulo es el encargado de presentar los datos de forma que sea entendible por individuos no especializados en el análisis de los datos, para que puedan extraer valor de los análisis realizados. Se pueden mostrar en formato de tablas o gráficos, así como generar alertas en tiempo real cuando se detecten estudiantes en riesgo de perder el curso.

10.3. Consentimiento del estudiante

En el capítulo 2.4. Learning Analytics, cuando se describen los actores participantes en la aplicación de Learning Analytics, se menciona el problema de confidencialidad que surge al almacenar datos personales ya que deben almacenarse de forma tal que se preserve el anonimato.

Para contemplar esto se debe implementar un módulo que sea lo más flexible posible en cuanto a los permisos de uso y almacenamiento de los datos que el usuario brinde.

En cuanto a la flexibilidad, se entiende y sugiere que el módulo brinde información detallada de la manipulación de los datos y la posibilidad de seleccionar el conjunto de los mismos a tratar.

Se considera que si se brinda un simple contrato de términos y uso con una casilla de verificación para aceptar, es el peor caso a implementar ya que no se ofrece transparencia, confianza ni la libertad de filtrar datos que por alguna razón no se desee compartir.

Además los permisos deben poder ser actualizados en cualquier momento por parte del usuario.

Por último, dado que puede ocurrir que ciertos datos sean muy importantes para algunos resultados como por ejemplo recomendaciones o predicciones, se sugiere indicarlo para que el usuario conozca las limitaciones o degradaciones en los resultados que puedan ocurrir.

10.4. Dashboard para el estudiante

Uno de los objetivos principales de la solución es brindarle al estudiante información útil de apoyo en la toma de decisiones con el fin de que el aprendizaje sea de la mayor calidad posible y se cumplan sus metas relacionadas al avance de la carrera o la finalización de cursos.

Para esto se propone implementar un dashboard que brinde la situación actual del estudiante, ya sea con gráficas, tablas, etc; recomiende métodos de estudio o recursos personalizados (ver videos, diapositivas, pdfs, libros, artículos) así como también cursos que pueden ser de interés; y realice predicciones, por ejemplo referidas a la aprobación de un curso o diploma.

10.5. Dashboard para el profesor

Otro de los objetivos clave de Learning Analytics es brindar herramientas para los responsables de la enseñanza de un estudiante, ya sea el profesor o los profesores como para otras personas con mayor jerarquía. Una herramienta clásica, es brindar un dashboard en el cual se puedan exponer datos relacionados a los estudiantes utilizando distintos métodos de visualización. Esto permite realizar análisis visuales aprovechando las habilidades humanas para descubrir patrones, dado que en muchos casos son más fáciles de descubrir que analizando datos numéricos o 'en crudo'.[55]

Los dashboards que se pueden implementar son muy variados y dependen de los requerimientos y las necesidades de cada proyecto.

A continuación se detallan ejemplos de funcionalidades interesantes que puede brindar un dashboard[56]:

- información referente a resultados obtenidos para una tarea dada y el grado de cumplimiento de la misma
- lista de los alumnos con:
- estado de riesgo en el curso (alto, medio, bajo)
- grado de participación en las distintas actividades de la asignatura (alto, medio, bajo)
- grafo de interacción curso-alumnos

Bibliografía

[1] 1st international conference on learning analytics and knowledge 2011. https://tekri.athabascau.ca/analytics/. Último acceso: 02-06-2019.

[2] Aprendizaje profundo y superficial. http://www2.udec.cl/ofem/recs/anteriores/vol412007/esq41.htm. Último acceso: 02-07-2019.

[3] Calendario de cursos. https://capacitacion.ces.com.uy. Último acceso: 02-10-2019.

[4] Estudio sobre deserción en facultad de ingeniería. https://www.fing.edu.uy/sites/default/files/claustro_citaciones/2013/distribuido/7962/19-2013%20Informe%20Deserci%C3%B3n%20UEFI.pdf. Último acceso: 02-07-2019.

[5] Estudio sobre deserción en facultad de ingeniería. https://www.fing.edu.uy/~enrich/claustro2013/desempenioEstudiantil/DesempenioEstudiantil.pdf. Último acceso: 02-07-2019.

[6] Machine Learning: What it is and why it matters. http://www.sas.com/en_id/insights/analytics/machine-learning.html. Último acceso: 02-06-2019.

[7] Machine Learning y Deep Learning. Todo lo que necesitas saber. https://spartanhack.com/machine-learning-y-deep-learning-todo-lo-que-necesitas-saber/. Último acceso: 02-06-2019.

[8] R^2 score, the coefficient of determination. https://scikit-learn.org/stable/modules/model_evaluation.html#r2-score. Último acceso: 02-06-2019.

[9] Sitio oficial de la sociedad solar. https://solaresearch.org/about/. Último acceso: 02-06-2019.

[10] Ley de protección de datos personales - ley nº 18331. https://www.impo.com.uy/bases/leyes/18331-2008, 2008.

[11] Educational data mining: A review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, IEEE Trans. Syst., Man, Cybern. C*, (6):601, 2010.

[12] The next step for learning analytics. *IT Professional, IT Prof*, (5):4, 2014.

[13] Mae and rmse — which metric is better? https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d, 2016. Último acceso: 02-06-2019.

[14] Ignasi Alcalde. Learning analytics: el big data de la educación. https://ignasialcalde.es/learning-analytics-el-big-data-de-la-educacion/. Último acceso: 02-06-2019.

[15] Doug Alexander. Data Mining. http://www.laits.utexas.edu/~anorman/BUS.FOR/course.mat/Alex/, 2010. Último acceso: 02-06-2019.

[16] Mohamed Amine Chatti, Anna Dyckhoff, Ulrik Schroeder, and Hendrik Thüs. A reference model for learning analytics. *International Journal of Technology Enhanced Learning*, 4:318–331, 01 2012.

[17] Cláudia Antunes. Anticipating student's failure as soon as possible. http://web.ist.utl.pt/claudia.antunes/artigos/antunes10hedm.pdf, 2014. Último acceso: 02-06-2019.

[18] Ryan S. J. D. Baker and Kalina Yacef. The state of educational data mining in 2009: A review and future visions. *Journal of Educational Data Mining*, 1(1):3 – 16, 2009.

[19] Paulo Cortez and Alice Maria Gonçalves Silva. Using data mining to predict secondary school student performance. 2008.

[20] Alan Descoins. Why accuracy alone is a bad measure for classification tasks, and what we can do about it. https://tryolabs.com/blog/2013/03/25/why-accuracy-alone-bad-measure-classification-tasks-and-what-we-can-do-about-it/, 2013. Último acceso: 02-06-2019.

[21] Georgios Drakos. How to select the right evaluation metric for machine learning models: Part 1 regression metrics. https://towardsdatascience.com/how-to-select-the-right-evaluation-metric-for-machine-learning-models-part-1-regrre 2018. Último acceso: 02-06-2019.

[22] Robert A. Ellis, Feifei Han, and Abelardo Pardo. Improving learning analytics–combining observational and self-report data on student learning. *Educational Technology Society*, 20(3):158 – 169, 2017.

[23] Center for Instructional Technology Training. Collecting data. http://citt.ufl.edu/online-teaching-resources/learning-analytics/collecting-data/, 2018. Último acceso: 02-06-2019.

[24] The Advanced Distributed Learning Initiative. Research development: The adl initiative develops and assesses distributed learning prototypes that enable more effective, efficient,

and affordable learner-centric solutions. https://www.adlnet.gov/research/scorm/. Último acceso: 02-06-2019.

[25] Holcomb John P. Applied regression analysis norman r. draper harry smith. *The American Statistician*, 53(2):170, 1999.

[26] Avita Katal, Mohammad Wazid, and R. H. Goudar. Big data: Issues, challenges, tools and good practices. *2013 Sixth International Conference on Contemporary Computing (IC3)*, pages 404–409, 2013.

[27] Smart Klass. Descripción de la empresa klass data. https://www.tecnoempleo.com/klass-data/re-176589. Último acceso: 02-06-2019.

[28] Smart Klass. Descripción de la empresa klass data. http://klassdata.com/smartklass-learning-analytics-plugin/. Último acceso: 02-06-2019.

[29] Igor Kononenko. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in Medicine*, 23(1):89 – 109, 2001.

[30] Frederic Lardinois. Duolingo adds offline mode and speech recognition to its mobile app. https://techcrunch.com/2013/03/14/duolingo-adds-offline-mode-and-speech-recognition-to-its-mobile-app/. Último acceso: 02-06-2019.

[31] Frederic Lardinois. Duolingo launches free language learning platform for schools. https://techcrunch.com/2015/01/08/duolingo-launches-free-language-learning-platform-for-schools/. Último acceso: 02-06-2019.

[32] Patricia Malagón. Las nuevas tecnologías como herramienta para mejorar la carrera académica. http://www.expansion.com/directivos/damos-cuerda/2015/08/06/55c31cbe46163f87248b4570.html. Último acceso: 02-06-2019.

[33] Moodle. Smartklass™ learning analytics moodle. https://moodle.org/plugins/local_smart_klass. Último acceso: 02-06-2019.

[34] Moodle. Plugin types. https://docs.moodle.org/dev/Plugin_types, 2017. Último acceso: 02-06-2019.

[35] OECD. Creating effective teaching and learning environments. http://www.oecd.org/education/school/43023606.pdf. Último acceso: 02-06-2019.

[36] Emil Protalinski. 100m users strong, duolingo raises 45*mledbygoogleata*470m valuation to grow language-learning platform. https://venturebeat.com/2015/06/10/100m-users-strong-duolingo-raises-45m-led-by-google-at-a-470m-valuation-to-grow-lar
Último acceso: 02-06-2019.

[37] Steve Ranger. Duolingo: The app teaching humans new languages while teaching machines how to learn. https://www.zdnet.com/article/duolingo-the-app-teaching-humans-new-languages-while-teaching-machines-how-to-learn Último acceso: 02-06-2019.

[38] Ferguson Rebecca. Learning analytics: drivers, developments and challenges. *TD Tecnologie Didattiche*, (3):138, 2014.

[39] SCORM. Scorm explained 201: A deeper dive into scorm. https://scorm.com/scorm-explained/. Último acceso: 02-06-2019.

[40] Christos Stergiou and Dimitrios Siganos. Neural networks. https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html, 2015. Último acceso: 02-06-2019.

[41] Wei Sun and Yanfeng Xu. Using a back propagation neural network based on improved particle swarm optimization to study the influential factors of carbon dioxide emissions in hebei province, china. *Journal of Cleaner Production*, 112:1282 – 1291, 2016.

[42] The Room 241 Team. Strategies to improve classroom behavior and academic outcomes. https://education.cu-portland.edu/blog/classroom-resources/strategies-to-improve-classroom-behavior-and-academic-outcomes/, 2012. Último acceso: 02-06-2019.

[43] TICAP. ¿por qué son importantes los estándares en el elearning? http://www.ticap.mx/blog-por-que-son-importantes-los-estandares-en-elearning/. Último acceso: 02-06-2019.

[44] Vincent Tinto. *From Theory to Action: Exploring the Institutional Conditions for Student Retention*, pages 51–89. Springer Netherlands, Dordrecht, 2010.

[45] Jacob Whitehill, Joseph Williams, Glenn Lopez, Cody Coleman, Justin Reich, and Society International Educational Data Mining. Beyond prediction: First steps toward automatic intervention in mooc student stopout., 2015.

[46] J. Willmott Cort and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79, 2005.