
Predicción Temprana del Rendimiento Académico en Cursos Universitarios: Un enfoque en Cálculo 1 con *Educational Data Mining* y *Learning Analytics*

Luis Pedro Cuéllar Pineda



UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Predicción Temprana del Rendimiento Académico en
Cursos Universitarios: Un enfoque en Cálculo 1 con
*Educational Data Mining y Learning Analytics***

Trabajo de graduación en modalidad de tesis presentado por
Luis Pedro Cuéllar Pineda
para optar al grado académico de Licenciado en Ciencias de la
Computación y TI

Guatemala,
2023

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Predicción Temprana del Rendimiento Académico en
Cursos Universitarios: Un enfoque en Cálculo 1 con
*Educational Data Mining y Learning Analytics***

Trabajo de graduación en modalidad de tesis presentado por
Luis Pedro Cuéllar Pineda
para optar al grado académico de Licenciado en Ciencias de la
Computación y TI

Guatemala,
2023

Vo.Bo.:



(f) _____
Carlos Ernesto Celada Correa

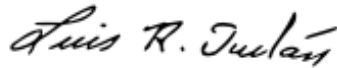
Tribunal Examinador:



(f) _____
Carlos Ernesto Celada Correa



(f) _____
Douglas Leonel Barrios Gonzales



(f) _____
Luis Roberto Furlan Collver

Fecha de aprobación: Guatemala, 21 denoviembre de 2023.

Agradecimientos

Durante el desarrollo del proyecto hubo varias personas que aportaron a que este fuera finalizado. Dentro esas personas quisiera resaltar y agradecer a cuatro grupos de personas en específico.

Primero, Lynette Garcia. Gracias Lynette por abrirme las puertas a su proyecto y permitirme aportar al mismo. Gracias por estar abierta a responder cualquier duda que iba surgiendo durante el desarrollo.

Segundo, Douglas Barrios. Gracias Douglas por el constante apoyo y seguimiento durante, no solo el desarrollo de este proyecto sino que también en la carrera entera. Gracias por brindarme este tema para yo poder realizarlo. Gracias también por servir de intermediario cuando se requería hablar con otra gente.

Tercero, Carlos Celada. Gracias Carlos por aceptar ser asesor de este proyecto. Por brindar ese apoyo para resolver cualquier duda que iba surgiendo. Por brindar recomendaciones a cualquier obstáculo con el que me topé en el desarrollo. También, por todas las insights que me ibas dando, no solo como profesional en el área, sino que también como catedrático.

Cuarto, Silvia Pineda, Osberto Cuéllar y María Isabel Montoya. Gracias a todos ustedes por apoyar siempre, nunca dejar que se me pasara el hacer la tesis. El constante recordatorio que la tenía que hacer y terminar. Gracias por compartirme los ánimos cuando ya no tenía ganas de hacer nada.

Gracias a estos cuatro grupos de personas, ya que sin ustedes, este trabajo no hubiera sido posible. Desde la idea, durante el desarrollo y hasta terminarlo.

Agradecimientos	III
Índice de figuras	VI
Lista de cuadros	VIII
Resumen	IX
1. Introducción	1
2. Objetivos	3
2.1. Objetivo general	3
2.2. Objetivos específicos	3
3. Justificación	4
4. Marco teórico	5
4.1. Aspectos conceptuales	5
4.1.1. Inteligencia artificial	5
4.1.2. Machine Learning	5
4.1.3. Ciencia de datos	8
4.1.4. Educational Data Mining	8
4.1.5. Learning Analytics	9
4.1.6. Educational Warning System	10
4.1.7. Feature Engineering	11
4.1.8. Selección de variables	13
4.1.9. Métricas de rendimiento	15
4.2. Caso de éxito	16
4.2.1. Usando Learning Analytics para desarrollar un sistema de alerta temprana	16
5. Metodología	18
5.1. Descripción general	18
5.2. Herramientas utilizadas	19
5.3. Primera etapa	19
5.3.1. Pre-procesamiento	19
5.3.2. Feature engineering	31
5.3.3. Selección de variables	41
5.3.4. Escoger el mejor algoritmo	42

5.4. Segunda etapa	45
6. Resultados	46
6.1. Primera etapa	46
6.2. Segunda etapa	56
7. Análisis de resultados	59
7.1. Primera etapa	59
7.1.1. Primer análisis	60
7.1.2. Segundo análisis	60
7.2. Segunda etapa	61
8. Conclusiones	63
9. Bibliografía	65

4.1. Diagrama de qué es aprendizaje por transferencia [2]	6
4.2. Diagrama de qué es aprendizaje por refuerzo [30]	7
4.3. Diagrama de qué es aprendizaje federado [44]	7
4.4. Diagrama de Venn que define qué es Ciencia de Datos [12]	8
5.1. Cursos y cantidad de estudiantes asignados en el dataset	21
5.2. Actividad en las cuales la proporción de nulos es 100 %	22
5.3. Una actividad en la que la proporción de nulos es menor al 100 %	22
5.4. Ambas actividades mostradas en la misma tabla	22
5.5. Estudiante cuya nota en todas las actividades es en nulo	23
5.6. Misma actividad pero con diferentes fechas en un mismo estudiante	23
5.7. Actividades con puntos posibles nulo	24
5.8. Actividades mostrando la diferencia de los valores puros vs los valores luego de la limpieza	25
5.9. Valores nulos en la columna de <i>FechaCalificacion</i> y el valor final	26
5.10. Valores nulos en la columna de <i>FechaVencimiento</i> y el valor final	26
5.11. Valores nulos en la columna de <i>FechaTodoElDia</i> y el valor final, algunos valores con nulo y otros con la fecha final	27
5.12. Valores nulos en la columna de <i>FechaTodoElDia</i> y el valor final, sin valores nulos	28
5.13. Nota en la escala de porcentajes	29
5.14. Nota en la escala de porcentajes con el símbolo %	29
5.15. Nota en la escala de porcentajes con el símbolo %	30
5.16. Misma <i>Actividad</i> con diferente <i>Nota</i> y <i>FechasCalificacion</i>	30
5.17. <i>Actividad</i> con <i>Nota</i> complete e incomplete	31
5.18. Saltos entre semanas, no hay una secuencia seguida	38
5.19. Ya no hay saltos entre semanas	40
5.20. Diagrama que muestra el proceso de RFE [17]	43
5.21. Diagrama de flujo del proceso sin Selección de Variables	44
5.22. Diagrama de flujo del proceso con Selección de Variables	44
5.23. Diagrama de flujo de la segunda etapa del proyecto	45
6.1. Matriz de confusión de SVM	47
6.2. Matriz de confusión de GBT	47
6.3. Matriz de confusión de LR	47
6.4. Matriz de confusión de KNN	47
6.5. Matriz de confusión de NN	48
6.6. Matriz de confusión de SVM después de Selección de Variables	52

6.7. Matriz de confusión de GBT después de Selección de Variables	52
6.8. Matriz de confusión de LR después de Selección de Variables	52
6.9. Matriz de confusión de KNN después de Selección de Variables	52
6.10. Matriz de confusión de NN después de Selección de Variables	52
6.11. Importancia de variables sin aplicar Selección de Variables	53
6.12. Importancia de variables después de aplicar Selección de Variables	54
6.13. Comparación de importancia de variables al no aplicar y aplicar Selección de Variables	55
6.14. Desempeño de las métricas conforme avanzan las semanas con el algoritmo de GBT, sin aplicar Selección de Variables	56
6.15. Matriz de confusión para cada semana utilizando el algoritmo GBT	57
6.16. Desempeño de las métricas conforme avanzan las semanas con el algoritmo de GBT, aplicando Selección de Variables	57
6.17. Matriz de confusión para cada semana utilizando el algoritmo GBT después de aplicar Selección de Variables	58

Lista de cuadros

5.1. Lista de programas y/o librerías y su versión que fue utilizado durante el desarrollo del proyecto.	19
5.2. Variables que se encuentran en el dataset con su tipo de dato y valores posibles . . .	20
5.3. Proporción y cantidad de nulos por variables	21
5.4. Proporción y cantidad de nulos por variables luego de una limpieza inicial	24
6.1. Métricas: primer análisis - 16 semanas - ningún método de Selección de Variables . .	46
6.2. Matriz de confusión: primer análisis - 16 semanas - ningún método de Selección de Variables	47
6.3. Lista de variables que se eliminaron del conjunto de datos junto a su descripción y el método por el que fueron eliminadas	49
6.4. Lista de variables finales con su descripción	51
6.5. Métricas: segundo análisis - 16 semanas - aplicando RFE	51
6.6. Matriz de confusión: segundo análisis - 16 semanas - aplicando RFE	51
6.7. Matriz de confusión: segunda etapa - evaluación por semanas del algoritmo GBT - ningún método Selección de Variables	56
6.8. Matriz de confusión: segunda etapa - evaluación por semanas del algoritmo GBT - aplicando RFE	58

Este proyecto de tesis se centra en la predicción temprana del rendimiento académico en el contexto universitario, específicamente en el curso de Cálculo 1. Se utilizaron técnicas de Educational Data Mining (EDM) y Learning Analytics (LA) para analizar grandes volúmenes de datos educativos y predecir el éxito o fracaso de los estudiantes.

EDM emplea técnicas de minería de datos para datos educativos y tiene como objetivo pronosticar el rendimiento estudiantil, incluyendo el abandono escolar. Por su parte, LA se enfoca en medir, recopilar y analizar datos para mejorar los resultados académicos y entornos educativos.

La predicción temprana es crucial para intervenir a tiempo y brindar apoyo a estudiantes en riesgo, mejorando sus posibilidades de éxito académico. Mediante análisis exploratorios y algoritmos de aprendizaje automático como Regresión Logística y Máquinas de Soporte Vectorial, se buscó identificar las características influyentes en el rendimiento de los estudiantes.

Este proyecto tiene como objetivo mejorar la toma de decisiones en el proceso educativo, proporcionar recomendaciones personalizadas y promover un ambiente de aprendizaje enriquecedor y equitativo. La combinación de EDM y LA presenta un potencial significativo para transformar la manera en que se aborda el rendimiento estudiantil en entornos universitarios.

CAPÍTULO 1

Introducción

En un mundo cada vez más impulsado por la información y la tecnología, la capacidad de predecir y comprender eventos futuros se ha convertido en un recurso invaluable. En este contexto, la predicción del rendimiento académico de los estudiantes se erige como un campo de estudio crucial, con aplicaciones significativas en la mejora de la calidad de la educación y la identificación temprana de desafíos académicos.

El presente trabajo se adentra en el ámbito de la predicción del desempeño de los estudiantes en un curso específico, explorando dos etapas distintas en el proceso de desarrollo de un modelo predictivo confiable. Este proyecto se basa en la premisa de que no sólo es fundamental anticipar quiénes podrían reprobado un curso, sino también hacerlo de manera temprana para permitir la implementación oportuna de intervenciones educativas.

La primera etapa se centra en evaluar el impacto del pre-procesamiento de datos y la selección de características en el rendimiento de modelos de clasificación. Los datos iniciales revelaron una serie de desafíos, como la presencia de valores nulos en más del 40 % de las columnas, lo que requirió un exhaustivo trabajo de limpieza y tratamiento de datos. La selección de características, a través de diversas técnicas, se convirtió en un elemento clave para la mejora del rendimiento de los modelos. Las métricas de evaluación, como precisión, exactitud, sensibilidad y especificidad, se utilizaron para medir el desempeño de los algoritmos en la identificación de estudiantes en riesgo.

En la segunda etapa, se aborda la cuestión de la predicción temprana, explorando cómo anticipar con seguridad el rendimiento académico de los estudiantes en distintos momentos del semestre. Para ello, se utilizó un algoritmo de Gradient Boosting Trees (GBT) y se evaluaron las métricas de rendimiento a lo largo del tiempo, desde la semana 3 hasta la semana 16 del curso. Este análisis permitió determinar cuándo se podía realizar una predicción confiable y cómo la selección de características afectaba la calidad de las predicciones.

En resumen, este trabajo se enfoca en la construcción de modelos predictivos para identificar a los estudiantes en riesgo de reprobado un curso, tanto de manera temprana como a lo largo del semestre. A través de un enfoque metódico que incluye pre-procesamiento de datos, selección de características y evaluación rigurosa de algoritmos, se busca proporcionar a las instituciones educativas herramientas efectivas para intervenir y apoyar a los estudiantes de manera más precisa y oportuna.

El análisis detallado de las etapas y resultados de este proyecto ofrece una visión enriquecedora de las complejidades involucradas en la predicción del rendimiento académico, destacando la im-

portancia de la ciencia de datos en el ámbito educativo y su capacidad para mejorar la toma de decisiones en el campo de la enseñanza y el aprendizaje.

2.1. Objetivo general

Predicción Temprana del Rendimiento Académico en Cursos Universitarios: Un Enfoque en Cálculo 1 con Educational Data Mining y Learning Analytics.

2.2. Objetivos específicos

- Aplicar las mejores prácticas de limpieza de datos, Feature Engineering y Selección de Variables.
- Realizar comparaciones de las métricas obtenidas de distintos algoritmos.
- Obtener a partir de qué semana se puede obtener una predicción confiable.

La predicción del rendimiento académico de los estudiantes es una tarea crucial, pero compleja en la educación [36]. Es vital para empoderar a los estudiantes a tomar el control, promover el aprendizaje autorregulado y permitir a los educadores identificar a los estudiantes en riesgo de fracaso e intervenir oportunamente [8].

Sin embargo, esta tarea supone un reto debido a los numerosos factores que pueden afectar al rendimiento de los estudiantes. Para hacer frente a esto, las técnicas de Educational Data Mining (EDM) y Learning Analytics (LA), como los Learning Management System (LMS) y los cursos masivos abiertos en línea (MOOC), sirven para analizar los grandes volúmenes de datos que reflejan los procesos de aprendizaje de los estudiantes [11]. Además, se pueden recopilar datos sobre los estudiantes en la educación tradicional presencial y en entornos mixtos (B-learning).

La aplicación de técnicas de EDM y LA para analizar datos tan extensos, ha permitido obtener perspectivas valiosas, interpretables y novedosas sobre los alumnos [15]. La EDM, que implica el uso de técnicas de minería de datos (DM) para datos educativos, incluidas las actividades de aprendizaje [5], tiene como objetivo predecir el rendimiento de los estudiantes, así como el fracaso, el éxito o el abandono escolar [40].

Por su parte, la LA se centra en la medición, recopilación, análisis y comunicación de datos sobre los alumnos y sus entornos para optimizar sus resultados [43]. Así pues, EDM y LA son campos estrechamente relacionados que comparten el objetivo común de predecir y guiar el aprendizaje de los estudiantes. La predicción temprana, también conocida como la aplicación de modelos predictivos para identificar a los estudiantes en riesgo de fracaso o abandono lo antes posible, es una tarea crítica en EDM [6] [46]. La detección temprana de los estudiantes en riesgo permite la intervención y el apoyo oportunos para promover el éxito de los estudiantes y prevenir el abandono o el fracaso.

La predicción temprana presenta desafíos en EDM debido a la naturaleza multifactorial del rendimiento de los estudiantes, pero es vital en la educación a través de diferentes etapas e instituciones en todo el mundo. La predicción temprana es esencial para implementar estrategias de prevención eficaces, proporcionar asesoramiento o recomendaciones y llevar a cabo acciones o intervenciones de recuperación para los estudiantes en riesgo [39].

4.1. Aspectos conceptuales

4.1.1. Inteligencia artificial

La Inteligencia artificial (**IA**) se refiere en al campo de la ciencia e ingeniería que se enfoca en crear máquinas o sistemas inteligentes. Involucra desarrollar entidades artificiales que poseen ciertas características comúnmente asociadas con la inteligencia humana. Los sistemas de IA han demostrado la habilidad de poder percibir su entorno, buscar información, reconocer patrones, planear y ejecutar acciones, y adaptarse a nuevas situaciones emergentes [35] [13] [28]. Esta perspectiva resalta las habilidades cognitivas de los sistemas de IA, lo cual le permite analizar y entender información compleja. [35]

Otra perspectiva, tomada de los conceptos de Alan Turing, se refiere a la habilidad que tiene una máquina de IA de comunicarse, razonar y operar de forma independiente, parecido al comportamiento humano. Sin embargo, reconoce que el uso actual del término de IA es comúnmente confundido con conceptos como *Machine Learning* y *Deep Learning*. [13]

En general, IA engloba un gran rango de tecnologías y acercamientos dirigidos a desarrollar máquinas que exhiben un comportamiento inteligente. Involucra el estudio e ingeniería de un comportamiento inteligente en humanos, animales y máquinas, con el fin de crear artefactos, como computadoras y tecnologías similares, capaces de realizar tareas que comúnmente requieren de inteligencia humana. [45]

4.1.2. Machine Learning

Machine Learning (**ML**) es una rama del campo de la Inteligencia Artificial (IA) que se centra en el desarrollo de algoritmos y modelos estadísticos. Permite a los sistemas mejorar automáticamente su rendimiento y aprende de forma automática una tarea específica mediante el análisis de datos. Sus aplicaciones abarcan diversos ámbitos como la salud, finanzas, marketing, toma de decisiones, educación, automovilismo, etc...

Uno de los aspectos más significativos de ML es su aplicabilidad en una gran cantidad de es-

cenarios. En salud, los modelos de ML pueden predecir enfermedades, recomendar tratamientos e incluso ayudar a descubrir nuevos fármacos. En finanzas, se utilizan para la detección de fraudes, el comercio algorítmico, evaluación de riesgos, y detectar deserción de clientes. En marketing, se beneficia de la segmentación de clientes, recomendaciones personalizadas y el análisis de opiniones. La adopción de ML en estos ámbitos permite obtener información basada en datos, reducir costes y mejorar la toma de decisiones.

A pesar de sus prometedoras aplicaciones, el aprendizaje automático presenta diversos retos. La naturaleza de “caja negra” de los modelos de ML plantea problemas de transparencia e interpretabilidad. Además, hay problemas relacionados al sesgo que pueden aplicarse sobre los datos y los modelos pueden perpetuar las desigualdades sociales. Garantizar el uso ético y responsable de las tecnologías de aprendizaje automático es un reto permanente que el sector debe abordar.

Los modelos de ML se nutren de los datos. En la época actual, la generación de millones de bytes por día, ha aportado mucho a sus avances. Con la proliferación de las fuentes de datos, desde sensores a redes sociales, los modelos de aprendizaje automático pueden procesar y analizar inmensos conjuntos de datos. La interacción de los macrodatos y el aprendizaje automático ha dado lugar a grandes avances en el análisis predictivo, la detección de anomalías y el reconocimiento de patrones.

En cuanto a su futuro, es prometedor. En los últimos años ha habido grandes innovaciones como el aprendizaje por transferencia (transfer learning), el aprendizaje por refuerzo (reinforcement learning), y el aprendizaje federado (Federated Learning). Todas estas innovaciones están ampliando los horizontes de la IA. Además, se está aventurando en la computación periférica, permitiendo el procesamiento y la toma de decisiones casi en tiempo real. Es probable que ML siga integrándose en diversos aspectos de nuestras vidas diarias.

- **Aprendizaje por transferencia:** El aprendizaje por transferencia consiste en tomar las características aprendidas en un problema y aprovecharlas en un nuevo problema similar. Por ejemplo, las características de un modelo que ha aprendido a identificar bicicletas pueden ser útiles para poner en marcha un modelo destinado a identificar motocicletas. [14]

El aprendizaje por transferencia suele aplicarse a tareas en las que el conjunto de datos es demasiado pequeño para entrenar un modelo completo desde cero. [14]

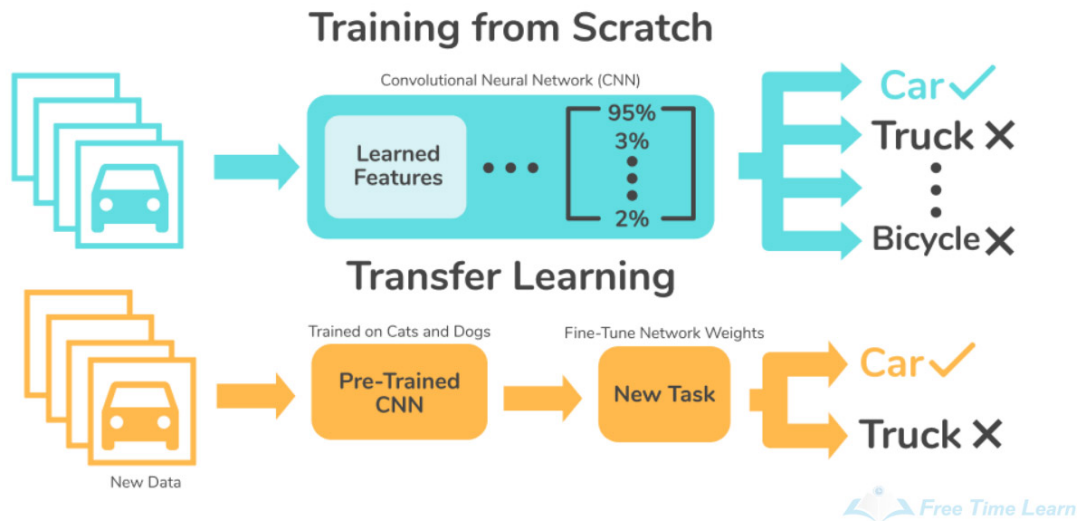


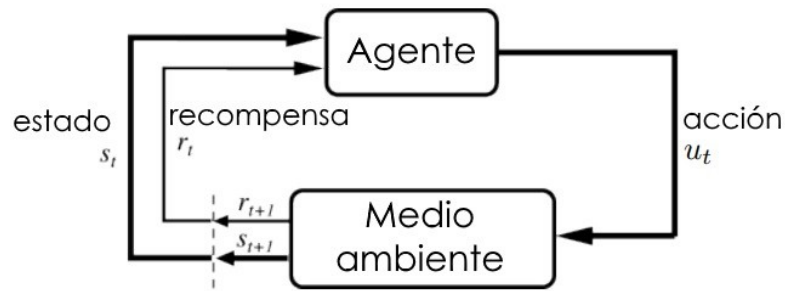
Figura 4.1: Diagrama de qué es aprendizaje por transferencia [2]

- **Aprendizaje por refuerzo:** El aprendizaje por refuerzo (RL por sus siglas en inglés) es

una técnica de aprendizaje de ML que entrena al software a tomar decisiones para lograr los resultados más óptimos. Imita el proceso de aprendizaje por ensayo y error que utilizan los humanos para alcanzar sus objetivos. Las acciones del software que contribuyen al logro del objetivo se ven reforzadas, mientras que las que lo desvían se ignoran [3].

Los algoritmos de RL utilizan un paradigma de recompensa y castigo cuando procesan datos. Aprenden de la retroalimentación de cada acción y autodescubren las mejores vías de procesamiento para alcanzar los resultados finales. Los algoritmos también son capaces de atrasar la gratificación [3].

La RL es un potente método para ayudar a los sistemas de IA a lograr resultados óptimos en entornos invisibles [3].



[Figure source: Sutton & Barto, 1998]

Figura 4.2: Diagrama de qué es aprendizaje por refuerzo [30]

- Aprendizaje Federado:** Aprendizaje Federado (**FL** por sus siglas en inglés) es un enfoque descentralizado propuesto inicialmente por Google para construir modelos de ML utilizando conjuntos de datos distribuidos a través de múltiples dispositivos. El objetivo principal es entrenar modelos estadísticos en dispositivos o centros de datos aislados sin transferir los datos a dispositivos centralizados. Incorpora ideas de múltiples áreas, incluyendo criptografía, ML, computación heterogénea y sistemas distribuidos [7].

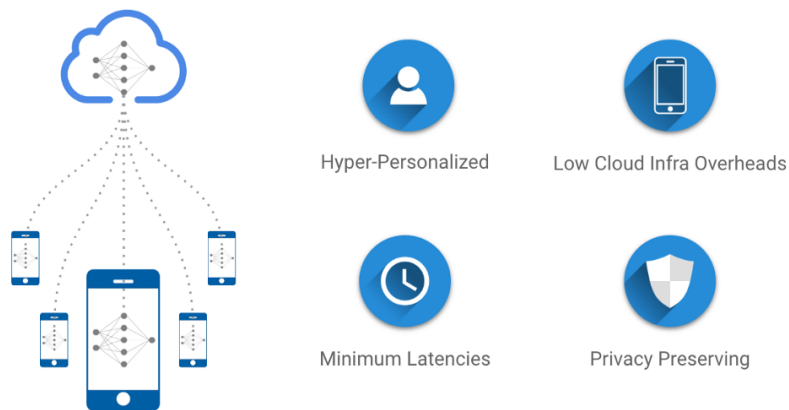


Figura 4.3: Diagrama de qué es aprendizaje federado [44]

4.1.3. Ciencia de datos

Después de buscar muchas definiciones de Ciencia de Datos, VanderPlass en su libro *Python Data Science Handbook* la define de una forma sencilla que engloba el resto de las otras definiciones, a grandes rasgos. La definición dice así:

“It is a suprisingly hard definition to nail down, especially given how ubiquitous the term has become. Vocal critics have variously dismissed the term as a superfluous label or a simple buzzword that only exists to salt résumés and catch the eye of overzealous tech recruiters.

...Ciencia de Datos, despite its hyper laden veneer, is perhaps the best label we have for the cross-disciplinary set of skills that are becoming increasingly important in many applications across industry and academia. The best existing definition of Ciencia de Datos is illustrated by Drew Conway’s Ciencia de Datos Venn Diagram, published on his blog in September 2010.



Figura 4.4: Diagrama de Venn que define qué es Ciencia de Datos [12]

...‘Ciencia de Datos’ is fundamentally an interdisciplinary subject. Ciencia de Datos comprises three distinct and overlapping areas: the skills of a statistician who knows how to model and summarize datasets; the skills of a computer scientist who can design and use algorithms to efficiently store, process and visualize this data; and the domain expertise-what we might think of as “classical” training in a subject-necessary both to formulate the right questions and to put their answers in context...” [27]

4.1.4. Educational Data Mining

Educational Data Mining (EDM) es definida según la página de la comunidad de The Educational Data Mining [31] como una disciplina emergente, que se ocupa de desarrollar métodos para explorar los tipos de datos únicos que proceden del entorno educativo, y de utilizar esos métodos para comprender mejor a los estudiantes y los entornos en los que aprenden. La EDM suele hacer hincapié en la mejora de los modelos de los alumnos, que denotan sus conocimientos, motivación, metacognición y actitudes actuales.

Existen varios tipos de implementación para EDM. Los sistemas pueden ser orientados para diferentes actores con cada uno teniendo un punto de vista particular:

- **Orientado hacia estudiantes:** El objetivo es recomendar a los alumnos actividades, recursos y tareas de aprendizaje que favorezcan y mejoren su aprendizaje, sugerir buenas experiencias de aprendizaje para los alumnos, sugerir acortamientos de caminos o simplemente enlaces a seguir, basándose en las tareas ya realizadas por el alumno y sus éxitos y en las tareas realizadas por otros alumnos similares, etc.. [10]
- **Orientado hacia catedráticos:** El objetivo es obtener una retroalimentación más objetiva para la instrucción, evaluar la estructura del contenido del curso y su eficacia en el proceso de aprendizaje, clasificar a los alumnos en grupos en función de sus necesidades de orientación y seguimiento, encontrar patrones de aprendizaje tanto regulares como irregulares, encontrar los errores cometidos con más frecuencia, encontrar actividades que sean más eficaces, descubrir información para mejorar la adaptación y personalización de los cursos, reestructurar los sitios para personalizar mejor los cursos, organizar los contenidos de forma eficaz según el progreso del alumno y construir de forma adaptativa planes de instrucción, etc.. [10]
- **Orientado hacia responsables académicos y administradores:** El objetivo es disponer de parámetros sobre cómo mejorar la eficiencia del sitio y adaptarlo al comportamiento de sus usuarios, disponer de medidas sobre cómo organizar mejor los recursos institucionales y su oferta educativa, mejorar la oferta de programas educativos y determinar la eficiencia del nuevo enfoque de aprendizaje a distancia mediado por un computador. [10]

4.1.5. Learning Analytics

El concepto Learning Analytics (LA) es comúnmente definido como la forma de medir, recolectar, analizar, y reportar toda la información acerca de estudiantes y sus contextos. Esto se hace con el fin de entender y optimizar el aprendizaje y los ambientes en el que ocurre [37]. Es un campo de práctica e investigación que utiliza métodos computacionales para analizar información de una gran variedad de fuentes, como sistemas de gestión del aprendizaje, herramientas de evaluación, redes sociales y encuestas a los estudiantes [4].

LA puede ser utilizado para:

- **Identificar estudiantes en riesgo:** Puede ser utilizado para identificar estudiantes que tienen un alto riesgo de reprobación o retirar un curso o también, dejar la universidad. Esta información puede ser luego utilizada para proveer apoyo específico y atención al estudiante para ayudar al estudiante a aprobar.
- **Mejorar la enseñanza y el aprendizaje:** Puede ser utilizado para llevar un registro del progreso del estudiante, identificar áreas en las cuales los estudiantes puedan tener dificultades, y darle retroalimentación al catedrático. Esta información puede ser luego utilizada para mejorar el diseño del material de estudio y actividades.
- **Aprendizaje personalizado:** Puede ser utilizado para personalizar la experiencia para cada estudiante. Por ejemplo, los estudiantes que se encuentran con dificultades con un concepto en específico, pueden ser apoyados con actividades o recursos o atención adicional para garantizar el entendimiento.
- **Mejor toma de decisiones:** También puede ser utilizada para asistir en la toma de decisiones sobre la política y la práctica educativas. Por ejemplo, los datos de la analítica del aprendizaje pueden utilizarse para evaluar la eficacia de nuevos métodos de enseñanza o para identificar las áreas a las que deben asignarse recursos.

- **Entender los comportamientos de aprendizaje de los estudiantes:** Puede ser utilizada para entender cómo es que los estudiantes aprenden, incluyendo sus tipos de aprendizaje preferidos, fortalezas y debilidades, y sus motivadores.
- **Mejorar el compromiso de los estudiantes:** Puede ser utilizada para identificar y abordar los factores que contribuyen a la disminución del compromiso de los estudiantes con el paso del tiempo.

[4](#) [37](#) [18](#) [42](#)

4.1.6. Educational Warning System

Los Sistemas de Alerta Educativa (EWS) son sistemas basados en datos que utilizan diversas fuentes de información, como las notas, la asistencia y el comportamiento de los alumnos, para identificar a los estudiantes que corren el riesgo de suspender un curso o abandonar los estudios. Los EWS pueden utilizarse entonces para proporcionar apoyo e intervenciones específicas para ayudar a estos estudiantes a tener éxito [34](#).

Los EWS se suelen implantar a nivel de escuela o de distrito, pero también existen varios productos comerciales de EWS. Algunas de las características más comunes de los sistemas de alerta temprana son las siguientes:

- **Recolección de datos:** Los EWS recopilan datos de una variedad de fuentes, como las calificaciones de los estudiantes, la asistencia, los registros de comportamiento y los resultados de las pruebas estandarizadas.
- **Análisis de datos:** Los EWS utilizan sofisticados algoritmos de minería de datos y aprendizaje automático para analizar los datos recopilados e identificar a los estudiantes que están en riesgo de fracaso.
- **Planificación de la intervención:** EWS se puede utilizar para desarrollar e implementar intervenciones específicas para estudiantes en riesgo. Estas intervenciones pueden incluir apoyo académico, tutoría, asesoramiento u otros servicios.
- **Seguimiento del progreso:** Los EWS pueden utilizarse para seguir el progreso de los estudiantes en situación de riesgo y evaluar la eficacia de las intervenciones que se están proporcionando.

Beneficios de los EWS

- **Identificación temprana:** El EWS puede identificar a los alumnos de riesgo en una fase temprana, antes de que se queden rezagados y sea más difícil ayudarles.
- **Intervenciones específicas:** El EWS puede utilizarse para proporcionar apoyo e intervenciones específicas a los estudiantes en situación de riesgo, en función de sus necesidades individuales.
- **Mejores resultados de los alumnos:** Se ha demostrado que los EWS mejoran los resultados de los estudiantes, como las calificaciones, la asistencia y las tasas de graduación.
- **Eficacia:** Los sistemas de alerta temprana pueden ayudar a las escuelas y distritos a utilizar sus recursos de manera más eficiente, dirigiendo sus intervenciones a los estudiantes que más las necesitan.

- **Equidad:** El EWS puede ayudar a promover la equidad garantizando que todos los estudiantes tengan acceso al apoyo y las intervenciones que necesitan para tener éxito.

16

Desafíos de los EWS

- **Calidad de los datos:** Los EWS se basan en datos precisos y puntuales. Es importante disponer de un sistema de recolección y gestión eficaz de estos datos.
- **Preocupación por la privacidad:** Los sistemas de alerta temprana recogen datos sensibles sobre los estudiantes. Es importante contar con políticas que protejan estos datos y garanticen su uso responsable.
- **Implementación:** La implementación de los sistemas de alerta temprana puede ser compleja y requiere la información del personal.
- **Coste:** La implementación y el mantenimiento de los sistemas de alerta temprana pueden resultar caros.

16

Conclusión

El EWS puede utilizarse de diversas maneras para apoyar a los estudiantes que corren el riesgo de fracasar. Por ejemplo, el EWS puede utilizarse para:

- **Identificar a los estudiantes que necesitan apoyo académico adicional:** EWS se puede utilizar para identificar a los estudiantes que están teniendo dificultades en una clase o materia en particular. A estos estudiantes se les puede proporcionar apoyo adicional, como tutorías o sesiones de ayuda extra.
- **Identificar a los estudiantes que corren el riesgo de abandonar los estudios:** El EWS puede utilizarse para identificar a los estudiantes que corren el riesgo de abandonar los estudios. Estos estudiantes pueden recibir intervenciones específicas para ayudarles a permanecer en la escuela y tener éxito.
- **Supervisar el progreso de los estudiantes que reciben intervenciones:** El EWS puede utilizarse para seguir el progreso de los estudiantes que están recibiendo intervenciones para ayudarles a tener éxito. Esta información puede utilizarse para evaluar la eficacia de las intervenciones y realizar los ajustes necesarios.

38

4.1.7. Feature Engineering

Es el proceso de transformar datos brutos en características que puedan ser utilizadas por modelos de machine learning para realizar predicciones. Es un paso fundamental en el proceso de machine learning, ya que la calidad de las características puede tener un impacto significativo en el rendimiento del modelo [1].

Beneficios

Feature Engineering puede aportar una serie de ventajas, entre ellas:

- **Mejora del rendimiento del modelo:** Al seleccionar y transformar cuidadosamente las características de su conjunto de datos, los profesionales del machine learning pueden mejorar la precisión y fiabilidad de sus modelos.
- **Reducción del sobreajuste:** El sobreajuste se produce cuando un modelo de machine learning aprende demasiado bien los datos de entrenamiento y es incapaz de generalizar a nuevos datos. La aplicación de feature engineering puede ayudar a reducir el sobreajuste creando características más representativas del mundo real.
- **Mayor interpretabilidad:** La interpretabilidad se refiere a la capacidad de entender cómo un modelo de machine learning hace predicciones. La aplicación de feature engineering puede ayudar a aumentar la interpretabilidad de los modelos creando características más significativas para los humanos.

33

Técnicas

Existen diversas técnicas de feature engineering que pueden utilizarse, dependiendo del conjunto de datos específico y de la tarea de machine learning. Algunas técnicas comunes son:

- **Limpieza y preprocesamiento de datos:** Consiste en eliminar el ruido y las incoherencias de los datos y convertirlos a un formato que pueda utilizar el modelo de machine learning. Por ejemplo, puede consistir en eliminar filas duplicadas, rellenar valores faltantes y convertir datos categóricos en datos numéricos.
- **Selección de Variables:** Consiste en seleccionar las características más relevantes e informativas del conjunto de datos. Para ello pueden utilizarse diversos métodos, como pruebas estadísticas, algoritmos de machine learning o el conocimiento expertos en el negocio.
- **Feature transformation:** Consiste en transformar las características para crear otras más útiles para la tarea de machine learning. Por ejemplo, puede consistir en escalar las características a un rango común o crear nuevas características que representen la relación entre diferentes características.
- **Feature creation:** Se trata de crear nuevas características a partir de las existentes. Por ejemplo, se pueden crear características que representen las relaciones temporales o espaciales entre puntos de datos, o características que representen las interacciones entre diferentes características.

9

Feature Engineering es una potente herramienta que puede utilizarse para mejorar el rendimiento de los modelos de machine learning. Seleccionando y transformando cuidadosamente las características de su conjunto de datos, los profesionales de machine learning pueden construir modelos más precisos, fiables e interpretables.

4.1.8. Selección de variables

Una “feature” es una propiedad individual medible del proceso observado. A partir de un conjunto de características, cualquier algoritmo de machine learning puede realizar una clasificación [19].

Selección de Variables es un método que consiste en reducir la cantidad de variables que nosotros le introducimos al modelo. Esto lo hacemos al usar únicamente las variables relevantes y eliminar esas variables que están introduciendo ruido y/o complejidad al modelo [29].

Existen dos tipos principales de modelos de Selección de Variables los que pueden ser aplicados en modelos supervisados y los no supervisados. Debido a que en este proyecto solo se aplicarán los de aprendizaje supervisados, únicamente se definen y clasifican estos.

Los modelos supervisados de Selección de Variables se refiere a a los métodos en los cuáles necesita la clase a la que pertenece para poder realizar el Selección de Variables. Utilizan luego las variables a evaluar para identificar qué variables son las que pueden incrementar o reducir la eficiencia del modelo. El resultado final es un conjunto de variables óptimo para el desempeño del modelo. Las subdivisiones del modelo supervisado son las siguientes:

Filter methods:

Los filter methods para selección de variables son una clase de métodos de machine learning supervisado que utilizan medidas estadísticas para evaluar la relevancia de las características para la variable objetivo. Funcionan evaluando independientemente cada característica y clasificándola en función de su puntuación. Luego, un grupo de las variables con mejor puntuación es seleccionado [19].

Los filter methods suelen ser rápidos y poco costos computacionalmente. Esto los hace adecuados para conjuntos de datos con un gran número de variables. También son fáciles de aplicar y pueden utilizarse con diversos algoritmos de machine learning [22].

Sin embargo, los filter methods tienen algunas limitaciones. Por ejemplo, no tienen en cuenta las interacciones entre variables y pueden ser sensibles a la elección de la medida estadística. Además, puede que no sean capaces de identificar todas las variables relevantes en un conjunto de datos [24].

A pesar de estas limitaciones, los métodos de filtrado son una herramienta potente y versátil para la selección de características. Algunos ejemplos de filter methods son:

- Básicos
 - Constantes
 - Cuasi-constantes
 - Duplicados
- Medidas Estadísticas
 - Puntuación Fisher
 - Métodos univariados
 - Información mutua
- Correlación

Wrapper methods:

Son un tipo de método supervisado de Selección de Variables que utiliza algoritmos de machine learning para evaluar el desempeño de un grupo de variables. Funcionan al añadir o remover de forma iterativa variables que pertenecen a un grupo de variables y luego medir el desempeño del modelo de machine learning. El resultado de este proceso, es un grupo de variables con el cual obtenemos el mejor desempeño del modelo [19].

A comparación de los filter methods, estos son más caros computacionalmente, pero pueden llegar a ser más efectivos al identificar las variables más relevantes, especialmente en un conjunto de datos en el cual hay interacciones complejas entre variables [21].

Al utilizar los wrapper methods hay que tener en cuenta, el resultado es el grupo de variables óptimo para un algoritmo o familia de algoritmos en específico. Por ejemplo, para un algoritmo que se base en árboles, el mismo grupo de variables puede servir para random-forests y gradient boosted trees, pero lo más probable es que no funcione con una Regresión Logística [26].

En general, los wrapper methods son una potente herramienta para Selección de Variables. Son especialmente adecuados para conjuntos de datos donde las interacciones entre variables son importantes y se dispone de recursos informáticos. Algunos ejemplos de Wrapper Methods son:

- Step forward selection
- Step backward selection
- Exhaustive search
- Feature shuffling

Embedded methods:

Este método integra el proceso de Selección de Variables dentro del entrenamiento de un modelo de machine learning. Esto significa que el modelo aprende a seleccionar las variables más importantes como parte de su proceso de entrenamiento. Esto funciona gracias a que utilizan una variedad de técnicas para penalizar o regularizar los pesos en las variables del modelo. Esta penalización o regularización fuerza al modelo a aprender y a depender de un grupo más pequeño de variables, lo cual puede conducir a una mejora en el desempeño y una reducción en sobreajuste del modelo.

Parecido a los wrapper methods, hay que tener en cuenta la interacción entre el modelo y las variables. Es decir, las variables seleccionadas para un algoritmo de la familia de los árboles probablemente no sean las mismas para un algoritmo de tipo de regresión. La diferencia es que son más baratos computacionalmente hablando. Esto se debe a que solamente se hace un fit del modelo, en lugar de varias iteraciones.

Cuando se compara este método con los dos anteriores, los embedded methods son más rápidos que los wrapper, más precisos que los filter, sí detecta la interacción entre variables, y encuentra el grupo óptimo de variables para el algoritmo que se está entrenando. Algunos ejemplos de embedded methods son:

- LASSO
- Decision tree derived importance
- Regression coefficients
- Recursive feature elimination

- Recursive feature addition

4.1.9. Métricas de rendimiento

El uso de métricas de rendimiento como significado en machine learning ha sido muy común. También se puede obtener de fuentes secundarias como libros de texto, estudios de investigación y tutoriales web.

Esta definición particular incluye conceptos procedentes de otras obras. Sin embargo, una fuente especialmente relevante es el libro “Machine Learning: A probabilistic perspective, escrito por Kevin Murphy”. Murphy ha descrito en el libro las métricas de rendimiento como medidas de la calidad de un modelo de aprendizaje automático. Se explican diferentes tipos de métricas de rendimiento y cómo pueden utilizarse para evaluar el rendimiento de los modelos de aprendizaje automático [32].

Aparte de los libros de texto, machine learning también se menciona en varios artículos de investigación. Por ejemplo, hay un artículo titulado “A Survey of Performance Metrics for Classification”, de David J. Hand, que resume varios tipos de métricas de rendimiento utilizadas en tareas de clasificación [25].

Métricas de clasificación

El tema de los problemas de clasificación es quizá el más investigado en la actualidad en cuanto a machine learning se refiere. Casi todos los entornos industriales y de producción tienen sus casos de uso. Reconocimiento de voz; reconocimiento facial; clasificación de textos, etc.

Dado que los modelos de clasificación se caracterizan por sus distintos resultados, necesitamos una métrica que empareje de algún modo las distintas clases. Las métricas de clasificación juzgan el rendimiento de un modelo y permiten saber si la clasificación es buena o mala a distintos niveles.

Existen varias métricas para evaluar los modelos de clasificación, estas son:

- **Matriz de confusión:** La matriz de confusión es una métrica donde el output puede ser de dos o más clases. Es una tabla que tiene cuatro diferentes tipos de combinaciones de los valores predichos y actuales [41].

		Valor predicho	
		Aprobó	Reprobó
Valor real	Aprobó	VP	FN
	Reprobó	FP	VN

Cada valor de la matriz de confusión significa lo siguiente:

- **VP (Verdadero Positivo):** El número de estudiantes que fueron asignados como aprobó y sí aprobaron el curso.
- **VN (Verdadero Negativo):** El número de estudiantes que fueron asignados como reprobó y sí reprobaron el curso.
- **FP (Falso Positivo):** El número de estudiantes que fueron asignados como aprobó y reprobaron el curso.
- **FN (Falso Negativo):** El número de estudiantes que fueron asignados como reprobó y aprobaron el curso.

- **Exactitud:** Es la proporción de sujetos correctamente clasificados sobre el total de sujetos, se calcula mediante la siguiente fórmula:

$$Accuracy = \frac{VP + VN}{VP + VN + FP + FN}$$

Los resultados pueden ser erróneos si las características de las clases se distribuyen de forma desigual. Por lo tanto, no puede actuar como una métrica de rendimiento independiente para el modelo [23].

- **Precisión:** La precisión intenta contestar al pregunta *¿Qué proporción de los que fueron identificados como positivos fueron correctos?* [20]. Se calcula con la siguiente fórmula:

$$Precision = \frac{VP}{VP + FP}$$

- **Sensibilidad:** Es la probabilidad de que el modelo clasifique a los estudiantes como aprobados de forma correcta. También, es llamada Recall [23]. Se calcula con la fórmula siguiente:

$$Sensitivity = \frac{VP}{VP + FN}$$

- **Especificidad:** Es la probabilidad de que el modelo clasifique a los estudiantes como reprobados de forma correcta [23]. Se calcula con la fórmula siguiente:

$$Specificity = \frac{VN}{VN + FP}$$

- **Medida F:** El uso únicamente de las medidas sensibilidad y especificidad puede generar sesgo al evaluar los diferentes modelos. La medida F, que es el promedio armónico de sensibilidad y especificidad, utiliza las dos métricas juntas [23].

$$F1 = \frac{2 * VP}{(2 * VP) + FP + FN}$$

4.2. Caso de éxito

4.2.1. Usando Learning Analytics para desarrollar un sistema de alerta temprana

Este paper [23], en el cual se basó este trabajo de graduación, se dividía en dos etapas. Al igual que este trabajo de graduación, primero se buscaba construir un modelo que con toda la data pudiera predecir de forma certera, qué estudiantes iban a reprobado el curso. Como segunda etapa se buscó determinar qué tan temprano se podía obtener una predicción que fuera lo suficientemente precisa para poder apoyar a los estudiantes.

Los datos que utilizaron para poder realizar este análisis fueron extraídos de un ambiente de aprendizaje virtual. Los datos consistían en 3,803 sesiones, 119,921 páginas visitadas, 4,566 mensajes escritos, 62 preguntas en los foros de discusión, 297 respuestas, 8,601 evaluaciones, y 3,937 etiquetas utilizadas para etiquetar los mensajes escritos.

A lo largo del artículo, se trata de forma extensa el pre-procesamiento, la extracción de nuevas variables y el proceso de Selección de Variables. Se menciona que el pre-procesamiento tiene una

notable importancia en los estudios de minería de datos. Es posible generar modelos mejores y más entendibles a través de las acciones tomadas durante esta etapa del proyecto.

Para el análisis de los datos, diferentes modelos de clasificación fueron probados, aplicando procesos de Selección de Variables ya que no siempre el grupo óptimo de variables que fue seleccionado con un modelo funciona con otros. En este caso se probaron con 7 diferentes tipos de modelos diferentes.

Como parte del entrenamiento, se analizó el efecto de Selección de Variables en todos los modelos, se utilizaron diferentes métricas como (precisión, sensitivity, specificity, y f-measure) para hacer la comparación entre todos los modelos. También, cabe mencionar, que este entrenamiento se hizo utilizando la validación de cross-fold.

En la evaluación de resultados, se observó que los modelos que utilizaban información categórica se desempeñaba mejor que los modelos que utilizaban solamente data continua. También, los resultados indicaron que los modelos, luego de pasar por el proceso de Selección de Variables se desempeñaban mejor que los modelos que tenían todas las variables. Finalmente se menciona que la selección de variables (Selección de Variables) no sólo ayuda al modelo, sino que también le ayuda al científico realizando el análisis a interpretarlo más fácil.

En la primera fase, que correspondía a obtener una predicción para estudiantes que puedan reprobar el curso utilizando toda la información, lograron construir un modelo con 83 % de precisión utilizando Redes Neuronales. Luego, para la segunda etapa lograron obtener resultados de 60 % de precisión desde la semana 3 del curso.

Por último, mencionan que el modelo desarrollado ayudará a los catedráticos a identificar a estudiantes con altas probabilidades de reprobar el curso y con esa información, pueden aplicar las medidas necesarias para prevenirlo. Esto ayudará a reducir las tasas de alumnos reprobados [23].

5.1. Descripción general

Este proyecto busca predecir si un estudiante va a reprobar el curso de Cálculo 1. El objetivo es notificar al estudiante con el suficiente tiempo para que brinde la ayuda necesaria al estudiante para lograr aprobar el curso.

El Proyecto fue dividido en dos etapas. La primera etapa consistió en obtener los datos necesarios, hacer la exploración y limpieza a los datos, aplicar técnicas de pre-procesamiento, escoger las variables a utilizar y determinar, y determinar cuál es el algoritmo predictivo que arroja los mejores resultados.

La segunda fase, consistió en determinar qué tan temprano se puede alertar al estudiante que, de acuerdo con el modelo, tiene una alta probabilidad de reprobar el curso.

Durante el proyecto, se respondieron las siguientes preguntas:

1. Cuando se comparan los modelos de clasificación formados a partir de todos los datos (16 semanas) con respecto a las métricas de rendimiento, ¿qué algoritmo/algoritmos de clasificación y técnicas de pre-procesamiento son mejores para predecir los alumnos que no aprueban?
 - ¿Cuál es el impacto de las distintas técnicas de transformación de datos utilizadas en la fase de pre-procesamiento sobre el rendimiento de la clasificación?
 - ¿Cómo influyen en la clasificación las distintas técnicas de “selección de características” utilizadas en la fase de pre-procesamiento?
 - ¿Cómo es el rendimiento de los modelos de clasificación establecidos utilizando un número reducido de características en comparación con otros modelos de clasificación formados utilizando todas las características?
 - ¿Qué características tienen más importancia en la predicción del rendimiento de los alumnos?
2. ¿Cuál es la precisión de los modelos de clasificación establecidos a partir de los datos obtenidos de las semanas 3, 5, 7, 10, 12, 14, y 16 utilizando el algoritmo y las técnicas de pre-procesamiento seleccionados en el ámbito de la primera pregunta de investigación para predecir el rendimiento académico de los estudiantes al final del trimestre?

5.2. Herramientas utilizadas

El proyecto fue trabajado utilizando el lenguaje de programación Python. Como herramientas o librerías que ayudaron con la manipulación al dataset, están Pandas, Spark, Jupyter Notebook y Numpy. Para poder realizar gráficas y entender de una mejor manera el dataset, se utilizaron seaborn y matplotlib. A continuación, una tabla con todos los paquetes utilizados junto a la versión:

Programa/Librería	Version
Python	3.9.10
feature-engine	1.6.1
matplotlib	3.7.2
numpy	1.25.1
pandas	2.0.3
pyspark	3.4.1
scikit-learn	1.3.0
seaborn	0.12.2

Tabla 5.1: Lista de programas y/o librerías y su versión que fue utilizado durante el desarrollo del proyecto.

El dataset con el que se trabajó contiene información de estudiantes que han tomado el curso de Cálculo 1 entre los años 2019 hasta 2022. Estos datos fueron obtenidos por medio del API del portal Canvas.

5.3. Primera etapa

5.3.1. Pre-procesamiento

El dataset cuenta con 155,158 registros y 17 columnas:

Variable	Descripción	Tipo	Valores posibles
Curso	Curso que se está analizando	Categorica	CALCULO 1, CÁLCULO 1, LABORATORIO DE CALCULO 1
Semestre	Semestre en el que se cursó el curso	Discreta	1 y 2
Seccion	Sección en la que el estudiante estaba asignado	Discreta	-
Anio	Año que el estudiante tomó el curso	Discreta	2019-2022
Actividad	Tipo de actividad que se realizó	Categorica	-
TipoCalificacion	Tipo de calificación que tiene esa actividad	Categorica	points, percent, pass_fail, not_graded
PuntosPosibles	El puntaje total que tenía esa actividad	Continua	-
RevisionPares	Indicador si la actividad era en grupos o individual	Categorica	True, False

FechaTodoElDia	Fecha de la entrega de la actividad	Discreta	-
FechaVencimiento	Fecha y hora máxima de la entrega de la tarea	Continua	-
TipoEntrega	Forma de entrega de la tarea	Catagórica	discussion_topic, online_text_entry, online_url, media_recording, online_upload, external_tool, on_paper, not_graded
Nota	La nota o el puntaje que se obtuvo de esa actividad	Continua	-
FechaCalificacion	Fecha y hora en la que la actividad fue calificada	Continua	-
TipoEnvio	Cómo fue entregada la tarea en la plataforma	Catagórica	online_quiz, discussion_topic, online_url, basic_lti_launch, online_upload, media_recording, online_text_entry
NombreEst	Nombre del estudiante asignado al curso	Catagórica	-
CantComentarios	Cantidad de comentarios que tiene esa actividad en la plataforma	Discreta	-
Est_ID	Identificador único del estudiante.	Discreta	-

Tabla 5.2: Variables que se encuentran en el dataset con su tipo de dato y valores posibles

Exploración y limpieza

Luego de un primer análisis exploratorio y lectura del del dataset, se identificaron varias columnas con valores faltantes. La cantidad de valores faltantes por columna es la siguiente:

Como se puede observar en la Tabla 5.2, hay varias columnas que tienen valores nulos. Hay algunas que tienen una gran proporción de valores nulos, entre ellas podemos identificar a *TipoEntrega*, *Nota*, *FechaCalificacion*, y *TipoEnvio*.

Limpieza general

Inicialmente se identificaron diferentes valores en la columna Curso. Como se puede observar en la imagen, existen tres valores distintos. De los tres, se puede asumir que **CÁLCULO 1** y **CALCULO 1** son los mismos solo escritos de forma distinta. Ahora, con el tercero, **LABORATORIO DE CALCULO 1** causó duda por dos razones. Una es porque no es parecido al nombre de los primeros dos y segundo, porque solamente tiene 32 estudiantes asignados. Cuando se entró más a detalle, parece ser que el curso es solamente dado durante el primer semestre de los años 2020, 2021, y 2022. La cantidad de estudiantes con respecto al total del dataset, representaba solamente el 1.30%. Debido a esto, se decidió eliminar del dataset todas las filas con el valor **LABORATORIO DE CALCULO 1** en la columna Curso.

Variable	Cantidad de nulos	Proporción de nulos
Curso	0	0%
Semestre	0	0%
Seccion	0	0%
Anio	0	0%
Actividad	0	0%
TipoCalificacion	0	0%
PuntosPosibles	220	0.14%
RevisionPares	0	0%
FechaTodoELDia	20,274	13.07%
FechaVencimiento	19,365	12.48%
TipoEntrega	33,094	21.33%
Nota	78,935	50.87%
FechaCalificacion	78,844	50.82%
TipoEnvio	105,176	68.79%
NombreEst	0	0%
CantComentarios	0	0%
Est_ID	0	0%

Tabla 5.3: Proporción y cantidad de nulos por variables

```

+-----+-----+
|Curso      |count(Est_ID)|
+-----+-----+
|CALCULO 1  |1333          |
|LABORATORIO DE CALCULO 1|32           |
|CÁLCULO 1  |1105          |
+-----+-----+

```

Figura 5.1: Cursos y cantidad de estudiantes asignados en el dataset

Durante esta limpieza inicial, se identificó la existencia de dos valores en el campo NombreEst que no correspondían a nombres reales de estudiantes. Estos valores son **Estudiante de prueba** y **Alumno de prueba**, por lo que se decidió eliminar todas las filas que presentaran estos valores en dicho campo.

Siguiendo con la limpieza, se crearon dos nuevas variables. La primera permitió identificar las actividades para las cuales ningún alumno de una misma sección, semestre y año tenía calificación. Es decir, el 100% de los valores de la nota para una misma sección, semestre, y año era nulos. Se decidió eliminar del análisis las actividades identificadas de esta manera, ya que no aportaban información a los modelos predictivos. En las imágenes a continuación, se pueden observar algunos casos:

Semestre	Anio	Seccion	Actividad	Nota	PuntosPosibles	Est_ID	proportion_nulls
2	2022	180	Examen Parcial 4	null	12.0	102	1.0
2	2022	180	Examen Parcial 4	null	12.0	1469	1.0
2	2022	180	Examen Parcial 4	null	12.0	1614	1.0
2	2022	180	Examen Parcial 4	null	12.0	1622	1.0
2	2022	180	Examen Parcial 4	null	12.0	1676	1.0
2	2022	180	Examen Parcial 4	null	12.0	1681	1.0
2	2022	180	Examen Parcial 4	null	12.0	1712	1.0
2	2022	180	Examen Parcial 4	null	12.0	1859	1.0
2	2022	180	Examen Parcial 4	null	12.0	1907	1.0
2	2022	180	Examen Parcial 4	null	12.0	1915	1.0
2	2022	180	Examen Parcial 4	null	12.0	192	1.0
2	2022	180	Examen Parcial 4	null	12.0	1951	1.0
...							
2	2022	180	Examen Parcial 4	null	12.0	950	1.0
2	2022	180	Examen Parcial 4	null	12.0	982	1.0

Figura 5.2: Actividad en las cuales la proporción de nulos es 100 %

Semestre	Anio	Seccion	Actividad	Nota	PuntosPosibles	Est_ID	proportion_nulls
2	2022	180	Parcial 4	76%	12.0	102	0.2222222222222222
2	2022	180	Parcial 4	96%	12.0	1614	0.2222222222222222
2	2022	180	Parcial 4	50%	12.0	1622	0.2222222222222222
2	2022	180	Parcial 4	53%	12.0	1676	0.2222222222222222
2	2022	180	Parcial 4	54%	12.0	1681	0.2222222222222222
2	2022	180	Parcial 4	82%	12.0	1712	0.2222222222222222
2	2022	180	Parcial 4	28%	12.0	1859	0.2222222222222222
2	2022	180	Parcial 4	82%	12.0	1907	0.2222222222222222
2	2022	180	Parcial 4	null	12.0	1915	0.2222222222222222
2	2022	180	Parcial 4	null	12.0	2034	0.2222222222222222
2	2022	180	Parcial 4	62%	12.0	2102	0.2222222222222222
2	2022	180	Parcial 4	0%	12.0	2302	0.2222222222222222
...							
2	2022	180	Parcial 4	52%	12.0	950	0.2222222222222222
2	2022	180	Parcial 4	75%	12.0	982	0.2222222222222222

Figura 5.3: Una actividad en la que la proporción de nulos es menor al 100 %

Semestre	Anio	Seccion	Actividad	Nota	PuntosPosibles	Est_ID	proportion_nulls
2	2022	180	Examen Parcial 4	null	12.0	102	1.0
2	2022	180	Parcial 4	76%	12.0	102	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1469	1.0
2	2022	180	Examen Parcial 4	null	12.0	1614	1.0
2	2022	180	Parcial 4	96%	12.0	1614	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1622	1.0
2	2022	180	Parcial 4	50%	12.0	1622	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1676	1.0
2	2022	180	Parcial 4	53%	12.0	1676	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1681	1.0
2	2022	180	Parcial 4	54%	12.0	1681	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1712	1.0
2	2022	180	Parcial 4	82%	12.0	1712	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1859	1.0
2	2022	180	Parcial 4	28%	12.0	1859	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1907	1.0
2	2022	180	Parcial 4	82%	12.0	1907	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	1915	1.0
2	2022	180	Parcial 4	null	12.0	1915	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	192	1.0
2	2022	180	Examen Parcial 4	null	12.0	1951	1.0
2	2022	180	Examen Parcial 4	null	12.0	2034	1.0
2	2022	180	Parcial 4	null	12.0	2034	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	2062	1.0
2	2022	180	Examen Parcial 4	null	12.0	2102	1.0
2	2022	180	Parcial 4	62%	12.0	2102	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	2302	1.0
2	2022	180	Parcial 4	0%	12.0	2302	0.2222222222222222
2	2022	180	Examen Parcial 4	null	12.0	2332	1.0
2	2022	180	Examen Parcial 4	null	12.0	265	1.0

La segunda variable creada fue la proporción de nulos pero esta vez, en un estudiante. Esta variable se utilizó para identificar estudiantes cuyas notas son nulas para el 100% de actividades durante el semestre. Estos registros no aportan información al modelo e incluso podrían generar ruido si se incluyen. Por lo anterior, se construyó esta variable y se eliminaron todos los registros de aquellos estudiantes cuya proporción de nulos para la misma sección, año, y semestre era del 100%. A continuación una demostración de lo observado:

Semestre	Año	Seccion	Actividad	Fecha	TodoEIDia	Nota	Est_ID	proportion_nulls_p_student
1	2019	10	Final	2019-06-01	null	1653	1.0	
1	2019	10	Parcial 1	2019-02-02	null	1653	1.0	
1	2019	10	Parcial 2	2019-02-23	null	1653	1.0	
1	2019	10	Parcial 3	2019-03-23	null	1653	1.0	
1	2019	10	Parcial 4	2019-05-04	null	1653	1.0	
1	2019	10	Portafolio (entrega 1)	2019-02-01	null	1653	1.0	
1	2019	10	Portafolio (entrega 2)	2019-02-26	null	1653	1.0	
1	2019	10	Portafolio (entrega 3)	2019-03-26	null	1653	1.0	
1	2019	10	Portafolio (entrega 4)	2019-05-08	null	1653	1.0	
1	2019	10	Simulacro 1	2019-01-26	null	1653	1.0	
1	2019	10	Simulacro 2	2019-02-16	null	1653	1.0	
1	2019	10	Simulacro 3	2019-03-16	null	1653	1.0	
1	2019	10	Simulacro 4	2019-04-27	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 1	2019-01-12	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 10	2019-05-18	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 2	2019-01-19	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 3	2019-02-09	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 4	2019-03-02	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 5	2019-03-09	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 6	2019-03-30	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 7	2019-04-06	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 8	2019-04-13	null	1653	1.0	
1	2019	10	Verificación del aprendizaje 9	2019-05-11	null	1653	1.0	

Figura 5.5: Estudiante cuya nota en todas las actividades es en nulo

Finalmente, se identificaron algunas actividades las cuales parecían no pertenecer al semestre que se estaba cursando. Había algunas actividades que, por ejemplo, el estudiante estaba cursando en el primer semestre del 2020 y habían actividades con fecha del segundo semestre de 2019. Esto se puede suponer que es causado cuando el catedrático copia un curso de un semestre anterior al nuevo y se copia con las fechas del anterior. Luego, ya al cambiarlas, es que aparece la nueva actividad con la fecha en el semestre correcto. Se decidió eliminar los registros de las actividades con fechas que corresponden a semestres anteriores al que se está cursando. En la siguiente imagen se muestra un ejemplo.

Semestre	Año	Seccion	Actividad	Fecha	TodoEIDia	Est_ID
1	2020	10	Ejercitación 1	2020-01-18		2176
1	2020	10	Ejercitación 1	2019-07-09		2176
1	2020	10	Ejercitación 10	2020-04-17		2176
1	2020	10	Ejercitación 10	2019-10-05		2176
1	2020	10	Ejercitación 11	2020-04-24		2176
1	2020	10	Ejercitación 11	2019-10-12		2176
1	2020	10	Ejercitación 12	2020-05-05		2176
1	2020	10	Ejercitación 12	2019-10-19		2176
1	2020	10	Ejercitación 13	2020-05-16		2176
1	2020	10	Ejercitación 13	2019-11-05		2176
1	2020	10	Ejercitación 14	2020-05-23		2176
1	2020	10	Ejercitación 14	2019-11-09		2176
...						
1	2020	10	Verificación 8	2020-04-27		2176
1	2020	10	Verificación 9	2020-05-15		2176

Figura 5.6: Misma actividad pero con diferentes fechas en un mismo estudiante

Al aplicar los 4 filtros mencionados anteriormente, la cantidad de registros en el dataset disminuyó a 79,797 registros, lo que equivale a una reducción del 48.57% de los registros iniciales. A pesar de disminuir casi un 50% el dataset, la cantidad de estudiantes disminuyó de 2,407 (dataset inicial) a 2,392, una reducción del 0.62%. Luego de esta limpieza inicial, la cantidad y proporción de nulos disminuyeron a lo siguiente:

Variable	Cantidad de nulos	% de nulos actual	% de nulos anterior
PuntosPosibles	39	0.05 %	0.14 %
FechaTodoElDia	4,618	5.79 %	13.07 %
FechaVencimiento	3,770	4.72 %	12.48 %
TipoEntrega	4,164	5.22 %	21.33 %
Nota	4,230	5.30 %	50.87 %
FechaCalificacion	4,216	5.15 %	50.82 %
TipoEnvio	30,430	38.13 %	68.79 %

Tabla 5.4: Proporción y cantidad de nulos por variables luego de una limpieza inicial

A pesar de que se lograron disminuir los valores faltantes de forma considerable solo una con una limpieza inicial, todavía habían varios nulos que eran necesarios tratarlos o filtrarlos para poder dar por terminada la fase de Limpieza de datos. Para lograr tratarlos, fueron trabajadas las variables de forma separada.

Puntos posibles

La columna de *PuntosPosibles* es una de las columnas más importantes en este dataset, ya que nos indicaba cuántos puntos netos vale cada actividad. Esto es de vital importancia porque hay muchas notas que no están como puntos netos, sino que están en una escala de porcentajes y para lograr pasarlo a una escala de puntos netos, fue necesaria esta columna. Dicho esto, fue necesario una buena limpieza a los nulos y también, que se corrigiera cualquier error o valor anormal.

Se partió con 39 nulos en esta columna. Primero, se hizo un filtro para lograr obtener todas las actividades que tenían nulo como valor. El resultado fue el siguiente:

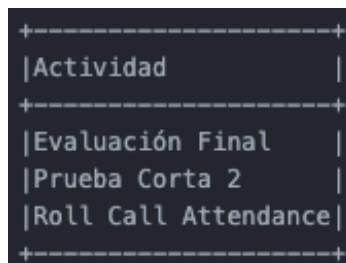


Figura 5.7: Actividades con puntos posibles nulo

Para quitarle el valor nulo y asignarle una nota a cada una de estas actividades, se hicieron dos cosas. Lo primero fue específicamente para la actividad **Roll Call Attendance**. Esta actividad representa la asistencia que un alumno tiene en un curso, siempre y cuando, la asistencia sea tomada por el catedrático. En la mayoría de los casos, la asistencia tenía en la columna de *TipoCalificacion* **not_graded**, por lo tanto, se le asignó un valor de 0. Lo segundo que se hizo para eliminar los nulos de esta columna fue asignarle el valor de la moda de la agrupación al nivel de *Semestre*, *Anio*, *Seccion* y *Actividad*. Con estos dos cambios, logramos dejar en cero los valores nulos en esta columna.

Ahora, una vez con los datos nulos en cero, se procedió a corregir cualquier valor que no hiciera sentido del todo. En la limpieza, se identificaron varias actividades que en lugar de tener los puntos posibles con un valor en la escala de puntos netos, lo tenían, pero en la escala de porcentaje. Para corregir esto, se identificaron cuáles eran estas actividades y se filtró todo el dataset por cada actividad. Esto con el fin de identificar cuál era el valor en puntos posibles que tenía la actividad en otras secciones del mismo año. Con esa solución se logró disminuir en gran cantidad estas actividades cuyo puntos posibles era 100, pero no se arreglaron todas, ya que por alguna razón habían ciertas actividades que solo se hacían en una sección en específico.

Para solucionar estas últimas actividades, ya que no se querían eliminar del dataset, lo que se realizó fue filtrar por año, sección y semestre y obtener el puntaje acumulado que tenían las columnas. Una vez ya se tenía el puntaje acumulado, se le restó 100 (la nota que tiene que tener un curso) y ese valor que restaba fue dividido entre las actividades que estaban con 100 puntos. Ya teniendo el residuo, fue asignado para quitarles el valor de 100 en puntos posibles y dejarlo todo en la escala de puntos netos.

Semestre	Anio	Seccion	Actividad	PuntosPosibles	ptos_posibles
1	2021	10	Evaluación Final	100.0	15.0
1	2021	10	Examen Corto No. 1	100.0	5.0
1	2021	10	Examen Final	100.0	15.0
1	2021	10	Examen corto 2	100.0	5.0
1	2021	10	Examen corto 3	100.0	5.0
1	2021	10	Examen corto 4	100.0	5.0
1	2021	10	Examen corto Implícita	100.0	5.0
1	2021	10	Examen corto Rectas Tangentes	20.0	20.0
1	2021	10	Examen corto razones relacionadas	100.0	5.0
1	2021	10	Gráficas de funciones polinomiales y racionales	100.0	4.375
1	2021	10	Hoja de trabajo gráfica seno y coseno	100.0	4.375
1	2021	10	Infografía Razones de Cambio	100.0	4.375
1	2021	10	Introducción a Límites	100.0	4.375
1	2021	10	Introducción al tema de funciones	100.0	4.375
1	2021	10	Perímetro cuadrado y círculo	100.0	4.375
1	2021	10	Problemas de ecuaciones	100.0	4.375
1	2021	10	Roll Call Attendance	100.0	0.0
1	2021	10	Video Tutorial	100.0	4.375

Figura 5.8: Actividades mostrando la diferencia de los valores puros vs los valores luego de la limpieza

Fecha de calificación

La columna *FechaCalificacion*, a comparación de *PuntosPosibles*, fue mucho más sencilla. La cantidad de nulos en un inicio era de 4,216. Viendo de dónde podían venir la cantidad de nulos en la *FechaCalificación*, se intentó encontrar alguna relación con la *Nota*. Resultó que siempre que la *FechaCalificación* es nulo, también lo es la nota, lo cual puede indicar que esta actividad no se hizo y no tiene nota ni *FechaCalificación* ya que en el portal no salía como entregada la actividad y el auxiliar/catedrático no le asignó puntos como tal.

Para solucionar los valores nulos en la *FechaCalificación*, se tuvo que comparar este estudiante contra el resto de la clase. Esto se hizo al obtener la fecha mínima de Calificación que resultaba de la agrupación a nivel de *Semestre*, *Anio*, *Seccion* y *Actividad*. Luego de aplicar esta forma de tratar los nulos, el resultado era el esperado, cero nulos en esta columna.

Semestre	Anio	Seccion	Actividad	FechaCalificacion	fec_calificacion
1	2019	10	Examen corto 2	null	2019-06-17 02:36:12.116
1	2019	10	Final	null	2019-05-29 16:50:52.411
1	2019	10	Parcial 1	null	2019-02-04 14:44:50.692
1	2019	10	Parcial 2	null	2019-02-25 14:02:47.741
1	2019	10	Parcial 3	null	2019-03-27 16:31:56.571
1	2019	10	Parcial 4	null	2019-05-08 16:11:14.902
1	2019	10	Portafolio (entrega 4)	null	2019-05-10 20:55:55.904
1	2019	10	Roll Call Attendance	null	2019-01-27 21:54:27.334
1	2019	10	Simulacro 1	null	2019-01-25 20:28:34.932
1	2019	10	Simulacro 2	null	2019-02-15 20:17:01.647
1	2019	10	Simulacro 3	null	2019-03-15 20:21:24.280
1	2019	10	Simulacro 4	null	2019-04-30 19:38:34.398
...					
1	2019	10	Verificación del aprendizaje 7	null	2019-04-11 16:10:12.653

Figura 5.9: Valores nulos en la columna de *FechaCalificacion* y el valor final

Fecha de vencimiento

La columna *FechaVencimiento*, al igual que *FechaCalificación*, no presentó dificultad a la hora de intentar llenar los valores faltantes. Un patrón curioso, siempre que la *FechaVencimiento* es nula, la *FechaTodoElDia* también lo es. El trato de nulos fue muy parecida a la de *FecCalificacion*. Se hizo la misma agrupación y se obtuvo la fecha mínima que dio como resultado.

Al mismo tiempo, no se intentó de forma muy extensa quitar todos los nulos ya que era una columna que desde un principio no se vio mucho potencial, ya sea para ayudar con la predicción o para generar nuevas variables. El resultado de la limpieza a la columna *FecVencimiento* fue disminuir la cantidad de nulos de 3,770 a 2,451.

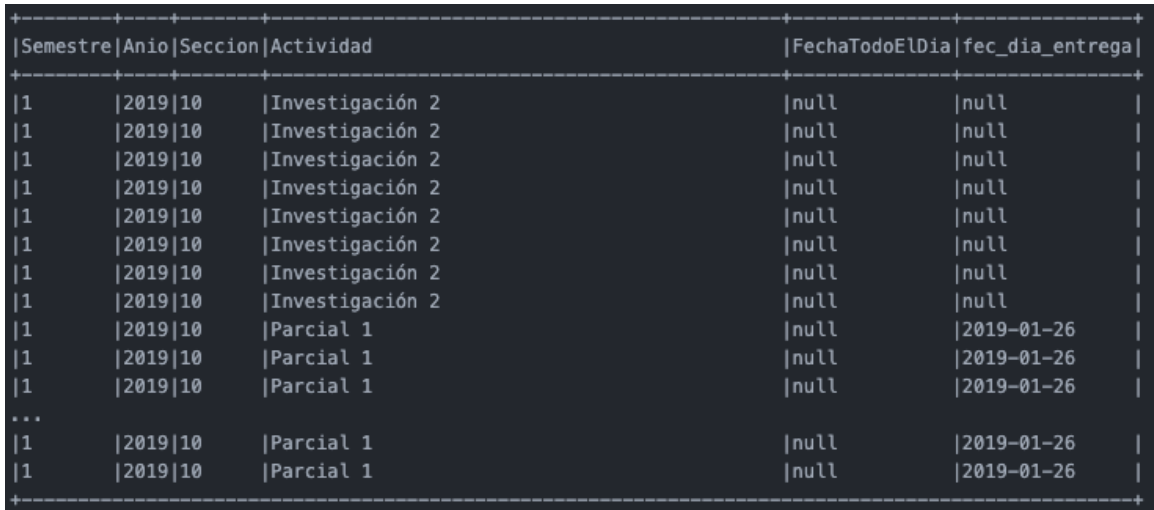
Semestre	Anio	Seccion	Actividad	FechaVencimiento	fec_venc
1	2019	10	HT	null	2019-04-12 21:00:00.000
1	2019	10	Parcial 1	null	2019-03-07 05:59:59.999
1	2019	10	Parcial 2	null	2019-04-18 05:59:59.999
1	2019	10	Parcial 3	null	2019-06-14 05:59:59.000
1	2019	10	Parcial 4	null	2019-05-04 05:59:59.000
1	2019	10	Simulacro 1	null	2019-01-26 05:59:59.000
1	2019	10	Simulacro 2	null	2019-02-16 05:59:59.000
1	2019	10	Simulacro 3	null	2019-03-16 05:59:59.000
1	2019	10	Simulacro 4	null	2019-04-27 05:59:59.000
1	2019	10	Verificación del aprendizaje 1	null	2019-01-12 05:59:59.000
1	2019	10	Verificación del aprendizaje 10	null	2019-05-18 05:59:59.000
1	2019	10	Verificación del aprendizaje 2	null	2019-01-19 05:59:59.000
...					
1	2020	10	Comprobación 1	null	2020-02-29 16:10:00.000

Figura 5.10: Valores nulos en la columna de *FechaVencimiento* y el valor final

Fecha todo el día

El caso de *FechaTodoElDia* era algo diferente a los casos de las dos fechas anteriores. A diferencia de las dos anteriores, esta columna iba a servir mucho, no sólo para la creación de variables, sino que también para ayudar a generar una línea del tiempo. La línea del tiempo es vital para terminar de crear la tabla base en el nivel de agrupación que escojamos y también, para poder ubicar al estudiante en cada una de las semanas del semestre. Por último, tuvo mucha importancia en la segunda fase del proyecto, el poder determinar a partir de qué semana se puede tener una predicción confiable.

Se empezó, teniendo 4,618 valores faltantes. Al igual que las dos fechas anteriores, se aplicó la misma agrupación para obtener la fecha mínima del resultado de la agrupación a nivel de *Semestre*, *Anio*, *Seccion* y *Actividad*.



```
+-----+-----+-----+-----+-----+-----+
|Semestre|Anio|Seccion|Actividad|FechaTodoElDia|fec_dia_entrega|
+-----+-----+-----+-----+-----+-----+
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Investigación 2|null|null|
|1|2019|10|Parcial 1|null|2019-01-26|
|1|2019|10|Parcial 1|null|2019-01-26|
|1|2019|10|Parcial 1|null|2019-01-26|
...
|1|2019|10|Parcial 1|null|2019-01-26|
|1|2019|10|Parcial 1|null|2019-01-26|
+-----+-----+-----+-----+-----+-----+
```

Figura 5.11: Valores nulos en la columna de *FechaTodoElDia* y el valor final, algunos valores con nulo y otros con la fecha final

A pesar del intento anterior, todavía quedaron varios registros con valor nulo. Debido a la importancia que tenía esta columna, no se podían dejar esos registros con nulo. Por ende, se procedió a ponerle la *FechaCalificacion* como la *FechaTodoElDia*. La decisión fue tomada de esta forma sobre poner *FechaVencimiento* por dos razones. Uno fue porque *FechaCalificacion* no tenía valores nulo, en cambio *FechaVencimiento* sí. La segunda, fue porque nos interesa más tener esa actividad en la fecha que fue calificada que en la fecha que se vencía (sabiendo que no siempre se encontraría fecha). Luego, de hacerte este arreglo, no quedaron valores nulos en la columna.

no. Para ambos casos, la escala estaba sobre ciento. Entonces, la solución fue quitar el porcentaje de la nota, para los casos que aplicara y aplicar la siguiente fórmula:

$$nota_tmp = \frac{Nota}{100} * PuntosPosibles$$

Semestre	Anio	Seccion	Actividad	Nota	ptos_posibles
1	2019	10	Reglas de derivadas	100	12.5
1	2019	10	Reglas de derivadas	100	12.5
1	2019	10	Reglas de derivadas	100	12.5
1	2020	10	Comprobación 1	80	5.9375
1	2020	10	Comprobación 1	100	5.9375
1	2020	10	Comprobación 1	60	5.9375
1	2020	10	Comprobación 1	100	5.9375
1	2020	10	Comprobación 1	90	5.9375
1	2020	10	Comprobación 1	100	5.9375
1	2020	10	Comprobación 1	80	5.9375
1	2020	10	Comprobación 1	90	5.9375
1	2020	10	Comprobación 1	100	5.9375
...					
2	2020	30	Verificación de aprendizaje 1	90	3.0
2	2020	30	Verificación de aprendizaje 1	90	3.0

Figura 5.13: Nota en la escala de porcentajes

Semestre	Anio	Seccion	Actividad	Nota	ptos_posibles
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	0%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	0%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	75%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
1	2019	10	Portafolio (entrega 3)	100%	2.0
...					
2	2021	150	Asincrónica 4	100%	1.0

Figura 5.14: Nota en la escala de porcentajes con el símbolo %

Ya corregido este caso, quedó como resultado lo siguiente:

Semestre	Anio	Seccion	Actividad	Nota	ptos_posibles	nota_tmp
1	2019 10		Portafolio (entrega 3)	100%	2.0	2.0
1	2019 10		Simulacro 1	83%	2.5	2.0749999999999997
1	2019 10		Verificación del aprendizaje 1	98%	3.0	2.94
1	2019 10		Verificación del aprendizaje 3	98%	3.0	2.94
1	2019 10		Verificación del aprendizaje 4	83%	3.0	2.4899999999999998
1	2019 10		Verificación del aprendizaje 5	100%	3.0	3.0
1	2019 10		Verificación del aprendizaje 8	94%	3.0	2.82
1	2019 10		Verificación del aprendizaje 9	100%	3.0	3.0
1	2019 10		Portafolio (entrega 3)	0%	2.0	0.0
1	2019 10		Simulacro 1	0%	2.5	0.0
1	2019 10		Verificación del aprendizaje 1	0%	3.0	0.0
1	2019 10		Verificación del aprendizaje 3	0%	3.0	0.0
...						
2	2022 110		Trabajo acumulativo 3	100%	3.0	3.0

Figura 5.15: Nota en la escala de porcentajes con el símbolo %

El siguiente caso que se tuvo que tratar fue en donde una misma *Actividad* aparecía dos veces con diferente nota y en casos, con diferentes *PuntosPosibles*. A veces, estas actividades aparecían con diferente *FechaTodoElDia* y *FechaCalificacion*. Esto ayudó al momento de corregir este caso ya que se hizo una validación para verificar que se obtuviera la *Nota* con la última *FechaCalificacion* para obtener la más reciente, siempre y cuando no fuera nula. A continuación un ejemplo del caso:

Semestre	Anio	Seccion	Actividad	Est_ID	Nota	ptos_posibles	nota_tmp	Nota_fix	fec_calificacion
1	2019 10		Simulacro 1 1353	83%	2.5	2.0749999999999997	83%	2019-01-25 20:34:46.183	
1	2019 10		Simulacro 1 1353	null	3.0	2.4899999999999998	83%	2019-01-25 20:28:34.932	
1	2019 10		Simulacro 1 1423	0%	2.5	0.0	0%	2019-02-04 21:27:35.346	
1	2019 10		Simulacro 1 1456	87%	2.5	2.175	87%	2019-02-04 14:50:39.837	
1	2019 10		Simulacro 1 1456	null	3.0	2.61	87%	2019-01-25 20:28:34.932	
1	2019 10		Simulacro 1 153	59%	2.5	1.4749999999999999	59%	2019-01-25 20:28:34.932	
1	2019 10		Simulacro 1 1663	70%	2.5	1.75	70%	2019-01-25 20:32:01.933	
1	2019 10		Simulacro 1 1663	null	3.0	2.0999999999999996	70%	2019-01-25 20:28:34.932	
1	2019 10		Simulacro 1 1713	55%	2.5	1.375	55%	2019-01-25 20:28:52.768	
1	2019 10		Simulacro 1 1725	80%	2.5	2.0	80%	2019-01-25 20:31:21.570	
1	2019 10		Simulacro 1 1725	null	3.0	2.4000000000000004	80%	2019-01-25 20:28:34.932	
1	2019 10		Simulacro 1 1770	93%	2.5	2.325	93%	2019-01-25 20:29:05.794	
...									
1	2019 10		Simulacro 1 968	55%	2.5	1.375	55%	2019-02-04 21:26:52.754	
1	2019 10		Simulacro 1 968	null	3.0	1.6500000000000001	55%	2019-01-25 20:28:34.932	

Figura 5.16: Misma *Actividad* con diferente *Nota* y *FechasCalificacion*

Como último caso a tratar con la Nota fue que habían algunas actividades que en lugar de tener *nota* numérica, tenían **complete** e **incomplete** y en la columna de *TipoCalificacion* tenían el valor **pass_fail**. La solución a este caso fue sencillo, si tenía como *Nota complete*, se le asignaba el valor que tenía en la columna *PuntosPosibles*, en caso el valor fuera **incomplete**, se le asignaba 0.

Semestre	Anio	Seccion	Actividad	Est_ID	TipoCalificacion	Nota	ptos_posibles	nota_tmp
1	2020	20	Clase 26/03	1065	pass_fail	complete	0.0	0.0
1	2020	20	clase2403	1065	pass_fail	complete	0.0	0.0
1	2020	20	Clase 26/03	1188	pass_fail	complete	0.0	0.0
1	2020	20	clase2403	1188	pass_fail	complete	0.0	0.0
1	2020	20	Clase 26/03	1199	pass_fail	complete	0.0	0.0
1	2020	20	clase2403	1199	pass_fail	incomplete	0.0	0
1	2020	20	Clase 26/03	1338	pass_fail	incomplete	0.0	0
1	2020	20	clase2403	1338	pass_fail	incomplete	0.0	0
1	2020	20	Clase 26/03	1552	pass_fail	complete	0.0	0.0
1	2020	20	clase2403	1552	pass_fail	complete	0.0	0.0
1	2020	20	Clase 26/03	1565	pass_fail	complete	0.0	0.0
1	2020	20	clase2403	1565	pass_fail	complete	0.0	0.0
...								
2	2020	150	Eliminatoria Integratón 2020	900	pass_fail	incomplete	0.0	0
2	2020	150	Hoja de trabajo 1 (7 de julio 2020)	900	pass_fail	complete	0.0	0.0

Figura 5.17: Actividad con Nota complete e incompleta

Con este último caso, completamos la limpieza y tratamiento de nulos a todas las variables que fueron de interés para las siguientes fases.

5.3.2. Feature engineering

Ya habiendo terminado con toda la limpieza y tratamiento de nulo, el siguiente paso fue hacer Feature Engineering. Lo primero que se hizo durante este paso, fue quedarnos con solo una observación por la combinación de *Sección - Anio - Semestre - Actividad - NombreEst - Est_ID*. Hasta el momento, se tienen casos en donde hay más de una observación en la combinación que se acaba de mencionar. Para lograr tener una sola observación, se hizo una agrupación y se obtuvo el último valor que aparecía. Esto fue hecho gracias a la función `last` de Spark. Esta función, si uno se lo especifica (es el `True` del código siguiente), puede obtener el último valor que no sea nulo entonces, era lo que se necesitaba para lograr el objetivo.

```

1 df_agg = df9.groupBy(
2     "Curso",
3     "Semestre",
4     "Seccion",
5     "Anio",
6     "Actividad",
7     "NombreEst",
8     "Est_ID"
9 ).agg(
10    f.last("ptos_posibles", True).alias("puntos_posibles"),
11    f.last("nota_tmp", True).alias("nota_tmp"),
12    f.last("RevisiónPares", True).alias("en_parejas"),
13    f.last("CantComentarios", True).alias("qty_comentarios"),
14    f.last("is_not_null", True).alias("no_es_nullo"),
15    f.last("proportion_nulls", True).alias("proposion_nulos"),
16    f.last("proportion_nulls_p_student", True).alias("proportion_nulls_p_student"),
17    f.last("fec_calificacion", True).alias("fec_calificacion"),
18    f.last("fec_venc", True).alias("fec_venc"),
19    f.last("fec_dia_entrega", True).alias("fec_entrega"),
20    f.last("tip_entrega", True).alias("tip_entrega"),
21    f.last("tip_envio", True).alias("tip_envio")
22 )

```

Listing 5.1: Código para obtener el nivel de agrupación en *Sección - Anio - Semestre - Actividad - NombreEst - Est_ID*

Aplicar este código en el dataset redujo los datos de 79,797 a 76,948.

Una vez ya teniendo este nivel de agregación, en donde solo tenemos un registro *Sección - Año - Semestre - Actividad - NombreEst - Est_ID*, se procedió a la creación de nuevas variables. Las variables creadas en esta etapa fueron las siguientes:

- nota acumulada por estudiante
- la nota final del estudiante
- nota acumulada por curso
- nota final del curso
- si es la primera que el estudiante se cursa el curso

Las variables anteriores, fueron creadas con el siguiente código:

```
1 df10 = df_agg.withColumn(  
2     "nota_acumulada",  
3     f.sum("nota_tmp").over(  
4         Window.partitionBy("Est_ID", "Anio", "Semestre").orderBy("fec_entrega").  
5         rangeBetween(Window.unboundedPreceding, Window.currentRow)  
6     )  
7 ).withColumn(  
8     "nota_final",  
9     f.max("nota_acumulada").over(  
10        Window.partitionBy("Est_ID", "Anio", "Semestre")  
11    )  
12 ).withColumn(  
13     "aprobado",  
14     f.when(f.col("nota_final") > 61, 1)  
15     .otherwise(0)  
16 ).withColumn(  
17     "semestre_anio",  
18     f.concat(f.col("Semestre"), f.lit("-"), f.col("Anio"))  
19 ).withColumn(  
20     "is_first_time",  
21     f.when(  
22         f.min("semestre_anio").over(  
23             Window.partitionBy("Est_ID").orderBy("Anio").rangeBetween(Window.  
24             unboundedPreceding, 0)  
25         ) == f.col("semestre_anio"), 1  
26     )  
27     .otherwise(0)  
28 ).withColumn(  
29     "nota_acumulada_curso",  
30     f.sum("puntos_posibles").over(  
31         Window.partitionBy("Est_ID", "Anio", "Semestre").orderBy("fec_entrega").  
32         rangeBetween(Window.unboundedPreceding, Window.currentRow)  
33     )  
34 ).withColumn(  
35     "nota_curso_final",  
36     f.max("nota_acumulada_curso").over(  
37         Window.partitionBy("Est_ID", "Anio", "Semestre")  
38     )  
39 )
```

Listing 5.2: Código para la creación de nuevas variables de nota final y aprobado

Lo siguiente fue crear el timeline por estudiante. Para esto, se obtuvo la fecha mínima y máxima del estudiante y las modas de ambas fechas. Esto se obtuvo de la columna *fec_entrega*. Una vez ya teníamos la fecha mínima y máxima y las modas de ambas, obteníamos la semana del año a la que pertenecían esas fechas. Se restó la semana del año de la fecha máxima con la mínima, para obtener

la duración del curso en semanas. Para ubicar al estudiante dependiente de la *fec_entrega* que tenía la *Actividad*, lo que se hizo fue obtener la semana del año de esa actividad. A la longitud del curso en semanas, se le restaba la semana de la fecha máxima con la resta de la semana de la actividad. La fórmula era:

$$week_course = length_weeks - (weekofyear(mode_max_fec) - weekofyear(fec_entrega))$$

El código es el siguiente:

```

1 df_fec_curso = df_final_f.withColumn(
2   "fec_entrega",
3   f.when(
4     (f.year(f.col("fec_entrega")) != f.col("Anio")), f.concat(f.col("Anio"), f.
5     lit("-"), f.month(f.col("fec_entrega")), f.lit("-"), f.dayofmonth(f.col("
6     fec_entrega"))).cast("date")
7   )
8   .otherwise(f.col("fec_entrega"))
9 ).withColumn(
10  "fec_entrega",
11  f.mode(f.col("fec_entrega")).over(
12    Window.partitionBy("Semestre", "Anio", "Seccion", "Actividad")
13  )
14 ).withColumn(
15  "min_fec",
16  f.min(f.col("fec_entrega")).over(
17    Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID")
18  )
19 ).withColumn(
20  "max_fec",
21  f.max(f.col("fec_entrega")).over(
22    Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID")
23  )
24 ).withColumn(
25  "mode_min_fec",
26  f.mode(f.col("min_fec")).over(
27    Window.partitionBy("Semestre", "Anio", "Seccion")
28  )
29 ).withColumn(
30  "mode_max_fec",
31  f.mode(f.col("max_fec")).over(
32    Window.partitionBy("Semestre", "Anio", "Seccion")
33  )
34 ).withColumn(
35  "week_year",
36  f.weekofyear(f.col("fec_entrega"))
37 ).withColumn(
38  "length_weeks",
39  f.weekofyear(f.col("mode_max_fec")) - f.weekofyear(f.col("mode_min_fec"))
40 ).withColumn(
41  "week_course",
42  f.col("length_weeks") - (f.weekofyear(f.col("mode_max_fec")) - f.weekofyear(f.col(
43  "fec_entrega")))
44 ).filter(
45  f.col("Actividad") != "Ejemplo Parcial 3 Cálculo"
46 ).withColumn(
47  "fec_entrega",
48  f.when(
49    f.col("week_course") < 0, f.col("fec_calificacion").cast("date")
50  )
51  .otherwise(f.col("fec_entrega"))
52 ).withColumn(
53  "week_course",
54  f.col("length_weeks") - (f.weekofyear(f.col("mode_max_fec")) - f.weekofyear(f.col(
55  "fec_entrega")))
56 )

```

Listing 5.3: Código para obtener la fecha inicial y final del curso y ubicar al estudiante en una semana del semestre

El siguiente paso, fue crear una variable con la que se pudieran categorizar algunas tareas. Luego de toda la limpieza, tratamiento de nulos, y la exploración que se tuvo realizando los dos pasos anteriores, se identificaron las siguientes categorías: Parciales, Proyectos, Cortos, Simulacros, y Refuerzo. La principal razón por la que se decidió hacer esta categorización de actividades es porque dentro de la columna de *Actividades* habían muchos nombres distintos entonces, el intentar

convertir estos nombres en variables iba a aumentar mucho la dimensionalidad del modelo. También, para poder crear nuevas variables que nos indiquen los cambios porcentuales entre la última actividad de la misma actividad. El código para hacer la categorización de actividades, indicando qué toma y no toma en cuenta para cada categoría es el siguiente:

```

1 df_tip_act = df_fec_curso.withColumn(
2   "tipo_actividad",
3   f.when(
4     (
5       (f.col("puntos_posibles") >= 15) |
6       (f.lower(f.col("Actividad")).contains("parcial")) |
7       (f.lower(f.col("Actividad")).contains("examen")) |
8       (f.lower(f.col("Actividad")).contains("final")) |
9       (f.lower(f.col("Actividad")).contains("evaluación"))
10    ) &
11    (
12      ~(
13        (f.lower(f.col("Actividad")).contains("proyecto")) |
14        (f.lower(f.col("Actividad")).contains("asincrónica")) |
15        (f.lower(f.col("Actividad")).contains("corto")) |
16        (f.lower(f.col("Actividad")).contains("discusión")) |
17        (f.lower(f.col("Actividad")).contains("ejerci")) |
18        (f.lower(f.col("Actividad")).contains("ejemplo")) |
19        (f.lower(f.col("Actividad")).contains("tarea")) |
20        (f.lower(f.col("Actividad")).contains("auto")) |
21        (f.lower(f.col("Actividad")).contains("trabajo"))
22      )
23    ), "parcial"
24  ).when(
25    (f.lower(f.col("Actividad")).contains("proyecto")), "proyecto"
26  ).when(
27    (f.lower(f.col("Actividad")).contains("verificación")) |
28    (f.lower(f.col("Actividad")).contains("comprobación")) |
29    (f.lower(f.col("Actividad")).contains("corto")) |
30    (f.lower(f.col("Actividad")).contains("corta")), "corto"
31  ).when(
32    (
33      (f.lower(f.col("Actividad")).contains("simulacro"))
34    ) &
35    (
36      ~(f.lower(f.col("Actividad")).contains("ejercicios"))
37    )
38  ), "simulacro"
39  ).when(
40    (f.lower(f.col("Actividad")).contains("ejercicio")) |
41    (f.lower(f.col("Actividad")).contains("ejercitación")) |
42    (f.lower(f.col("Actividad")).contains("tarea")) |
43    (f.lower(f.col("Actividad")).contains("hoja de trabajo")) |
44    (f.lower(f.col("Actividad")).contains("hojas de trabajo")) |
45    (f.lower(f.col("Actividad")).contains("h.t. ")) |
46    (f.lower(f.col("Actividad")).contains("ht")) |
47    (f.lower(f.col("Actividad")).contains("repaso")), "ejercicios"
48  )
49  .otherwise("refuerzo")
50 )

```

Listing 5.4: Código para la categorización de actividades

Ahora, al ya tener las categorizaciones hechas, se pudo calcular variables como promedios en cada una de los tipos de actividad y el promedio porcentual de las actividades. Esto se hizo de la siguiente forma:

```

1 .withColumn(
2   "prom_tipo_actividad",
3   f.when(
4     f.col("tipo_actividad") == "tipo_actividad", f.avg(f.col("nota_tmp")).over(
5       Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID", "
6         tipo_actividad").orderBy(f.col("week_course")).rangeBetween(Window.
7         unboundedPreceding, 0)
8     )
9   )
10 ).withColumn(
11   "prom_tipo_actividad_por",
12   f.when(
13     f.col("tipo_actividad") == "tipo_actividad", f.avg(f.col("nota_tmp")/f.col("
14     puntos_posibles")).over(
15     Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID", "
16     tipo_actividad").orderBy(f.col("week_course")).rangeBetween(Window.
17     unboundedPreceding, 0)
18   )
19 )

```

Listing 5.5: Código para obtener los promedios de los puntos netos y del porcentaje de cada tipo de actividad

Teniendo estas nuevas variables creadas, se procedió con el siguiente paso, que consistía en reducir el nivel de agrupación del dataset nuevamente. Anteriormente, se tenía en *Sección - Anio - Semestre - Actividad - NombreEst - Est_ID*. Para este paso, se necesitaba que estuviera en *Semestre - Seccion - Anio - NombreEst - Est_ID - week_course*. Esto se hizo con el fin de poder tener una observación por semana por estudiante. Se decidió llegar a este nivel de agregación porque esto es lo que tendría que recibir el modelo para poder analizar semana por semana quiénes van a aprobar o reprobar. El código es el siguiente:


```

1 df_final_agg = df_proms.groupBy(
2     "Curso",
3     "Semestre",
4     "Seccion",
5     "Anio",
6     "NombreEst",
7     "Est_ID",
8     "week_course"
9 ).agg(
10    f.max("length_weeks").alias("duracion_curso"),
11    f.max("nota_acumulada").alias("nota_acumulada_estudiante"),
12    f.sum("nota_tmp").alias("nota_semana_estudiante"),
13    f.max("nota_acumulada_curso").alias("nota_acumulada_curso"),
14    f.sum("puntos_posibles").alias("nota_semana_curso"),
15    f.max("qty_comentarios").alias("qty_comentarios"),
16    f.max("aprobado").alias("aprobado"),
17    f.max("is_first_time").alias("primera_asignacion"),
18    f.last("prom_parciales").alias("prom_parciales"),
19    f.last("prom_parciales_por").alias("prom_parciales_por"),
20    f.last("prom_proyectos").alias("prom_proyectos"),
21    f.last("prom_proyectos_por").alias("prom_proyectos_por"),
22    f.last("prom_cortos").alias("prom_cortos"),
23    f.last("prom_cortos_por").alias("prom_cortos_por"),
24    f.last("prom_simulacros").alias("prom_simulacros"),
25    f.last("prom_simulacros_por").alias("prom_simulacros_por"),
26    f.last("prom_refuerzos").alias("prom_refuerzos"),
27    f.last("prom_refuerzos_por").alias("prom_refuerzos_por"),
28    f.last("prom_ejercicios").alias("prom_ejercicios"),
29    f.last("prom_ejercicios_por").alias("prom_ejercicios_por")
30 ).fillna(
31     value = 0,
32     subset = [
33         "prom_parciales",
34         "prom_parciales_por",
35         "prom_proyectos",
36         "prom_proyectos_por",
37         "prom_cortos",
38         "prom_cortos_por",
39         "prom_simulacros",
40         "prom_simulacros_por",
41         "prom_refuerzos",
42         "prom_refuerzos_por",
43         "prom_ejercicios",
44         "prom_ejercicios_por"
45     ]
46 )

```

Listing 5.6: Código para obtener el nivel de agregación *Semestre - Seccion - Anio - NombreEst - Est_ID - week_course*

Con lo que tenemos hasta el momento, pareciera que ya terminamos esta fase y estamos listos para cambiar de paso del proyecto, pero no hay un problema. Actualmente sólo se tienen las semanas en donde hubo una actividad. Hay semanas en donde el estudiante no aparece, ya sea porque no se calificó nada o no se entregaba nada o simplemente no aparecía en el API de Canvas. Esto afecta al entrenamiento del modelo porque no se van a tener siempre los mismos datos dependiendo de la semana que se esté evaluando.

Curso	Semestre	Seccion	Anio	Est_ID	week_course
CALCULO 1	2	20	2019	1728	0
CALCULO 1	2	20	2019	1728	1
CALCULO 1	2	20	2019	1728	2
CALCULO 1	2	20	2019	1728	3
CALCULO 1	2	20	2019	1728	4
CALCULO 1	2	20	2019	1728	5
CALCULO 1	2	20	2019	1728	6
CALCULO 1	2	20	2019	1728	7
CALCULO 1	2	20	2019	1728	8
CALCULO 1	2	20	2019	1728	10
CALCULO 1	2	20	2019	1728	11
CALCULO 1	2	20	2019	1728	12
CALCULO 1	2	20	2019	1728	13
CALCULO 1	2	20	2019	1728	14
CALCULO 1	2	20	2019	1728	15
CALCULO 1	2	20	2019	1728	17

Figura 5.18: Saltos entre semanas, no hay una secuencia seguida

Para corregir lo anterior, lo que se hizo fue hacer una copia del dataset. A esta copia del dataset, se determinó la semana mínima que aparece un estudiante. Con esto, se obtuvo una secuencia de las semanas desde la primera hasta la última fecha. Teniendo la secuencia, se modificó el dataset para que quedara un registro por cada semana obtenido por la secuencia. Teniendo esta secuencia, se le hizo un left join al dataset original. Esto nos dió como resultado el dataset original, pero con las semanas que anteriormente no estaban con valor nulo en todas las columnas. Para arreglar y quitar los valores nulos, se aplicó nuevamente la función **last** para que obtenera los datos de la última semana que no fuera nulo el dato. El código es el siguiente:

```

1 df_fec_completas = .withColumn(
2   "min_week",
3   f.min(f.col("week_course")).over(
4     Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy("
5     week_course")
6   )
7 ).groupBy(
8   "Curso",
9   "Semestre",
10  "Seccion",
11  "Anio",
12  "NombreEst",
13  "Est_ID",
14  "min_week",
15  "duracion_curso"
16 ).agg(
17   ## APLICAMOS LA FUNCION sequence PARA OBTENER UNA SECUENCIA DESDE LA FECHA
18   ## MINIMA
19   ## HASTA LA FECHA MAXIIMA DADA EN UNA LISTA
20   f.sequence("min_week", "duracion_curso").alias("semanas")
21 ).withColumn(
22   "week_course",
23   ## CON LA FUNCION EXPLODE, PODEMOS CONVERTIR UNA COLUMNA QUE ES UNA LISTA A
24   ## UNA FILA POR ELEMENTO DE LA LISTA
25   f.explode("semanas")
26 ).drop("semanas", "min_week", "duracion_curso").join(
27   df_final_agg,
28   ["Curso", "Semestre", "Seccion", "Anio", "NombreEst", "Est_ID", "week_course"],
29   "left"
30 )
31 ## LLENAMOS LOS VALORES NULOS DE CADA COLUMNA CON EL ULTIMO VALOR NO NULO
32 for i in df_fec_completas.columns:
33   df_fec_completas = df_fec_completas.withColumn(
34     i,
35     f.when(
36       f.col(i).isNull(), f.last(f.col(i), True).over(
37         Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy(f
38         .col("week_course"))
39       )
40     )
41     .otherwise(f.col(i))
42 )

```

Listing 5.7: Código para obtener aquellas semanas en las que un estudiante no apareciera

Curso	Semestre	Seccion	Anio	Est_ID	week_course
CALCULO 1	2	20	2019	1728	0
CALCULO 1	2	20	2019	1728	1
CALCULO 1	2	20	2019	1728	2
CALCULO 1	2	20	2019	1728	3
CALCULO 1	2	20	2019	1728	4
CALCULO 1	2	20	2019	1728	5
CALCULO 1	2	20	2019	1728	6
CALCULO 1	2	20	2019	1728	7
CALCULO 1	2	20	2019	1728	8
CALCULO 1	2	20	2019	1728	9
CALCULO 1	2	20	2019	1728	10
CALCULO 1	2	20	2019	1728	11
CALCULO 1	2	20	2019	1728	12
CALCULO 1	2	20	2019	1728	13
CALCULO 1	2	20	2019	1728	14
CALCULO 1	2	20	2019	1728	15
CALCULO 1	2	20	2019	1728	16
CALCULO 1	2	20	2019	1728	17

Figura 5.19: Ya no hay saltos entre semanas

Como último paso de Feature Engineering, se calcularon variables de cambio para cada una de las actividades. La temporalidad que se escogió para hacer estas variables de cambio es 3 y 5 semanas. También, se creó una nueva variable que indicaba el porcentaje de puntos que un estudiante obtuvo durante cada semana. El código para las variables es el siguiente:

```

1 df_final_final = df_fec_completa.withColumn(
2     "por_puntos_obtenidos_semana",
3     f.when(
4         f.col("nota_semana_curso") == 0, f.col("nota_semana_estudiante") / (f.col("
5         nota_semana_curso") + 0.00001)
6     )
7     .otherwise(f.col("nota_semana_estudiante") / f.col("nota_semana_curso"))
8 ).withColumn(
9     "prom_tipo_actividad_3w",
10    f.avg(f.col("prom_tipo_actividad")).over(
11        Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy("
12        week_course").rangeBetween(-3, Window.currentRow)
13    )
14 ).withColumn(
15     "prom_tipo_actividad_5w",
16    f.avg(f.col("prom_tipo_actividad")).over(
17        Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy("
18        week_course").rangeBetween(-5, Window.currentRow)
19    )
20 ).withColumn(
21     "prom_tipo_actividad_por_3w",
22    f.avg(f.col("prom_tipo_actividad_por")).over(
23        Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy("
24        week_course").rangeBetween(-3, Window.currentRow)
25    )
26 ).withColumn(
27     "prom_tipo_actividad_por_5w",
28    f.avg(f.col("prom_tipo_actividad_por")).over(
29        Window.partitionBy("Semestre", "Anio", "Seccion", "Est_ID").orderBy("
30        week_course").rangeBetween(-5, Window.currentRow)
31    )
32 )

```

Listing 5.8: Código para obtener los promedios de 3 y 5 semanas para atrás

El resultado de Feature Engineering fue obtener un dataset de 51,620 registros y 52 columnas.

5.3.3. Selección de variables

En esta parte del proceso, se analizó el impacto de las diferentes variables en los modelos.

1. Se hizo una búsqueda para obtener el listado de variables constantes o cuasi constantes. Esto sirve ya que se eliminaron estas variables. La razón detrás es porque cuando una variable es constante o cuasi constante, no le aporta nada al modelo. El modelo no puede aprender de una variable que es o siempre la misma o casi siempre la misma. Por esta razón, se eliminaron las variables que resultaron ser constantes o cuasi constantes. Las variables que se eliminaron durante este proceso fueron:

- *week_course*
- *prom_proyectos*
- *prom_proyectos_por*
- *prom_proyectos_3w*
- *prom_proyectos_5w*
- *prom_proyectos_por_3w*
- *prom_proyectos_por_5w*
- *len_course*

5.3.4. Escoger el mejor algoritmo

Durante esta etapa, se utilizó todo lo que se había trabajado previamente. Se utilizó el dataset que se estuvo trabajando en la limpieza y exploración y las nuevas variables y el nivel de agregación de Selección de Variables. En esta última parte de la primera fase, se unió el proceso de Selección de Variables con el entrenamiento del modelo. Esto se hizo debido a la técnica de Selección de Variables que se está utilizando.

Durante esta parte, se decidió entrenar y evaluar en la semana 16 (no equivale a la semana 16 del semestre). Se cargaron los datos luego de hacerles la limpieza y después de pasar por una primera fase de Selección de Variables. Al tener cuatro años (2019, 2020, 2021, 2022) de información, se decidió que el dataset de entrenamiento iba a ser todo 2019, 2020, 2021 y el primer semestre de 2022. Lo que nos deja con el segundo semestre de 2022 como dataset de test. Primero, se separó la información de entrenamiento y la de prueba. Segundo, se balancearon los datos de entrenamiento, es decir, la misma cantidad de estudiantes aprobados y reprobados. Para lograr este balanceo, lo que se utilizó fue **Random Under Sampling**, reduciendo la población de alumnos que aprobaron el curso. Tercero, se eliminaron las variables constantes y quasi-constantes. Cuarto, previo a empezar con el entrenamiento (para algunos modelos), se normalizó la data. Quinto, una vez ya normalizada la data (si es necesario, dependiendo el modelo a evaluar), se procedió a hacer el entrenamiento de los modelos.

En el entrenamiento del modelo, se realizaron pruebas con los siguientes algoritmos: Support Vector Machine (SVM), Gradient Boosted Trees (GBT), Regresión Logística (LR), KNN, y Redes Neuronales (NN). Para cada algoritmo se utilizaron parámetros aleatorios y Cross Validation. Una vez entrenado el modelo con todas las variables y obtenidas sus métricas, se hizo el proceso de Recursive Feature Elimination (RFE) para eliminar las variables que no tenían un aporte significativo al modelo. Después de eliminar esas variables, se repitió el proceso de entrenamiento y recopilación de métricas. Este proceso fue repetido para cada uno de los algoritmos que se evaluaron.

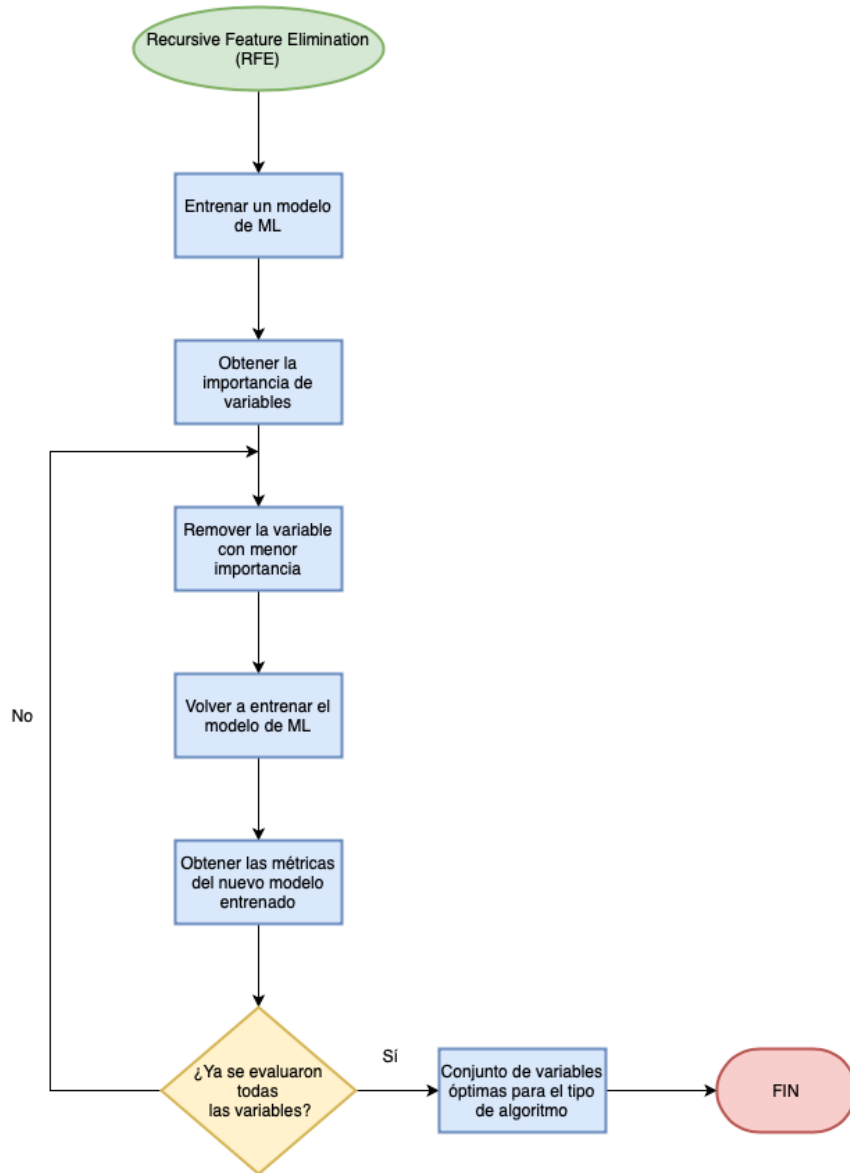


Figura 5.20: Diagrama que muestra el proceso de RFE [17]

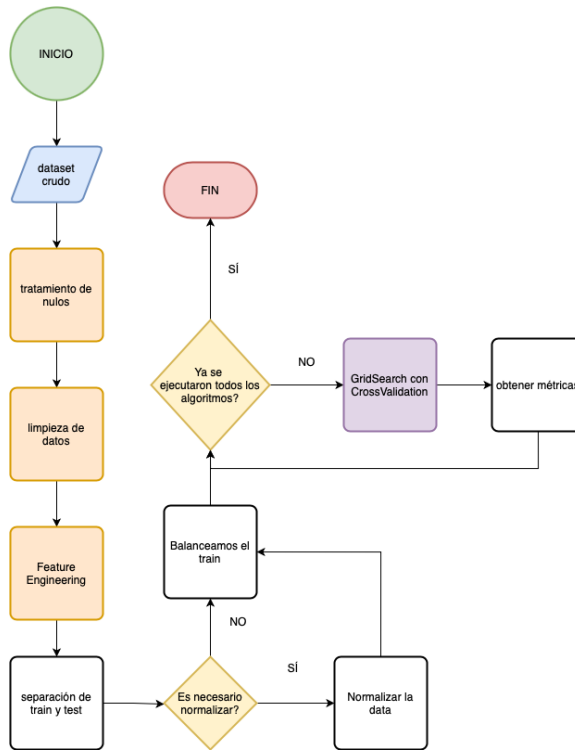


Figura 5.21: Diagrama de flujo del proceso sin Selección de Variables

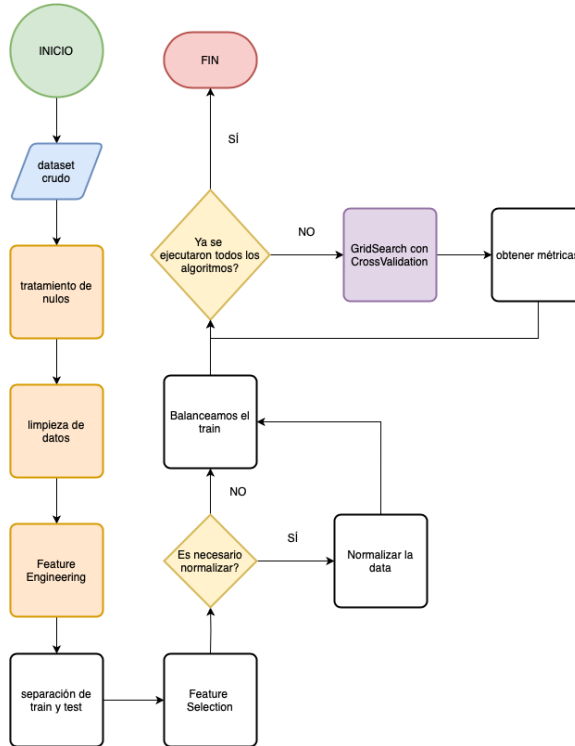


Figura 5.22: Diagrama de flujo del proceso con Selección de Variables

Una vez ya se habían ejecutado todos los modelos, se recopilaron todas las métricas y fueron analizados por medio de gráficas para poder determinar cuál de todos los algoritmos tenía el mejor desempeño en identificar a aquellos estudiantes que reprueban el curso. También, se analizaron los resultados comparando los resultados obtenidos previo y después de aplicarle Selección de Variables. Por último, se seleccionó el mejor modelo.

5.4. Segunda etapa

Una vez obtenidos los resultados de la primera fase, se procedió a determinar qué tan temprano se puede llegar a predecir con una precisión en los alumnos reprobados mayor o igual a 75% a un estudiante que va a reprobado el curso. Para esta fase se repitieron muchos de los pasos que se realizaron durante la primera fase. La diferencia, es que ahora varió el tiempo con el que con el cual las variables fueron calculadas.

Como se dijo anteriormente, se repitió todo lo que se hizo en la primera fase. Esto incluye el proceso de Selección de Variables, el probar con todos los algoritmos mencionados antes y después de aplicarles Selección de Variables, y la forma de analizar los resultados.

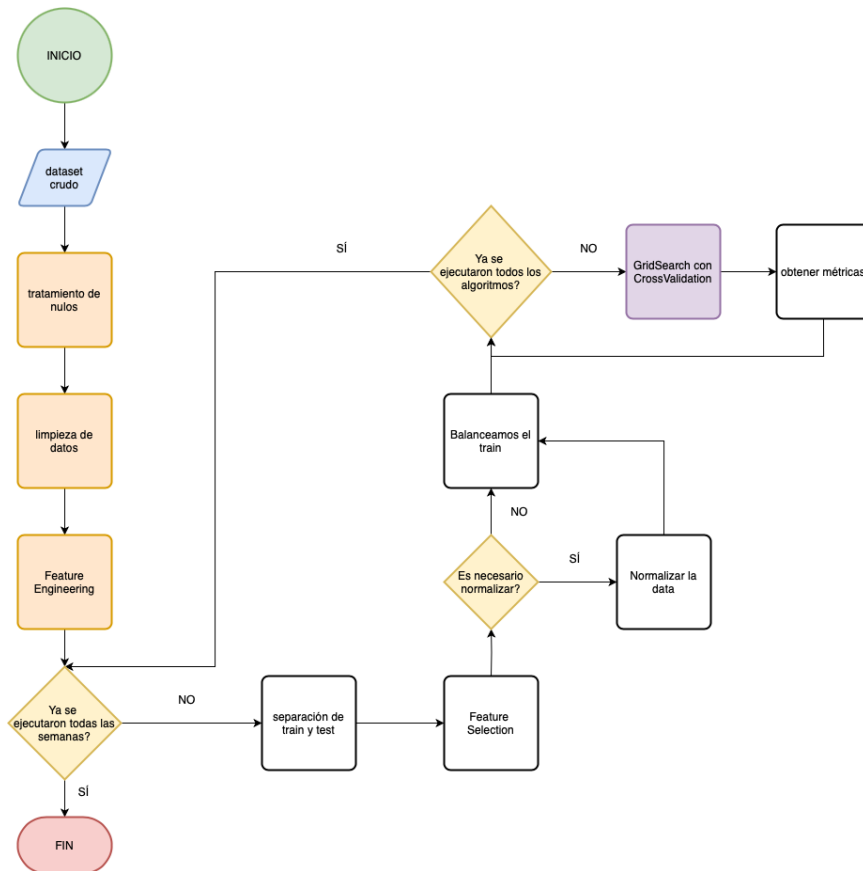


Figura 5.23: Diagrama de flujo de la segunda etapa del proyecto

Durante el proyecto fueron realizados 14 pruebas y análisis para poder determinar qué combinación entre aplicación de Selección de Variables y algoritmo es el que mejor logra predecir qué estudiantes van a reprobado un el curso de Cálculo 1. La primera prueba realizada fue para lograr encontrar el algoritmo que mejor desempeño tenía utilizando 16 semanas sin ningún tipo de transformación en los datos. El segundo, fue replicar la misma prueba que la primera pero aplicando técnicas de Selección de Variables como Recursive Feature Elimination. Cabe mencionar que durante las 14 pruebas, se estuvieron utilizando los mismos algoritmos (con parámetros aleatorios), mismas métricas, y un método de 5 cross-fold validation.

6.1. Primera etapa

Modelo	Precision	Accuracy	Sensitivity	Specificity	F1
SVM	0.916	0.927	0.995	0.698	0.954
GBT	0.898	0.901	0.983	0.627	0.938
LR	0.859	0.873	1.0	0.452	0.924
KNN	0.819	0.76	0.883	0.349	0.85
NN	0.769	0.769	1.0	0.0	0.869

Tabla 6.1: Métricas: primer análisis - 16 semanas - ningún método de Selección de Variables

Durante el primer análisis, en donde no se aplicó ningún método de Selección de Variables luego del proceso de limpieza y Feature Engineering, se obtuvieron los resultados de la **Tabla 6.1**. Se lograron obtener modelos con buenas métricas. El SVM llegó a obtener 93.7% en Accuracy, el puntaje más alto. El resto de algoritmos tuvieron un desempeño de 76.9% para arriba.

Modelo	VN	FP	FN	VP
SVM	88	38	2	417
GBT	79	47	7	412
LR	57	69	0	419
KNN	44	82	49	370
NN	0	126	0	419

Tabla 6.2: Matriz de confusión: primer análisis - 16 semanas - ningún método de Selección de Variables

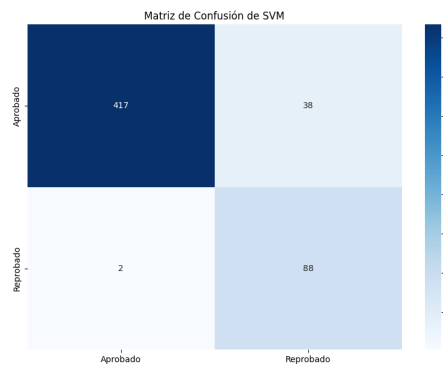


Figura 6.1: Matriz de confusión de SVM

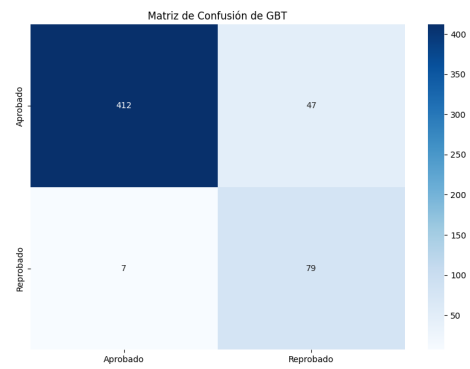


Figura 6.2: Matriz de confusión de GBT

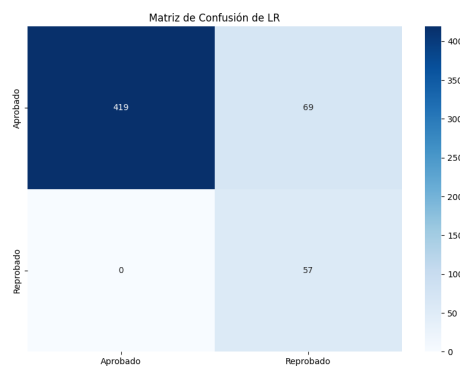


Figura 6.3: Matriz de confusión de LR

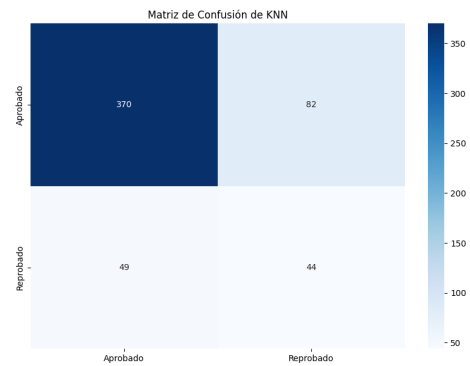


Figura 6.4: Matriz de confusión de KNN

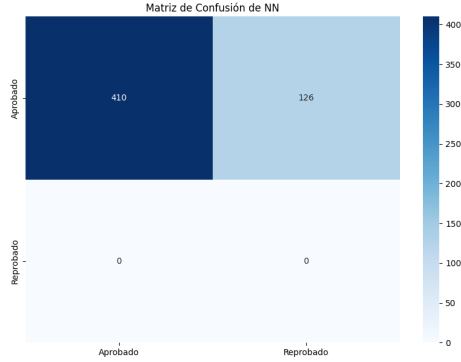


Figura 6.5: Matriz de confusión de NN

Al ver la **Tabla 6.2**, podemos ver cómo es que los modelos están separando a los estudiantes entre estudiantes que aprueban y reprobaban. Al ver más a detalle cómo los modelos asignaban a los estudiantes, se pudo observar que las métricas obtenidas en la **Tabla 6.1** eran un poco engañosas. Se puede observar que todos los modelos asignan muy bien a los estudiantes que aprueban los cursos, siendo el mejor LR y NN. A pesar de saber identificar los estudiantes que aprueban los cursos, que no están asignando estudiantes en alguna de las categorías. También, se puede observar que el SVM es el que más asigna estudiantes que reprobaban el curso de forma correcta.

Para este segundo análisis, se mantuvo el tiempo de entrenamiento (16 semanas), los mismos algoritmos, y las mismas transformaciones durante el proceso de limpieza y Feature Engineering. A diferencia del primer análisis, se aplicó el método de Selección de Variables llamado Recursive Feature Elimination (RFE). En conjunto, el método de RFE, la eliminación de variables constantes y cuasi-constantes, se eliminaron un total de 20 variables, entre ellas estaban:

Variable	Descripción	Método
<i>Est_ID</i>	Identificador único del estudiante	RFE
<i>week_course</i>	Semana en la que se encuentra el estudiante dentro del semestre	Cuasi-Constante
<i>nota_acumulada_curso</i>	Puntos que se llevan del curso, obtenida de los puntos posibles de las actividades	RFE
<i>prom_parciales_por</i>	Promedio obtenido de los puntos netos obtenidos durante esa semana del tipo de actividad que corresponda a parciales	RFE
<i>prom_proyectos</i>	Promedio obtenido de los puntos netos obtenidos durante esa semana del tipo de actividad que corresponda a proyectos	Cuasi-Constante
<i>prom_proyectos_por</i>	Promedio obtenido del porcentaje obtenido de la actividad durante esa semana del tipo de actividad que corresponda a proyectos	Cuasi-Constante

<i>prom_simulacros_por</i>	Promedio obtenido del porcentaje obtenido de la actividad durante esa semana del tipo de actividad que corresponda a simulacros	RFE
<i>prom_refuerzos_por</i>	Promedio obtenido del porcentaje obtenido de la actividad durante esa semana del tipo de actividad que corresponda a refuerzos	RFE
<i>prom_ejercicios</i>	Promedio obtenido de los puntos netos obtenidos durante esa semana del tipo de actividad que corresponda a ejercicios	RFE
<i>prom_ejercicios_por</i>	Promedio obtenido del porcentaje obtenido de la actividad durante esa semana del tipo de actividad que corresponda a ejercicios	RFE
<i>por_puntos_obtenidos_semana</i>	Porcentaje de los puntos obtenidos de la semana	RFE
<i>prom_parciales_5w</i>	Promedio del tipo de actividad que corresponda a parciales. Es calculada obteniendo el promedio de los puntos netos de las últimas 5 semanas de donde se está parado para atrás	RFE
<i>prom_proyectos_3w</i>	Promedio del tipo de actividad que corresponda a proyectos	Cuasi-Constante
<i>prom_proyectos_5w</i>	Promedio del tipo de actividad que corresponda a proyectos	Cuasi-Constante
<i>prom_proyectos_por_3w</i>	Promedio del tipo de actividad que corresponda a proyectos	Cuasi-Constante
<i>prom_proyectos_por_5w</i>	Promedio del tipo de actividad que corresponda a proyectos	Cuasi-Constante
<i>prom_cortos_5w</i>	Promedio del tipo de actividad que corresponda a cortos	RFE
<i>prom_refuerzos_por_3w</i>	Promedio del tipo de actividad que corresponda a proyectos	RFE
<i>prom_ejercicios_5w</i>	Promedio del tipo de actividad que corresponda a ejercicios	RFE
<i>len_course</i>	Longitud en semanas que dura el curso	Contante

Tabla 6.3: Lista de variables que se eliminaron del conjunto de datos junto a su descripción y el método por el que fueron eliminadas

Luego de aplicar los métodos de Selección de Variables se obtuvo un listado de variables a eliminar (**Tabla 6.3**). Al eliminar las variables anteriores, el conjunto de datos final contenía las variables que se encuentran en la **Tabla 6.4**.

Variable	Descripción
<i>Semestre</i>	Semestre en el que se está cursando
<i>Seccion</i>	Sección a la que el estudiante está asignado
<i>Anio</i>	Año en el cual el estudiante está cursando el curso
<i>duracion_curso</i>	Duración que tiene el curso en semanas
<i>nota_semana_estudiante</i>	Puntos netos obtenidos durante la semana
<i>nota_semana_curso</i>	Puntos netos posibles del curso
<i>tqy_comentarios</i>	Suma de la cantidad de comentarios que se tienen en las actividades de la semana
<i>primera_asignacion</i>	Indicador si es la primera vez que un estudiantes toma el curso de cálculo 1
<i>prom_parciales</i>	Promedio de los puntos netos obtenidos en el tipo de actividad que corresponde a parciales durante la semana
<i>prom_cortos</i>	Promedio de los puntos netos obtenidos en el tipo de actividad que corresponde a cortos durante la semana
<i>prom_cortos_por</i>	Promedio del porcentaje obtenido en el tipo de actividad que corresponde a cortos durante la semana
<i>prom_simulacros</i>	Promedio de los puntos netos obtenidos en el tipo de actividad que corresponde a simulacros durante la semana
<i>prom_refuerzos</i>	Promedio de los puntos netos obtenidos en el tipo de actividad que corresponde a refuerzos durante la semana
<i>prom_refuerzos_por</i>	Promedio del porcentaje obtenido en el tipo de actividad que corresponde a refuerzos durante la semana
<i>prom_parciales_3w</i>	Promedio de los puntos netos obtenidos en las últimas 3 semanas que corresponde al tipo de actividad parciales
<i>prom_parciales_por_3w</i>	Promedio de los porcentajes obtenidos en las últimas 3 semanas que corresponde al tipo de actividad parciales
<i>prom_parciales_por_5w</i>	Promedio de los porcentajes obtenidos en las últimas 5 semanas que corresponde al tipo de actividad parciales
<i>prom_cortos_3w</i>	Promedio de los puntos netos obtenidos en las últimas 3 semanas que corresponde al tipo de actividad cortos
<i>prom_cortos_por_3w</i>	Promedio de los porcentajes obtenidos en las últimas 3 semanas que corresponde al tipo de actividad cortos
<i>prom_cortos_por_5w</i>	Promedio de los porcentajes obtenidos en las últimas 5 semanas que corresponde al tipo de actividad cortos
<i>prom_simulacros_3w</i>	Promedio de los puntos netos obtenidos en las últimas 3 semanas que corresponde al tipo de actividad simulacros
<i>prom_simulacros_5w</i>	Promedio de los puntos netos obtenidos en las últimas 5 semanas que corresponde al tipo de actividad simulacros
<i>prom_simulacros_por_3w</i>	Promedio de los porcentajes obtenidos en las últimas 3 semanas que corresponde al tipo de actividad simulacros

<i>prom_simulacros_por_5w</i>	Promedio de los porcentajes obtenidos en las últimas 5 semanas que corresponde al tipo de actividad simulacros
<i>prom_refuerzos_3w</i>	Promedio de los puntos netos obtenidos en las últimas 3 semanas que corresponde al tipo de actividad refuerzos
<i>prom_refuerzos_5w</i>	Promedio de los puntos netos obtenidos en las últimas 5 semanas que corresponde al tipo de actividad refuerzos
<i>prom_refuerzos_por_5w</i>	Promedio de los porcentajes obtenidos en las últimas 5 semanas que corresponde al tipo de actividad refuerzos
<i>prom_ejercicios_3w</i>	Promedio de los puntos netos obtenidos en las últimas 3 semanas que corresponde al tipo de actividad ejercicios
<i>prom_ejercicios_por_3w</i>	Promedio de los porcentajes obtenidos en las últimas 3 semanas que corresponde al tipo de actividad ejercicios
<i>prom_ejercicios_por_5w</i>	Promedio de los porcentajes obtenidos en las últimas 5 semanas que corresponde al tipo de actividad ejercicios

Tabla 6.4: Lista de variables finales con su descripción

Modelo	Precision	Accuracy	Sensitivity	Specificity	F1
SVM	0.956	0.954	0.986	0.849	0.971
GBT	0.916	0.923	0.99	0.698	0.952
LR	0.935	0.945	0.998	0.77	0.965
KNN	0.85	0.855	0.986	0.421	0.913
NN	0.769	0.769	1.0	0.0	0.869

Tabla 6.5: Métricas: segundo análisis - 16 semanas - aplicando RFE

Al ejecutar los mismos pasos del primer análisis, obtuvimos las métricas de la **Tabla 6.5**. Los resultados de este segundo análisis, aplicando RFE, fueron mejores con respecto al primer análisis. Se nota un incremento en la Precision, Accuracy y en Specificity. También, se nota un decremento en la mayoría de los modelos en la métrica de Sensitivity. A pesar del decremento en Sensitivity, la métrica F1 aumentó. Ahora, el puntaje más alto en Accuracy fue de 95.4%, mientras que el resto de algoritmos tuvieron un desempeño de 76.9%.

Modelo	VN	FP	FN	VP
SVM	107	19	6	413
GBT	88	38	4	415
LR	97	29	1	418
KNN	53	73	6	413
NN	0	126	0	419

Tabla 6.6: Matriz de confusión: segundo análisis - 16 semanas - aplicando RFE

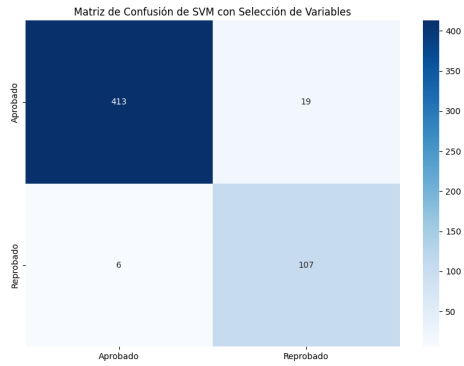


Figura 6.6: Matriz de confusión de SVM después de Selección de Variables

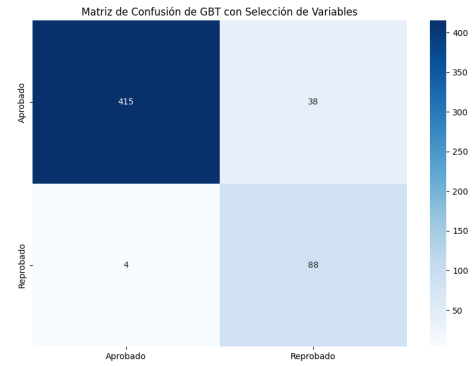


Figura 6.7: Matriz de confusión de GBT después de Selección de Variables

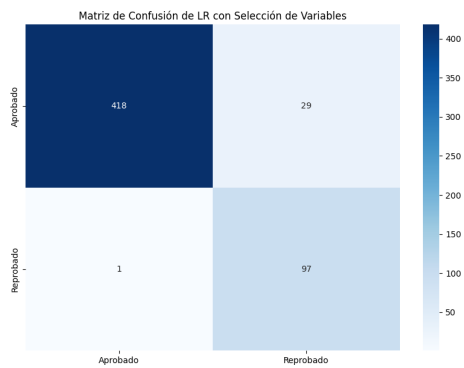


Figura 6.8: Matriz de confusión de LR después de Selección de Variables

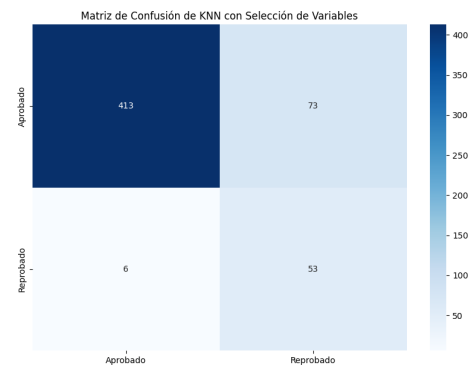


Figura 6.9: Matriz de confusión de KNN después de Selección de Variables

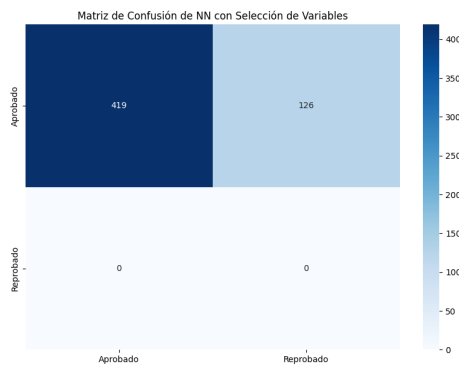


Figura 6.10: Matriz de confusión de NN después de Selección de Variables

Luego de analizar la matriz de confusión del primer análisis (**Tabla 6.2**) contra el segundo (**Tabla 6.6**), se puede observar que hay un incremento considerable en la asignación de estudiantes que perdieron el curso. También, hubo un decremento en la asignación de estudiantes que se predijo que iban a aprobar el curso y en realidad lo reprueban.

Como último paso, previo a dar por terminada la primera etapa del proyecto, se obtuvo la

importancia de las variables para ambos análisis. Al observar la **Figura 6.11**, se notó que la variable *prom_parciales_por* es la variable que más importancia tiene en el modelo, con más de 30%. Luego, es seguida por *prom_simulacros_por*, y *prom_ejercicios_por*. Curiosamente, las variables que más importancia tenían en el primer análisis, pertenecían al grupo de variables que por medio de Selección de Variables, fueron eliminadas del conjunto de datos. Esto dio como resultado la **Figura 6.12**, en donde la variable con mayor importancia es *prom_parciales* con más de 30%. Luego, es seguida por *prom_ejercicios_por_3w* y *prom_simulacros_por_3w*. Por último, en la **Figura 6.13**, se puede ver la comparación de la importancia de variables entre ambos análisis.

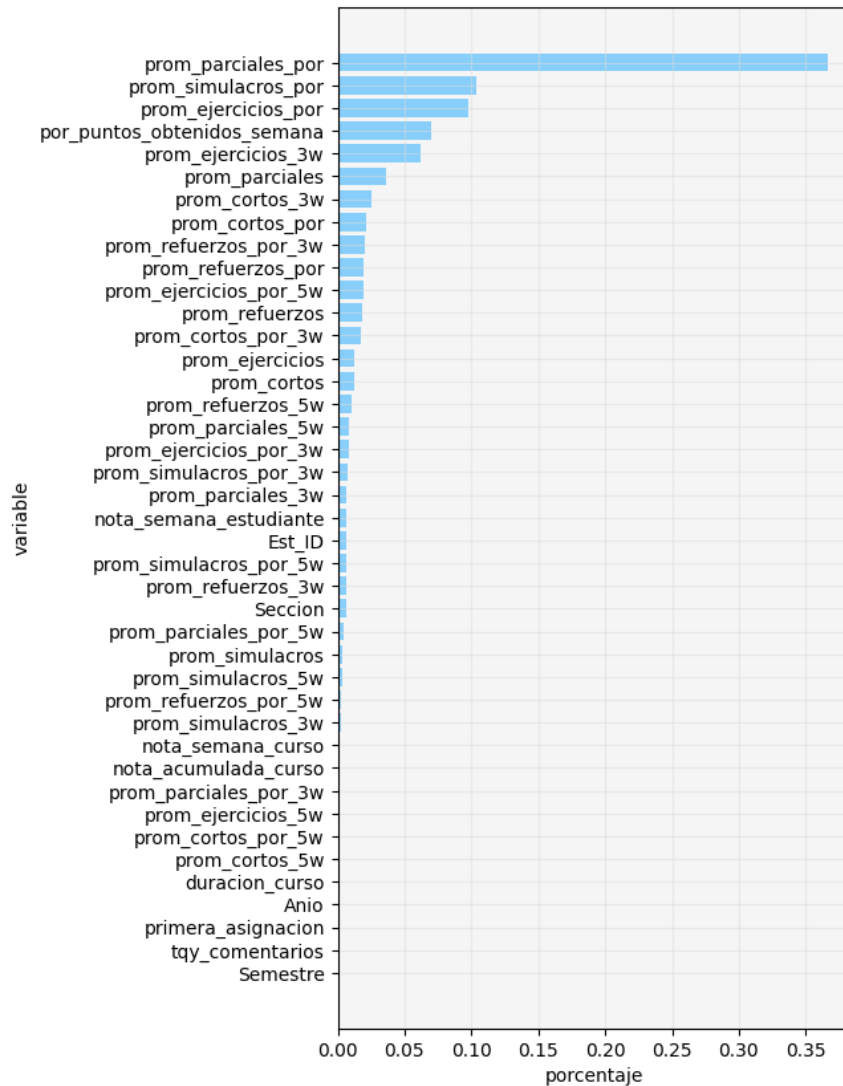


Figura 6.11: Importancia de variables sin aplicar Selección de Variables

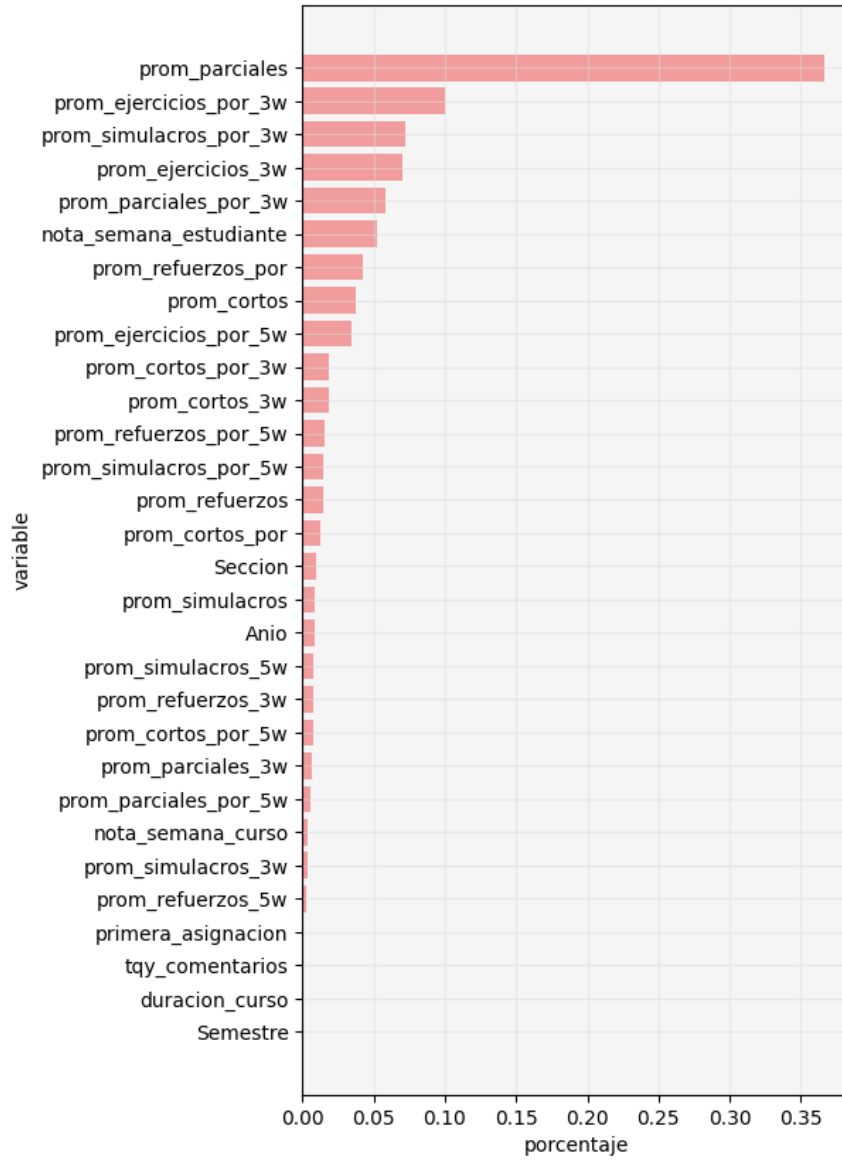


Figura 6.12: Importancia de variables después de aplicar Selección de Variables

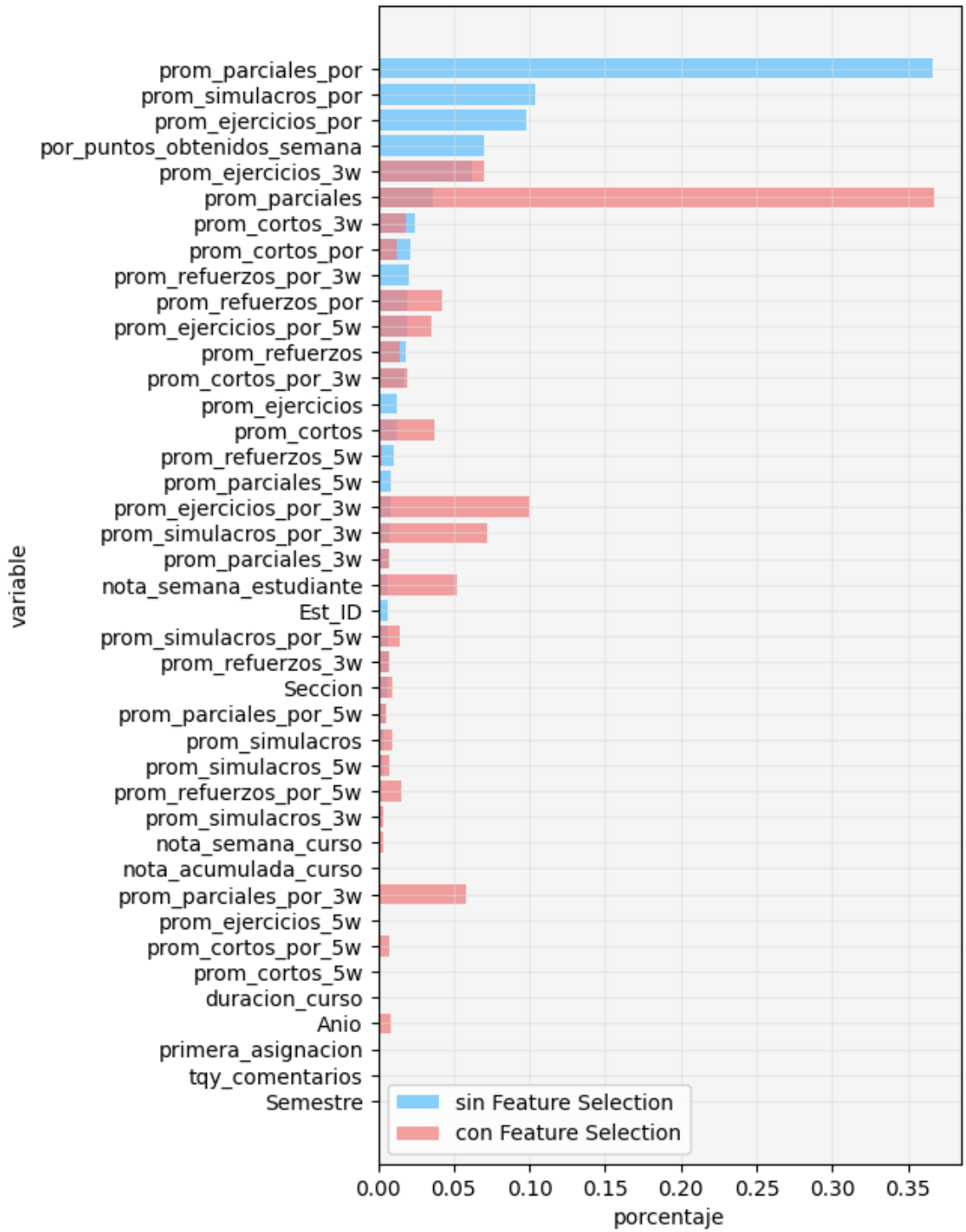


Figura 6.13: Comparación de importancia de variables al no aplicar y aplicar Selección de Variables

6.2. Segunda etapa

En la segunda etapa del proyecto, se hicieron un total de 12 análisis. Los 12 análisis fueron muy parecidos a los dos que se hicieron durante la primera etapa. La diferencia es que en cada par de análisis, se iba disminuyendo la información que contenía el conjunto de datos.

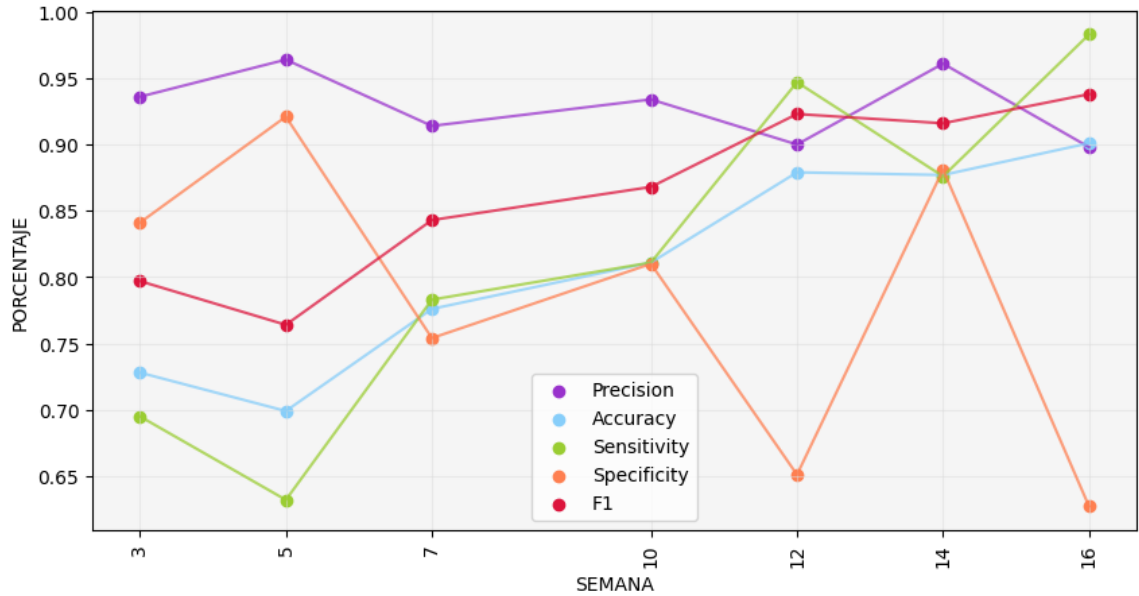


Figura 6.14: Desempeño de las métricas conforme avanzan las semanas con el algoritmo de GBT, sin aplicar Selección de Variables

En la **Figura 6.14** se puede observar cómo el algoritmo de GBT se comporta en las diferentes métricas en los diferentes puntos de evaluación. Se comparan las métricas utilizadas durante la primera fase para determinar a partir de qué semana es que ya se puede obtener una predicción confiable. Como se puede observar en la gráfica, desde la tercera semana, se tiene un Accuracy de cerca del 73% con un Specificity cerca del 85%. Conforme va aumentando la cantidad de semanas con el que el modelo fue entrenado, se puede ver cómo todas las métricas a excepción de Specificity alcanzan su mejor valor en la decimosexta semana.

Semana	VN	FP	FN	VP
Semana 3	106	20	128	291
Semana 5	116	10	154	265
Semana 7	95	31	91	328
Semana 10	102	24	79	340
Semana 12	82	44	22	397
Semana 14	111	15	52	367
Semana 16	79	47	7	412

Tabla 6.7: Matriz de confusión: segunda etapa - evaluación por semanas del algoritmo GBT - ningún método Selección de Variables

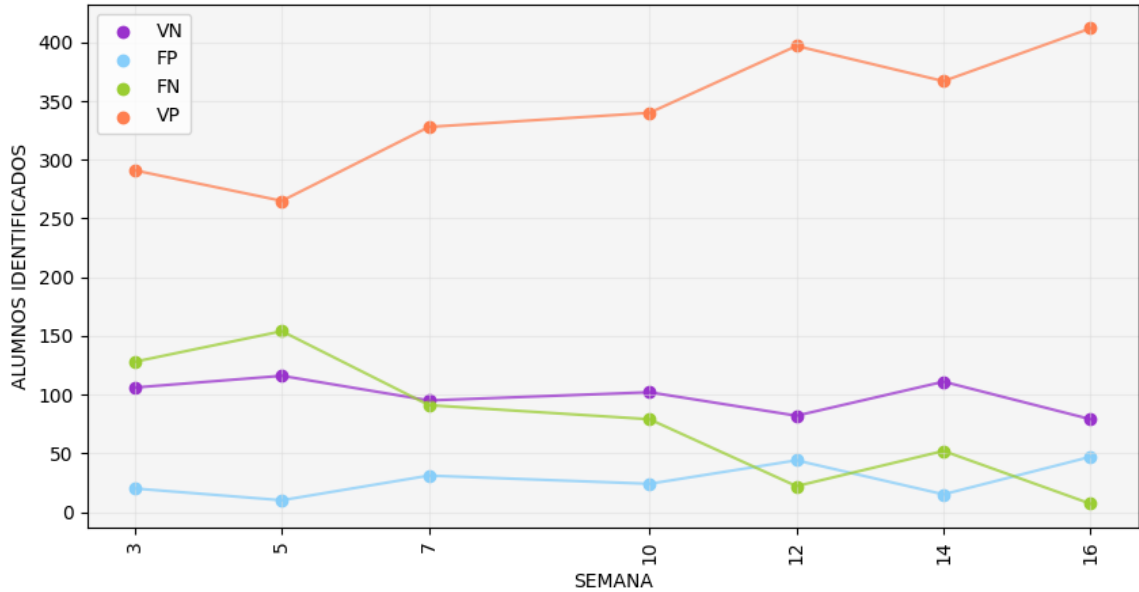


Figura 6.15: Matriz de confusión para cada semana utilizando el algoritmo GBT

Al obtener la matriz de confusión del algoritmo GBT para cada una de las semanas evaluadas se obtuvo la **Tabla 6.7**. Hay semanas en donde asigna muy bien a los alumnos que reprobaban la clase, pero no tan bien a los alumnos que aprueban y viceversa. En alumnos cuya correcta predicción en la clase reprobados, tenemos un mínimo de 79 y un máximo de 116. En alumnos cuya correcta predicción en la clase aprobados, hay un mínimo de 291 y un máximo de 412.

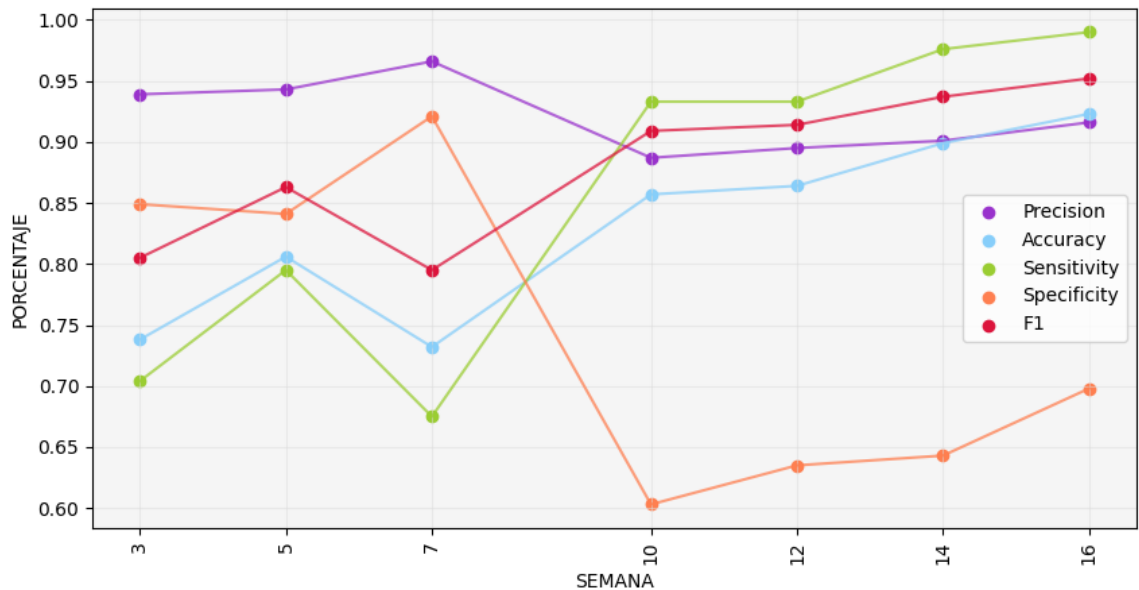


Figura 6.16: Desempeño de las métricas conforme avanzan las semanas con el algoritmo de GBT, aplicando Selección de Variables

Al aplicar las técnicas de eliminación de variables por medio de variables constantes, cuasi-constantes y RFE, siempre al algoritmo de GBT, se obtuvo los datos de la **Figura 6.16**. Se puede

observar que a comparación de la tercera semana de la **Figura 6.14**, tenemos un Accuracy de un poco menor al 75 % y un Specificity cerca de 85 %. También, se puede observar cómo el resultado final es el mismo a pesar de tener un comportamiento distinto, todas las métricas mejoran, alcanzando su punto máximo en la decimosexta semana, a excepción de la métrica Specificity.

Semana	VN	FP	FN	VP
Semana 3	107	19	124	295
Semana 5	106	20	86	333
Semana 7	116	10	136	283
Semana 10	76	50	28	391
Semana 12	80	46	28	391
Semana 14	81	45	10	409
Semana 16	88	38	4	415

Tabla 6.8: Matriz de confusión: segunda etapa - evaluación por semanas del algoritmo GBT - aplicando RFE

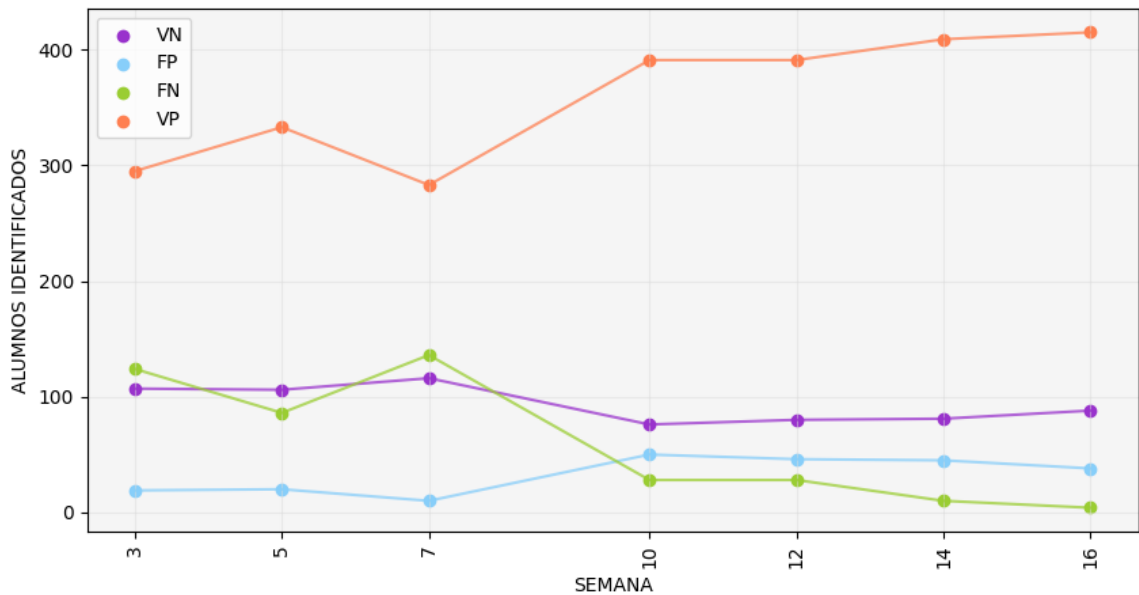


Figura 6.17: Matriz de confusión para cada semana utilizando el algoritmo GBT después de aplicar Selección de Variables

Como último paso de esta segunda fase, se obtuvo la matriz de confusión de cada iteración del algoritmo GBT en su respectiva semana. En cuanto a los alumnos que reprobaron, en la semana 10 se obtuvo el valor más bajo con 28 y en la semana 7 el valor más alto con 136. En cuanto a los alumnos que aprobaron, en la semana 3 se obtuvo el valor más bajo con 295 y en la semana 16 el valor más alto con 415.

El objetivo del proyecto era obtener un modelo que fuera lo suficientemente confiable no solo para predecir a final del curso quién reprobó el curso, sino que también para obtener la predicción lo más temprano posible. En el capítulo de *Metodología*, específicamente en la sección **5.1. Descripción General**, se hicieron varias preguntas que se deseaban responder durante el desarrollo del proyecto. Eran dos preguntas principales, cada una corresponde a una etapa distinta del proyecto. Por ende, el análisis de resultados se ve dividido en dos partes.

7.1. Primera etapa

Para la primera etapa, se hicieron preguntas relacionadas al impacto que tenía el uso de técnicas de pre-procesamiento y Selección de Variables en el rendimiento de los modelos de clasificación. También, se hizo la pregunta de qué variable resultó tener la mayor importancia en el rendimiento de los modelos.

Para empezar a responder esta pregunta, es necesario entender cómo cambió el dataset a lo largo del proyecto. El dataset original contaba con 155,158 registros y 17 columnas. Al explorar el dataset, se descubrió que más del 40 % de las columnas tenían valores nulos. Dentro del 40 % de columnas que tenían valores nulos, habían 3 columnas en específico (*FechaCalificacion* - 50.82 %, *Nota* - 50.87 %, y *TipoEnvio* - 68.79 %), que tenían más del 50 % de proporción de nulos con respecto al total del dataset. Gracias al pre-procesamiento realizado, se logró disminuir la cantidad de nulos a 0 en todas las columnas que eran de interés, no sólo para ayudar a la predicción, sino que también para la creación de nuevas variables. Fueron utilizadas varias técnicas, no solo para la limpieza del dataset, sino que también para el tratamiento de nulos. Sin todo el trabajo de pre-procesamiento realizado, no hubiera sido posible obtener resultados.

Previo a discutir los resultados, es necesario explicar qué significa cada una de las métricas que fueron utilizadas durante el proyecto dentro del contexto del proyecto. Dentro de las métricas están:

- **Precision:** La precisión indica qué proporción de los que fueron identificados como positivos son correctos. Hay casos, como este, en donde los datos están desbalanceados que puede que no refleje los verdaderos resultados. Este, como se mencionó, fue uno de ellos ya que la población que se intentaba predecir de forma correcta era la población que tiene menos observaciones.

Por lo tanto, se tuvieron que manejar los resultados de esta métrica con pinzas.

- **Accuracy:** Accuracy o exactitud nos indica qué tan bien se están prediciendo a los Verdaderos Positivos y Verdaderos Negativos. Esta métrica al igual que la Precisión puede llegar a estar sesgada en datasets desbalanceados. Puede que tenga un puntaje alto, pero las asignaciones en la clase que tiene una minoría (en el contexto del proyecto, los reprobados) no se esté asignando de forma correcta. De nuevo, los resultados de esta métrica fueron manejados con pinzas.
- **Sensitivity:** Esta métrica, a comparación de las dos anteriores, no se ve afectada o sesgada por datasets desbalanceados. Esto se debe a que gracias a lo que intenta representar. Sensitivity o también llamada Recall, intenta mostrar qué tan bien el modelo está identificando a aquellos estudiantes que aprobaron el curso.
- **Specificity:** Al igual que Sensitivity, no se ve afectada o sesgada por datasets desbalanceados. Lo que intenta mostrar es qué tan bien el modelo está identificando a los estudiantes que reprobaban el curso. Esta métrica fue súper importante dentro del contexto de este proyecto, ya que queremos identificar de forma correcta a aquellos estudiantes que reprobaban el curso.

7.1.1. Primer análisis

Al analizar las métricas obtenidas durante el primer análisis, se encontraron resultados mixtos. Tres de los cinco algoritmos mostraron un Accuracy mayor 85% en la métrica Accuracy, el mejor siendo Support Vector Machine (SVM) con un puntaje de 92.7%. Con respecto a la métrica Sensitivity, todos los algoritmos estaban por encima del 85%, incluso dos algoritmos (LR y NN) tuvieron un puntaje perfecto. Por último, la métrica Specificity tiene resultados diferentes a las primeras dos métricas. Hay un algoritmo (NN) que tiene 0%, otros dos (KNN y LR) menor a 50% y los últimos (GBT y SVM) por arriba del 62%. Nuevamente, SVM es el algoritmo con mejor puntaje, con 69.8%.

Los algoritmos tuvieron buenas métricas y tuvieron una tasa alta en la predicción de aquellos estudiantes que sí aprobaron el curso. Se pudo observar que en la matriz de confusión, cuatro de los cinco algoritmos asignaron más de 410 estudiantes como aprobado de forma correcta.

Se logró identificar el problema de esta fase al ver métrica por métrica de una forma más detallada. Al analizar la métrica de Specificity, a comparación de Sensitivity y Accuracy, ninguno de los algoritmos pudo obtener un puntaje mayor al 70%. Al ver la matriz de confusión, se pudo entender el resultado de la métrica Specificity de una mejor manera. De los cinco algoritmos, tres tuvieron más Falsos Positivos que Verdaderos Negativos. Luego, los últimos dos algoritmos asignaban casi en una proporción de por cada dos Verdaderos Negativos, un Falso Positivo. Los algoritmos de esta fase tenían dificultad en asignar a los estudiantes que reprobaban el curso de forma correcta.

Por último, se obtuvo la importancia de cada variable en los algoritmos. Como se puede observar en la **Figura 6.11**, las variables que mayor importancia tuvieron fueron *prom_parciales_por*, *prom_simulacros_por*, *prom_ejercicios_por*, *por_puntos_obtenidos_semana*, y *prom_ejercicios_3w*. De la importancia de las variables se puede observar que los algoritmos lograron identificar un patrón en cómo se diferenciaban los alumnos que reprobaban contra los que aprobaban el curso utilizando esas cinco variables principalmente. Dentro de las variables está los porcentajes que se tienen durante la semana en actividades de tipo parciales, simulacros y ejercicios.

7.1.2. Segundo análisis

El segundo análisis realizado, a diferencia del primero, sufrió de los métodos de Selección de Variables (Constantes, Cuasi-Constantes, Recursive Feature Elimination). El resultado de esto, fue la eliminación de 20 variables del dataset original. De las 20 variables eliminadas, 9 tenían información

que correspondía a porcentajes de la nota obtenida en su respectiva actividad, 8 correspondían a información agrupada de 3 o 5 semanas atrás.

Después de aplicar los métodos de Selección de Variables y conservar únicamente las variables que tenían mayor importancia se procedió a ejecutar los algoritmos del primer análisis. Tres de cinco algoritmos tuvieron un puntaje superior a 92% en Accuracy, siendo SVM el mejor algoritmo con un puntaje de 95.4%. Al ver la métrica Sensitivity, los cinco modelos se encuentran con un puntaje mayor a 98%, el algoritmo NN tuvo un puntaje perfecto mientras que los algoritmos SVM y KNN tuvieron el resultado más bajo con 98.6%. Por último, al ver Specificity, se obtuvieron mejores resultados en comparación a los del primer análisis. El algoritmo con el puntaje más alto fue el SVM con 84.9% y el más bajo siguió siendo la NN con un puntaje de 0%.

A comparación del primer análisis, se puede ver una mejora en todos los aspectos. Mejoró el Accuracy, Sensitivity y Specificity. La mejora en estas métricas se entendió de una mejor manera al observar la matriz de confusión. Los cinco algoritmos asignaron de forma correcta más de 410 alumnos que aprobaron el curso. Tres de los cinco algoritmos asignaron, de forma correcta, más de 85 estudiantes a la categoría de reprobado.

Respondiendo la primera pregunta de la primera etapa, se pudo apreciar un incremento en todas las métricas utilizadas para evaluar a los algoritmos. Este incremento en las métricas fue obtenido gracias a la aplicación de diferentes métodos de Selección de Variables sobre el dataset original. Se logró observar que el incremento más significativo va en la categoría que se estuvo intentando optimizar, los alumnos que reprueban el curso.

Por último, respondiendo a la segunda pregunta, se puede apreciar en la **Figura 6.12** cómo cambia la importancia de las variables con respecto al primer análisis. Las primeras cuatro variables que mayor importancia tenían en el primer análisis fueron eliminadas por el RFE. El nuevo modelo, sustituyó esas variables por algunas variables de puntos netos como *prom_parciales* y *prom_ejercicios_3w*. Después, varias variables que tienen la información de las últimas 3 semanas ganaron mucha importancia en la predicción.

7.2. Segunda etapa

Para la segunda etapa de este proyecto, se hizo la pregunta de qué tan temprano en el semestre es posible predecir que un estudiante va a reprobar el curso con seguridad. Para esta etapa, se reusó la metodología de trabajo del segundo análisis de la primera etapa, solo modificando la semana en la que se situaba.

Durante esta etapa, se probó obtener predicciones estando ubicados en la semana 3, 5, 7, 10, 12, 14, y 16. Para cada una de las iteraciones de la semana, se probó con los 5 algoritmos para obtener el mejor resultado posible. Para cada una de las predicciones obtenidas, se les calculó todas las métricas que se utilizaron durante la primera etapa del proyecto.

Para estos resultados, se utilizó el algoritmo de GBT únicamente. La razón por la que se escogió GBT en lugar de SVM fue porque al calcular las predicciones de algunas semanas con SVM daba error o simplemente se quedaba corriendo por horas sin terminar. Otra de las razones fue porque al querer obtener la importancia de variables, daba error con el SVM, pero con el GBT no.

Como se pudo observar en la **Figura 6.14**, desde la semana 3, ya se tenían resultados aceptables. Se tuvo un Accuracy de casi 80%, una Sensitivity de casi 70%, y un Specificity de cerca de 85%. Conforme fue avanzando las semanas las métricas también cambiaron. En algunas semanas las métricas disminuían como el paso de la semana 3 a 5 y en otras, aumentaba como el paso de la semana 7 a la 10. Para todas las métricas, a excepción de Specificity, alcanzaron su punto máximo en la semana 16.

Cuando se compararon los resultados de la **Figura 6.14** con la **Figura 6.16** se puede observar como las métricas están más cercanas entre ellas. No existe una gran diferencia entre cada métrica en la misma semana. Adicionalmente, existen menos cambios entre semanas. En la **Figura 6.16** hubo un comportamiento extraño entre la semana 7 y la 10. Mientras todas las métricas disminuyeron su puntaje, Specificity aumentó en la semana 7. Durante la semana 10, el comportamiento fue inverso, mientras todas las métricas aumentaron su puntaje, Specificity lo disminuyó de forma considerable. A partir de la semana 10 en adelante, hubo incrementos en todas las métricas, alcanzando así su punto máximo (menos Specificity) durante la semana 16.

Al observar ambas gráficas junto con las matrices de confusión, se puede deducir que desde la semana 3 del semestre de cálculo, ya se puede tener una predicción lo suficientemente confiable como para actuar en base a ella. No existe mucha diferencia cuando se ubicó en la semana 3 entre aplicar o no Selección de Variables. Las mejoras al aplicar Selección de Variables, no son considerables. La razón por la que se está tomando la aplicación de Selección de Variables sobre la que no lo tiene es por una combinación de factores. Las métricas son un poco mejores, conforme avanza el tiempo, las predicciones son más estables, es menos tiempo de ejecución el que se necesita al ser menos variables y la interpretación de los resultados es más sencilla.

Respondiendo la pregunta de la segunda etapa, ubicados en la semana 3, se tendría una capacidad de identificar a las personas que van a reprobado el curso cerca del 85%, asignando a más de 100 estudiantes.

Como se menciona varias veces durante el trabajo escrito del proyecto, uno de los objetivos principales del proyecto era el poder determinar con seguridad qué estudiantes iban a reprobado el curso de Cálculo 1 lo antes posible. Durante el proyecto se tuvo que realizar trabajos de limpieza de datos, feature engineering, selección de variables, grid search, cross validation, y obtención y análisis de métricas.

Este dataset tuvo cierto grado de dificultad. Había muchas inconsistencias en el dataset que no deberían de estar ya que Cálculo 1 es un curso coordinado. Todas las secciones del mismo semestre y año deberían de tener las mismas actividades con los mismos puntos. Esto puede ser causado por la forma en que se registran las notas, por diferencias en el uso que cada catedrático hace de la plataforma o por transformaciones aplicadas previo al presente análisis. Considero que el trabajo realizado en esta fase del proyecto fue el adecuado para dejar el dataset listo para construir cualquier modelo sobre él. También, considero que las técnicas aplicadas fueron necesarias para ayudarnos a mantener la mayor cantidad de datos posibles.

Después de la limpieza del dataset, se aplicó Feature Engineering. Durante esta etapa se crearon más de 20 variables de todo tipo. Varias de las variables nuevas creadas se encuentran entre las variables más importantes del modelo predictivo. Esto muestra que los cálculos hechos contienen información importante y valiosa para los modelos desarrollados.

Para reducir la dimensionalidad de los modelos, se decidió ponerle mucho énfasis durante todo el proyecto a la fase de Selección de Variables. En lo personal, considero que esta etapa de los proyectos de ciencia de datos y sus técnicas ayudan mucho a mejorar un modelo o a simplificarlo. Durante los capítulos de **Resultados** y **Análisis de Resultados** se puede ver el impacto que tiene el aplicar esta fase en los proyectos, mejorando considerablemente las métricas de diferentes modelos. Como lleva los modelos de no tener métricas del todo buenas a mejorarlas considerablemente. El ejemplo perfecto se pudo observar en este proyecto durante la primera etapa, logrando incrementar el Sepcificity de 69% a 85%.

Finalmente, es claro que los objetivos del proyecto se cumplieron. Se construyó un algoritmo que puede identificar al final del semestre quiénes reprobarán el curso y un modelo que logra identificar en la tercera semana quiénes reprobarán con una certeza del 85%.

Como posibles áreas de mejora para el proyecto se podría buscar cómo obtener una mayor estandarización entre las diferentes secciones del curso. También, se podrían agregar variables que

actualmente no están en el LMS (Canvas). Por último, se podrían hacer diferentes pruebas con más algoritmos, hiperparámetros definidos que se sabe que funcionan, más recursos de cómputo para agilizar los tiempos y realizar más pruebas, tener un asesoramiento o apoyo con el departamento de matemática para resolver cualquier duda, etc...

-
- [1] Aggarwal, C. C.: *Data mining: The textbook*. Springer, 2015.
- [2] AI, SKY ENGINE: *What is Transfer Learning?*, 2023. <https://skyengine.ai/se/skyengine-blog/128-what-is-transfer-learning>
- [3] AWS: *What is Reinforcement Learning?*, 2023. [https://aws.amazon.com/what-is/reinforcement-learning/#:~:text=Reinforcement%20learning%20\(RL\)%20is%20a,use%20to%20achieve%20their%20goals.](https://aws.amazon.com/what-is/reinforcement-learning/#:~:text=Reinforcement%20learning%20(RL)%20is%20a,use%20to%20achieve%20their%20goals.)
- [4] Baker, R. S. J. d., Inventado C. S.: *Educational data mining and learning analytics: An introduction to the field, its goals and benefits*. In R. S. J. d. Baker C. S. Inventado (Eds.). *Educational data mining and learning analytics: Techniques for improving research, practice, and policy* (pp. 1-13). Springer, 2014.
- [5] Barnes, T., Desmarais M. Romero C. Ventura S.: *Educational Data Mining 2009 [Conference presentation]*. 2nd International Conference On Educational Data Mining, 2009.
- [6] Berens, J., Schneider K. Gortz S. Oster S. Burghoff J.: *Early detection of students at risk-predicting student dropouts using administrative student data from German universities and machine learning methods*. *Educational Data Mining*, 11(3)(zenodo.3594771):1–41, 2019. <https://doi.org/10.5281/zenodo.3594771>.
- [7] Bermolen, Paola. Capdehourat, Germán. Etcheverry Lorena. Fachola Christian. Fariello María Inés. Tornaría Agustín.: *FLEA: Aprendizaje Federado aplicado a Analíticas de Aprendizaje*. Facultad de Ingeniería, Universidad de la República, 2022.
- [8] Bogarín, A., Cerezo R. Romero C.: *Discovering learning processes using Inductive Miner: A case study with Learning Management Systems (LMSs)*. *Psicothema*, 30(3)(psicothema2018.116):322–329, 2018. <http://dx.doi.org/10.7334/psicothema2018.116>.
- [9] Brownlee, J.: *Feature engineering for machine learning: Traditional techniques and a practical guide with Python*. *Machine Learning Mastery*, 2019.
- [10] C. Romero, S. Ventura: *Educational data mining: A survey from 1995 to 2005*. *Expert Systems with Applications*, 33:135–146, 2007. <https://doi.org/10.1016/j.eswa.2006.04.005>.
- [11] Castro, F., Vellido A. Nebot À. Mugica F.: *Applying Data Mining Techniques to e-Learning Problems*. Springer, 62:183–221, 2007. https://doi.org/10.1007/978-3-540-71974-8_8.

- [12] Conway, D.: *THE DATA SCIENCE VENN DIAGRAM*, 2010. <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>.
- [13] Du-Harpur, X. y cols.: *What is AI? Applications of artificial intelligence to dermatology*. British Journal of Dermatology, 183:423–430, 2020. <https://doi.org/10.1111/bjd.18880>.
- [14] F., Chollet: *Transfer learning fine-tuning*, 2023. https://www.tensorflow.org/guide/keras/transfer_learning.
- [15] Fayyad, U., Piatetsky Shapiro G. Smyth P.: *Data Mining to Knowledge Discovery in Databases*. Artificial Intelligence Magazine, 13(3)(aimag.v17i3.1230):37, 1996. <https://doi.org/10.1609/aimag.v17i3.1230>.
- [16] Ferguson, R., Watson J.: *Educational data mining and learning analytics: A practical guide for educators*. 2013.
- [17] Galli, Soledad: *Recursive feature elimination with Python*, 2022. <https://www.blog.trainindata.com/recursive-feature-elimination-with-python/>.
- [18] García-Peñalvo, F. J., Martínez Aldán R.: *Learning analytics: Current landscape and future trends*. Progress in Artificial Intelligence, 5(4):335–350, 2016.
- [19] Girish Chandrashekar, Ferat Sahin: *A survey on feature selection methods*. Computers Electrical Engineering, 2014.
- [20] Google: *Evaluate Models Using Metrics*, 2022. <https://developers.google.com/machine-learning/testing-debugging/metrics/metrics>.
- [21] Guyon, I., Elisseeff A.: *Feature selection: A data mining perspective*. The MIT Press, 2003.
- [22] Guyon, Isabelle; Elisseeff, Andr  ©: *Feature selection for machine learning: A review of the state-of-the-art*. Data Mining and Knowledge Discovery, 2003.
- [23] G  khan Ak  apinar, Arif Altun y Petek Askar: *Using Learning analytics to debelop early-warning system for at-risk studentes*. International Journal of Educational Technology in Higher Education, 2019.
- [24] Hall, Mark A.; Frank, Eibe; Holmes Geoffrey; Pfahringer Bernhard; Reutter Sebastian; Witten Ian H.: *Feature selection: A data perspective*. Data Mining and Knowledge Discovery, 2009.
- [25] Hand, D.: *Assessing the Performance of Classification Methods*. International Statistical Review, 2012.
- [26] Huang, J., Ding C. Liu H.: *A review of wrapper methods for feature selection*. Data Mining and Knowledge Discovery, 2012.
- [27] J., VanderPlas.: *Python Data Science Handbook*. O’Reilly, 2016.
- [28] L. Chen, P. Chen y Z. Lin: *Artificial Intelligence in Education: A Review*. IEEE Access, 8:75264–75278, 2020. [10.1109/ACCESS.2020.2988510](https://doi.org/10.1109/ACCESS.2020.2988510).
- [29] Menon, K.: *Feature Selection in Machine Learning: All You Need To Know*, 2023. <https://www.simplilearn.com/tutorials/machine-learning-tutorial/feature-selection-in-machine-learning#:~:text=Feature%20Selection%20is%20the%20method,you%20are%20trying%20to%20solve.>
- [30] Merino, Marcos: *Conceptos de inteligencia artificial: qu   es el aprendizaje por refuerzo*, 2019. <https://www.xataka.com/inteligencia-artificial/conceptos-inteligencia-artificial-que-aprendizaje-refuerzo>.
- [31] Mining, Educational Data: *Educational Data Mining*. <https://educationaldatamining.org>.

- [32] Murphy, K.: *Machine Learning: A Probabilistic Perspective*. MIT Press, 201w.
- [33] Müller-Wittig, W.: *Feature engineering for machine learning and data mining*. Springer, 2018.
- [34] Research, American Institutes for: *Early warning systems: A review of the evidence*. Washington, DC: U.S. Department of Education, Office of Planning, Evaluation and Policy Development, 2013.
- [35] Rezazade Mehrizi, M.H., van Ooijen P. Homan M.: *Applications of artificial intelligence (AI) in diagnostic radiology: a technography study*. *Eur Radiol*, 31:1805–1811, 2021. <https://doi.org/10.1007/s00330-020-07230-9>.
- [36] Romero, C., Ventura S.: *Data mining in education*. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(widm.1075Romero):12–27, 2013. <https://doi.org/10.1002/widm.1075Romero>.
- [37] Romero, C., Ventura S.: *Educational data mining: A review of the state of the art*. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 43(6):1630–1647, 2013.
- [38] Romero, C., Ventura S.: *Educational data mining: A review of the state of the art*. *IEEE Transactions on Systems, Man, and Cybernetics*. Part C: Applications and Reviews, 2013.
- [39] Romero, C., Ventura S.: *Guest Editorial: Special Issue on Early Prediction and Supporting of Learning Performance*. *IEEE Transactions on Learning Technologies*, 12(2)(TLT.2019.290810):145–147, 2019. <https://doi.org/10.1109/TLT.2019.290810>.
- [40] Romero, C., Ventura S.: *Educational data mining and learning analytics: An updated survey*. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(3)(widm.1355):e1355, 2020. <https://doi.org/10.1002/widm.1355>.
- [41] S., Narkhede: *Understanding Confusion Matrix*, 2018. <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>.
- [42] Siemens, G.: *Learning analytics: The emergence of a discipline*. *American Behavioral Scientist*, 55(4):388–400, 2012.
- [43] Siemens, G.: *Learning Analytics: The Emergence of a Discipline*. *American Behavioral Scientist*, 57(10)(000276421349885):1380–1400, 2013. <https://doi.org/10.1177/000276421349885>.
- [44] Sukanya, Bag: *Federated Learning – A Beginners Guide*, 2023. <https://www.analyticsvidhya.com/blog/2021/05/federated-learning-a-beginners-guide/>.
- [45] Tahiru, F.: *AI in Education: A Systematic Literature Review*. *Journal of Cases on Information Technology (JCIT)*, 23:1–20, 2021. <http://doi.org/10.4018/JCIT.2021010101>.
- [46] Yu, L. C., Lee C. W. Pan H. I. Chou C. Y. Chao P. Y. Chen Z. H. Tseng S. F. Chan C. L. Lai K. R.: *Improving early prediction of academic failure using sentiment analysis on self-evaluated comments*. *Computer Assisted Learning*, 34(jcal.12247):358–365, 2018. <https://doi.org/10.1111/jcal.12247>.