
Proyecto: Predicción del Rendimiento en Matemáticas de Estudiantes mediante Aprendizaje de Máquina

John Alexander Acevedo Serna – jaacevedos@eafit.edu.co

José Manuel Camargo Hoyos – jmcamargoh@eafit.edu.co

Santiago Rodríguez Duque – srodrigu16@eafit.edu.co

EAFIT

Resumen

Este proyecto aplica un proceso completo de aprendizaje automático (Machine Learning) siguiendo la metodología CRISP-DM para abordar un problema educativo: predecir si un estudiante aprobará el examen de matemáticas a partir de características demográficas y académicas. El conjunto de datos utilizado (*Students Performance in Exams*) contiene información de 1000 estudiantes, incluyendo género, nivel educativo de los padres, tipo de almuerzo y preparación previa para exámenes, junto con sus calificaciones en matemáticas, lectura y escritura.

El trabajo inicia con la comprensión del negocio, identificando la necesidad de anticipar el riesgo académico para facilitar intervenciones tempranas que mejoren el rendimiento y reduzcan la deserción escolar. Posteriormente se realiza un análisis exploratorio de datos (EDA) para evaluar la calidad de la información, descubrir patrones y relaciones entre variables, y construir una “data card” que describe el conjunto de datos y posibles sesgos.

1. Introducción

En el ámbito educativo, anticipar el rendimiento de los estudiantes puede ayudar a diseñar estrategias de apoyo y mejorar los resultados académicos. El dataset “*Students Performance in Exams*” provee información demográfica y académica (género, nivel educativo de los padres, tipo de almuerzo, preparación previa, entre otros) junto con las calificaciones en matemáticas, lectura y escritura.

En este proyecto se busca desarrollar un modelo de aprendizaje automático capaz de predecir si un estudiante aprobará matemáticas, identificando factores asociados al éxito o fracaso. Este tipo de análisis permite tomar decisiones preventivas, como asignar tutorías o recursos adicionales a estudiantes en riesgo.

2. Objetivos

2.1. Objetivo General

Desarrollar un modelo de machine learning que permita predecir si un estudiante aprobará el

examen de matemáticas a partir de variables demográficas y académicas (género, nivel educativo de los padres, tipo de almuerzo, preparación previa, entre otras), con el fin de identificar tempranamente a estudiantes en riesgo académico y apoyar la toma de decisiones en intervenciones educativas.

2.2. Objetivos Específicos

- Analizar y caracterizar el conjunto de datos Students Performance in Exams, identificando la calidad de la información, variables relevantes y posibles sesgos.
- Definir y preparar la variable objeto `pass_math`, que indica si un estudiante aprueba o no matemáticas.
- Realizar un análisis exploratorio de datos (EDA) que permita entender patrones, correlaciones y factores asociados al desempeño.
- Construir un modelo baseline que sirva como punto de referencia inicial.
- Establecer métricas de evaluación adecuadas para clasificación binaria, considerando posibles desbalances de clases.

3. Trabajos Relacionados

Cortez & Silva (2008) - Using Data Mining to Predict Secondary School Student Performance

Este trabajo es uno de los primeros referentes académicos sobre predicción de rendimiento escolar. Los autores recopilaban datos de estudiantes de educación secundaria en Portugal, incluyendo factores como características demográficas, sociales y escolares (edad, sexo, consumo de alcohol, asistencia, entre otros). Aplicaron técnicas de minería de datos y aprendizaje automático como árboles de decisión, redes neuronales y regresión

logística para predecir calificaciones finales. Un hallazgo clave fue que variables relacionadas con el entorno familiar y hábitos de estudio tienen fuerte impacto en el desempeño. Este artículo es relevante porque valida el enfoque de usar datos no académicos inmediatos para anticipar el rendimiento.

Dey et al. (2015) - Predicting Students' Performance using Advanced Learning Analytics

En este estudio se aplicaron técnicas de analítica de aprendizaje avanzada (Learning Analytics) para pronosticar el desempeño estudiantil. Los autores integraron datos demográficos, académicos y de participación en cursos en línea para entrenar modelos como máquinas de soporte vectorial (SVM) y árboles de decisión. Resaltan la importancia de seleccionar métricas apropiadas cuando existe desbalance de clases (ej. estudiantes que reprueban son minoría), recomendando métricas como F1-score y ROC-AUC en lugar de solo Accuracy.

Sahai et al. (2020) - Performance Prediction of Students Using Machine Learning Techniques

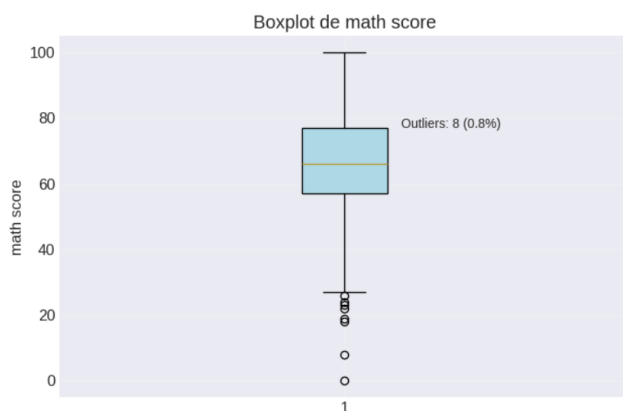
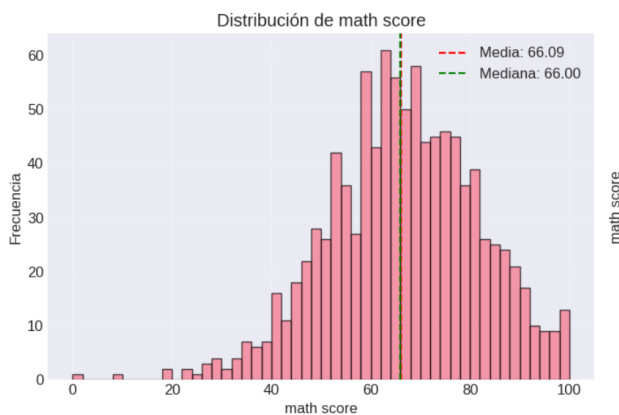
Este trabajo compara varios algoritmos de clasificación (KNN, Random Forest, XGBoost, SVM) para predecir el desempeño de estudiantes, mostrando que los métodos de ensamble logran mayor estabilidad y capacidad de generalización. Además, destacan la importancia de que las instituciones entiendan cómo el modelo llega a sus predicciones; para esto recomiendan herramientas de interpretabilidad, como el análisis de importancia de variables (qué factores pesan más en la predicción) y métodos como SHAP. Esto es relevante para nuestro proyecto porque no solo buscamos un modelo con buen desempeño, sino uno que pueda ser explicado a profesores o coordinadores para apoyar decisiones pedagógicas.

4. Datos y EDA

4.1. Descripción y Calidad de los Datos

El conjunto de datos contiene información de 1000 estudiantes con 8 variables, clasificadas principalmente como categóricas y tres variables numéricas de puntaje.

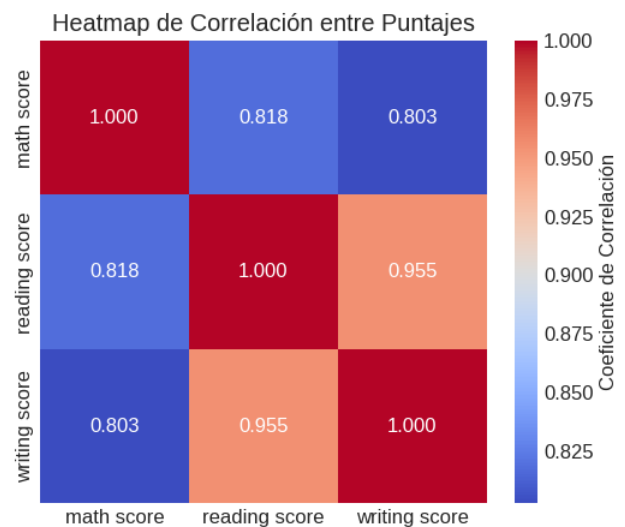
Distribución de la Variable Objetivo (math score): La distribución de la nota de matemáticas es casi simétrica, con la media y la mediana centradas en un valor similar (aproximadamente 67 puntos). Se identificó una presencia muy baja de outliers (0.8%), concentrados en el rango de notas más bajas.



4.2. Hallazgos Clave del Análisis Bivariado

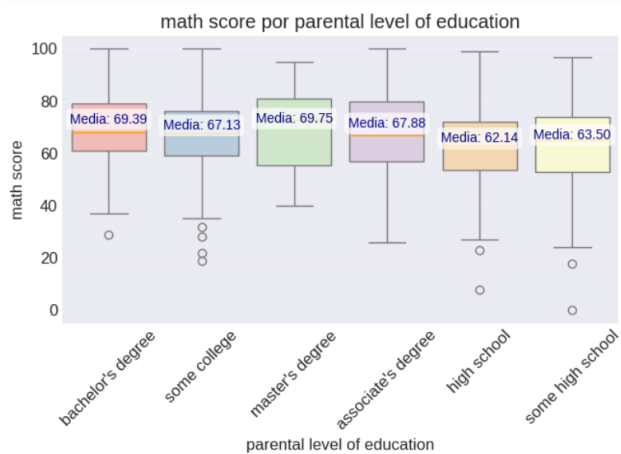
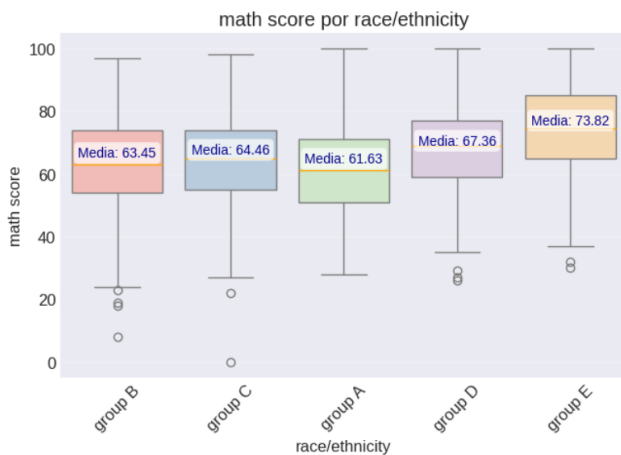
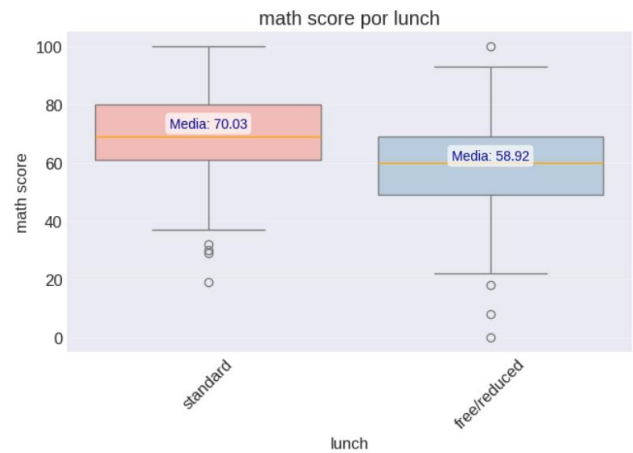
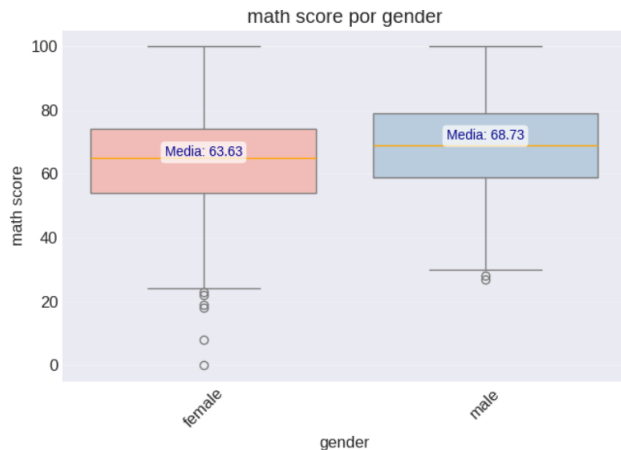
El análisis bivariado se centró en identificar qué variables categóricas introducen una variación significativa en la nota de matemáticas y qué variables numéricas son redundantes.

Al analizar la relación entre las variables de rendimiento, se encontró que el puntaje de matemáticas presenta una correlación extremadamente alta (por encima de 0.8) con los resultados de las otras competencias. Esta alta correlación es un indicio de multicolinealidad.



Para el objetivo de predecir el riesgo académico, esta relación no es representativa de la causa del problema, sino del efecto (habilidad académica general), volviendo a las notas unas variables redundantes. Por lo tanto, se justifica descartar estas variables en la fase de preparación de datos, enfocando el modelado en predictores demográficos y comportamentales que sí ofrezcan una relación real con la nota final.

A diferencia de los puntajes académicos redundantes, el análisis de las variables categóricas (demográficas y comportamentales) reveló diferencias estadísticamente significativas en la nota de matemáticas, lo que las convierte en predictores cruciales.

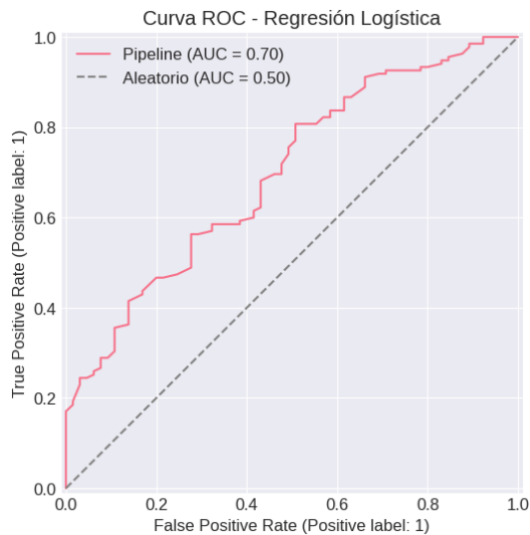


5. MODELOS Y RESULTADOS

El primer modelo implementado fue una **Regresión Logística** con sus hiperparámetros por defecto. Este modelo sirve como punto de referencia para medir el impacto de optimizaciones futuras.

Tras completar la fase de entrenamiento, el *pipeline* del modelo fue sometido a una evaluación rigurosa. Se utilizó un conjunto de datos de prueba (o *test set*) independiente, compuesto por 200 observaciones no vistas previamente por el modelo, para estimar su capacidad de generalización y rendimiento en un escenario productivo.

Para evaluar la capacidad de discriminación del modelo, es decir, su habilidad para asignar una probabilidad más alta a los casos positivos que a los negativos. Se calculó el **estadístico AUC (Área Bajo la Curva ROC)**. El modelo alcanzó un valor AUC de **0.69**. Este resultado confirma la capacidad de ordenamiento del modelo como moderada, indicando que, si bien es significativamente mejor que una clasificación aleatoria (AUC de 0.5), se encuentra lejos de una discriminación perfecta (AUC de 1.0).



Un análisis más detallado del rendimiento por clase, visible en el reporte de clasificación y la matriz de confusión, muestra un comportamiento asimétrico:

Classification report:				
	precision	recall	f1-score	support
0	0.595	0.338	0.431	65
1	0.736	0.889	0.805	135
accuracy			0.710	200
macro avg	0.665	0.614	0.618	200
weighted avg	0.690	0.710	0.684	200

Matriz de Confusión

	Pred: No Reprobado	Pred: Reprobado
Real: No Reprobado	22	43
Real: Reprobado	15	120

El modelo exhibe una alta sensibilidad para encontrar los reprobados, alcanzando un **89%**. Esto significa que el *pipeline* es muy eficaz para identificar la clase de interés (los casos que "sí reprueban"), encontrando correctamente a 89 de cada 100 instancias positivas reales.

En contraparte, el modelo muestra una debilidad notable en la especificidad, la cual se situó en un **43%**. Esta métrica indica que el modelo solo logró identificar correctamente al 43% de los casos negativos (aquellos que "no reprueban").

La estrategia de modelado se centrará en variables categóricas, como *gender* y *test preparation course*, las cuales demuestran una influencia estadísticamente significativa y accionable sobre el rendimiento. Esto asegura que el modelo se enfoque en factores que explican la causa del riesgo y faciliten la intervención para alcanzar la meta de aprobación.

Optimización de Hiperparámetros

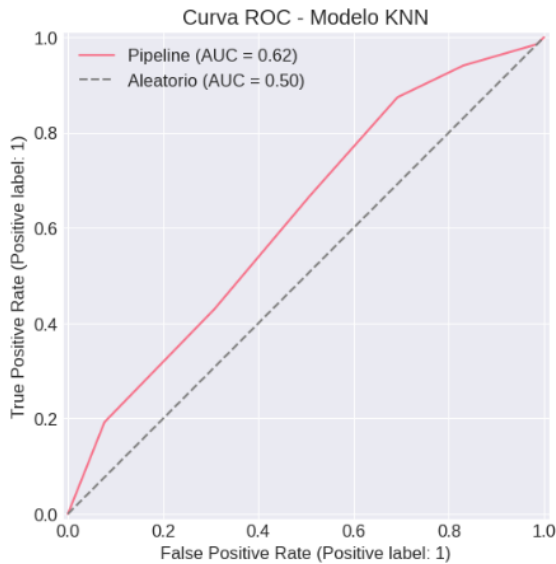
Como segundo enfoque, se evaluó un modelo no paramétrico de K-Nearest Neighbors (KNN). A diferencia del modelo base de Regresión Logística, este algoritmo requiere la sintonización de hiperparámetros clave.

Se implementó una búsqueda de cuadrícula (*GridSearch*) sobre el conjunto de entrenamiento, utilizando validación cruzada interna (CV) con el objetivo de maximizar la métrica F1-Score. La búsqueda determinó que la configuración óptima para este problema consiste en un valor de $k=7$ vecinos (parámetro *n_neighbors*) y una ponderación uniforme (*weights='uniform'*).

El pipeline optimizado de KNN fue entonces evaluado sobre el mismo conjunto de prueba (o test set) de 200 observaciones.

El modelo obtuvo una Exactitud (Accuracy) global del 69%. Este rendimiento es ligeramente inferior al 71% obtenido por el modelo base de Regresión Logística.

La capacidad de discriminación del modelo, medida por el AUC-ROC, fue de 0.62. Este valor confirma la observación inicial de que la capacidad de ordenamiento del modelo es limitada, y resulta inferior al 0.69 alcanzado por la Regresión Logística.



Un análisis más profundo del desempeño por clase, visible en el reporte de clasificación y la matriz de confusión, revela los siguientes puntos:

KNN Classification report:				
	precision	recall	f1-score	support
0	0.541	0.308	0.392	65
1	0.724	0.874	0.792	135
accuracy			0.690	200
macro avg	0.632	0.591	0.592	200
weighted avg	0.664	0.690	0.662	200

	Pred: (No Reprobado)	Pred: (Reprobado)
Real: (No Reprobado)	20	45
Real: (Reprobado)	17	118

El modelo KNN optimizado retiene una alta capacidad para detectar la clase positiva, identificando correctamente al 87.4% (redondeado a 87%) de los casos que "sí reprueban" (clase 1).

El modelo muestra una debilidad notable para identificar a los que "no reprueban" (clase 0). El f1-score para esta clase fue de solo 0.392 (o 39%), y su Especificidad (el recall de la clase 0) fue aún más baja, situándose en 0.308 (o 31%). Esto confirma que el modelo tiende a clasificar

erróneamente a muchos casos negativos como positivos.

Finalmente, se evaluó un modelo de Support Vector Machine (SVM) utilizando un kernel no lineal (Radial Basis Function, RBF), con el objetivo de capturar relaciones más complejas en los datos.

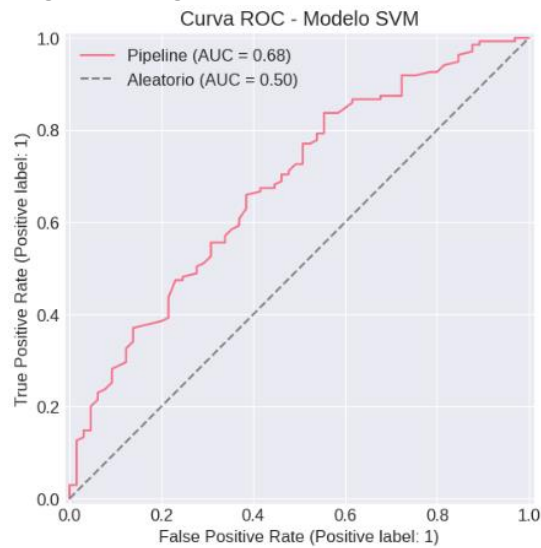
Se ejecutó una búsqueda de cuadrícula (GridSearch) para optimizar los hiperparámetros del pipeline, específicamente `model__C` (parámetro de regularización) y `model__gamma` (coeficiente del kernel). La búsqueda arrojó como parámetros óptimos una configuración de `C=0.1` y `gamma=0.01`, maximizando el F1-Score en la validación cruzada (CV F1-Score: 0.787).

La evaluación del pipeline optimizado sobre el conjunto de prueba de 200 observaciones arrojó resultados mixtos que deben ser analizados en dos partes: el rendimiento de las probabilidades y el de la clasificación final.

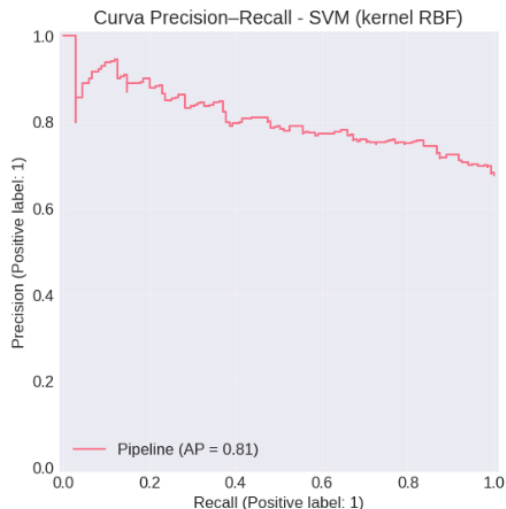
Para evaluar la capacidad de discriminación del modelo (su habilidad para asignar un score de riesgo más alto a los casos positivos), se analizaron la Curva ROC y la Curva Precision-Recall.

El modelo obtuvo un AUC-ROC de 0.68, indicando una capacidad de ordenamiento moderada, que se alinea con los resultados de la

Regresión Logística.



De forma complementaria, la **Curva Precision-Recall (PR)**, que es más robusta ante el desbalanceo de clases, muestra un **Average Precision (AP) de 0.81**. Esto sugiere que las probabilidades generadas por el modelo son, de hecho, bastante informativas.



A pesar del rendimiento aceptable de las probabilidades, al aplicar el umbral de decisión por defecto (0.5), el modelo sufre un colapso total en la clasificación.

Como se observa en el reporte de clasificación y la matriz de confusión, el modelo predice la clase positiva ("Repr") para el 100% de las 200

observaciones, ignorando por completo a la clase negativa.

SVM Classification report:

	precision	recall	f1-score	support
0	0.000	0.000	0.000	65
1	0.675	1.000	0.806	135
accuracy			0.675	200
macro avg	0.338	0.500	0.403	200
weighted avg	0.456	0.675	0.544	200

	Pred: (No Repr)	Pred: (Repr)
Real: (No Repr)	0	65
Real: (Repr)	0	135

El modelo es incapaz de identificar un solo caso negativo. Su **Especificidad (Recall de la clase 0) es del 0.0%**.

Esto produce una **Sensibilidad (Recall de la clase 1) artificialmente perfecta del 100.0%**, ya que acierta todos los casos "Repr" simplemente porque predice esta clase para todo el conjunto.

6. Modelado - Entrega 3

En la tercera fase del proyecto, el equipo se concentró en construir, optimizar y evaluar diversos modelos de machine learning para predecir la aprobación en matemáticas. Un paso metodológico clave fue el uso de pipelines que incluían StandardScaler para la normalización de datos en todos los entrenamientos.

La evaluación del rendimiento se basó en tres métricas fundamentales: **Accuracy**, para el porcentaje total de aciertos; **F1-Score**, crucial debido al desbalanceo de los datos (la clase "Reprobado" representa solo el 32.3%); y **ROC-AUC**, para medir la capacidad de discriminación de los modelos.

XGBoost y Stacking

Se continuó la evaluación con modelos de ensamble. Se probaron dos configuraciones de

XGBoost (Modelo A y B) y un StackingClassifier que combinaba la Regresión Logística, KNN y XGBoost como estimadores bases, usando una Regresión Logística como meta-modelo.

[Modelo A] Entrenando con n_estimators=1500, max_depth=5...

--- Resultados Modelo A ---

Test Accuracy: 0.635

Test F1 : 0.6970954356846473

Test ROC-AUC : 0.6923646723646724

	precision	recall	f1-score	support
0	0.457	0.662	0.541	65
1	0.792	0.622	0.697	135
accuracy			0.635	200
macro avg	0.625	0.642	0.619	200
weighted avg	0.684	0.635	0.646	200

Matriz de Confusión (Modelo A):

	Pred: (No Reprobado)	Pred: (Reprobado)
Real: (No Reprobado)	43	22
Real: (Reprobado)	51	84

[Modelo B] Entrenando con n_estimators=1300, max_depth=7...

--- Resultados Modelo B ---

Test Accuracy: 0.63

Test F1 : 0.6991869918699187

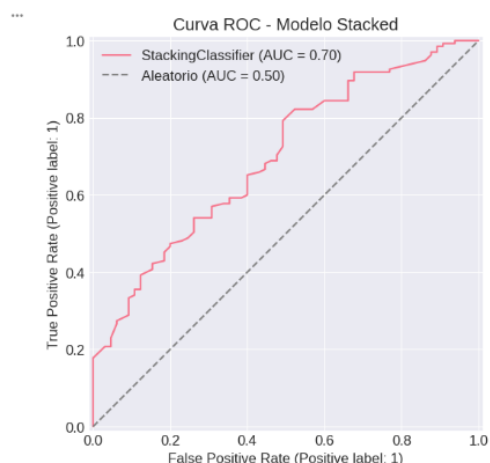
Test ROC-AUC : 0.6945299145299145

	precision	recall	f1-score	support
0	0.449	0.615	0.519	65
1	0.775	0.637	0.699	135
accuracy			0.630	200
macro avg	0.612	0.626	0.609	200
weighted avg	0.669	0.630	0.641	200

Matriz de Confusión (Modelo A):

	Pred: (No Reprobado)	Pred: (Reprobado)
Real: (No Reprobado)	40	25
Real: (Reprobado)	49	86

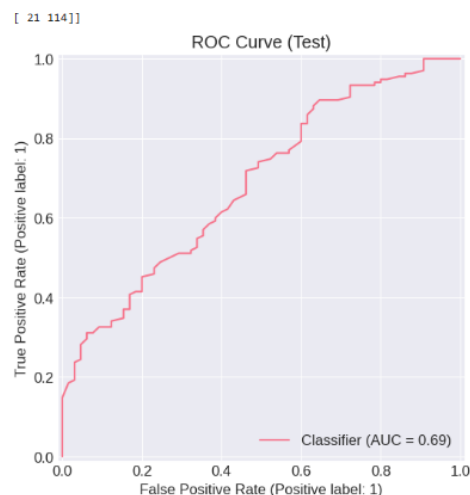
Los resultados del XGBoost A en el Test Set mostraron un Accuracy de 0.635, un F1-Score de 0.697 y un ROC-AUC de 0.692. Por su parte, el XGBoost B obtuvo un Accuracy de 0.620, un F1-Score de 0.686 y un ROC-AUC de 0.695. El modelo de Stacking alcanzó un Accuracy de 0.675, un F1-Score de 0.806 y un ROC-AUC de 0.702.



El análisis reveló un fallo del modelo. Los modelos XGBoost base (A y B) mostraron un rendimiento inferior a los modelos más simples. Más importante aún, el StackingClassifier sufrió el mismo colapso que el SVM. Su matriz de confusión, $\begin{bmatrix} 0 & 65 \\ 0 & 135 \end{bmatrix}$, revela que predijo 'Aprobado' (clase 1) para el 100% de las muestras del conjunto de prueba, lo que lo convierte en un modelo completamente inútil para la tarea.

AutoML

Como paso final, se utilizó la biblioteca FLAML para buscar automáticamente el mejor modelo y los mejores hiperparámetros, dándole un presupuesto de tiempo de 120 segundos



El mejor estimador encontrado fue 'xgboost', con un mejor Accuracy en validación de 0.7125. Los resultados en el Test Set fueron un Accuracy de 0.690 y un F1-Score para la Clase 1 de 0.79.

El análisis indica que FLAML identificó un modelo xgboost que, a diferencia de los probados manualmente, sí logró un rendimiento competitivo y balanceado. La matriz de confusión $\begin{bmatrix} 25 & 40 \\ 21 & 114 \end{bmatrix}$ muestra que es el modelo que mejor identificó a la clase minoritaria 'Reprobado' (clase 0), clasificando correctamente a 25 estudiantes y superando a la Regresión Logística (22) y al KNN (20). Aunque su Accuracy (0.690) es ligeramente inferior al de la Regresión Logística (0.710), su equilibrio lo perfila como uno de los candidatos más viables.

7. Conclusiones

En cuanto a los modelos viables, de todos los probados, solo la Regresión Logística y el XGBoost (optimizado por FLAML) demostraron serlo. La Regresión Logística obtuvo el mejor Accuracy (0.710), mientras que el modelo de FLAML (Accuracy 0.690) mostró el mejor equilibrio para identificar a la clase minoritaria ('Reprobado').

Respecto a los modelos fallidos, los modelos Support Vector Machine (SVM) y StackingClassifier colapsaron. Ambos aprendieron a predecir únicamente la clase mayoritaria ('Aprobado') y fallaron en identificar a un solo estudiante que 'Reprueba' en el conjunto de prueba, lo que los descarta por completo.

Se evidenció un incumplimiento del KPI. El objetivo de negocio era un Recall superior a 0.85 para la clase "Reprobado" (identificar al 85% de los que reprueban), pero ningún modelo se acercó. El mejor fue el XGBoost-FLAML con un Recall de 0.385 para esa clase (calculado como $25 / (25 + 40)$). Esto indica que, con la configuración actual, los modelos son muy malos para detectar a los estudiantes en riesgo.

Finalmente, se concluyó que existe un poder predictivo limitado. El hecho de que ningún modelo (ni siquiera los optimizados por AutoML) superara el 71% de Accuracy sugiere que las variables demográficas y contextuales (sin usar las notas de lectura y escritura) tienen un poder predictivo limitado para esta tarea.

8. Enlace al Repositorio

Incrementar el tiempo de búsqueda de AutoML (time_budget) si el ambiente lo permite. Esto podría permitir explorar más configuraciones y modelos adicionales, aumentando la posibilidad de encontrar modelos superiores.

Dado que el objetivo es alcanzar un Recall mayor a 0.85 (identificar al 85% o más de los estudiantes que reprobarán), se debe realizar una afinación explícita del umbral de probabilidad de los modelos (la probabilidad por encima de la cual se predice "Reprueba").

Explorar técnicas de optimización de hiperparámetros más exhaustivas, como Optuna o GridSearch, para validar si FLAML ya alcanzó un óptimo o si hay espacio para mejoras adicionales.

Planificar el despliegue del modelo ganador (posiblemente XGBoost o Regresión Logística, dependiendo de la tolerancia al riesgo) en un entorno de producción (como un microservicio o una herramienta interna) y establecer un monitoreo de rendimiento.

Documentar claramente cada etapa del flujo de trabajo, pues esto facilita la reproducibilidad y la entrega formal del proyecto.

[4829_Academic_Performance_Prediction_Using_Machine_Learning_Approaches_A_Survey](#)

9. Enlace al Repositorio

<https://github.com/JoseCamargo10/machinelearning-project.git>

10. Bibliografía

Cortez, P., & Silva, A. M. G. (2008). *Using data mining to predict secondary school student performance*.

https://www.researchgate.net/publication/228780408_Using_data_mining_to_predict_secondary_school_student_performance

Dey, et al. (2015). *Predicting Students' Performance using Advanced Learning Analytics*.
https://www.researchgate.net/publication/315837527_Predicting_Student_Performance_using_Advanced_Learning_Analytics

Sahai, et al. (2020). *Performance Prediction of Students Using Machine Learning Techniques*.
<https://www.researchgate.net/publication/39015>