

Machine Learning and Vibe Coding Applied to Predictive Academic Analytics

Carrizo Claudio, Javier Saldarini, Raúl Navarro Peláez, Daiana Viotti
Facultad Regional San Francisco – Universidad Tecnológica Nacional
Av. de la Universidad 501 - San Francisco - Córdoba - Tel. 03564-421147
{cjcarrizo77, saldarinijavier, ingrnavarropelaez, dayviotti2015}@gmail.com

Abstract. University dropout represents one of the main challenges in higher education, especially in engineering careers, where the first year is a critical stage. In the Information Systems Engineering program at a Public University, high rates of poor performance were detected in partial evaluations of subjects in the programming area, which frequently leads to abandonment of the program. Given this problem, a process of predictive academic analytics is developed aimed at anticipating student performance in instances of partial evaluation. The present work addresses the comparative study of six supervised learning algorithms (Decision Tree, Random Forest, Gradient Boosting, k-Nearest Neighbors, Support Vector Machine, Naive Bayes), applied on an institutional dataset. As a methodological innovation, the Vibe Coding paradigm is incorporated, using Claude Code (claude-sonnet-4-6) as a Generative Artificial Intelligence tool to assist coding in Python. The benchmarking uses performance metrics such as accuracy, precision, recall, F1-Score and confusion matrix. The expected benefits include the selection of the most appropriate predictive model, the generation of early alerts for teachers and academic managers, and the reduction of the dropout rate through timely pedagogical interventions.

Keywords: *predictive academic analytics, machine learning, vibe coding, Claude Code*

Aprendizaje Automático y Vibe Coding aplicados a la Analítica Académica Predictiva

Resumen. La deserción universitaria representa uno de los principales desafíos en la educación superior, especialmente en carreras de ingeniería, donde el primer año es una etapa crítica. En la carrera de Ingeniería en Sistemas de Información de una Universidad Pública, se detectaron altas tasas de bajo rendimiento en evaluaciones parciales de materias del área de programación, lo que frecuentemente deriva en el abandono de la carrera. Ante esta problemática, se desarrolla un proceso de analítica académica predictiva orientado a anticipar el desempeño estudiantil en instancias de evaluación parcial. El presente trabajo aborda el estudio comparativo de seis algoritmos de aprendizaje supervisado (Árbol de Decisión, Random Forest, Gradient Boosting, k-Nearest Neighbors, Support Vector Machine, Naive Bayes), aplicados sobre un dataset institucional. Como innovación metodológica, se incorpora el paradigma Vibe Coding, utilizando Claude Code (claude-sonnet-4-6) como herramienta de Inteligencia Artificial Generativa para asistir la codificación en Python. La evaluación comparativa emplea métricas de desempeño como exactitud, precisión, recall, F1-Score y matriz de confusión. Los beneficios esperados incluyen la selección del modelo predictivo más adecuado, la generación de alertas tempranas para docentes y gestores académicos, y la reducción de la tasa de deserción mediante intervenciones pedagógicas oportunas.

Palabras clave: *analítica académica predictiva, aprendizaje automático, vibe coding, Claude Code*

1. Introducción

El fracaso académico en el primer año de las carreras universitarias de ingeniería constituye uno de los fenómenos más estudiados en el ámbito de la educación superior. En particular, las materias de programación del ciclo inicial presentan elevadas tasas de reprobación y abandono, situación que afecta tanto la trayectoria individual de los estudiantes como la eficiencia institucional (García-Martínez et al., 2023). En la carrera de Ingeniería en Sistemas de Información (ISI) de una universidad pública argentina, se han identificado patrones recurrentes de bajo rendimiento en el primer parcial de materias como Algoritmos y Estructuras de Datos (AED), Matemática Discreta y Lógica y Estructuras Discretas, lo que motiva el desarrollo de estrategias de intervención temprana.

La analítica académica predictiva surge como una respuesta tecnológica a esta problemática, combinando técnicas de aprendizaje automático (machine learning), con datos institucionales para anticipar el desempeño estudiantil antes de que se produzcan situaciones de riesgo (Aldowah et al., 2019). Diversas investigaciones han demostrado que modelos basados en algoritmos supervisados pueden alcanzar niveles de precisión aceptables para clasificar a los estudiantes según su rendimiento esperado, permitiendo la activación de mecanismos de alerta temprana y apoyo pedagógico (Waheed et al., 2020).

En paralelo, el surgimiento del paradigma Vibe Coding (Muñoz et al., 2026), entendido como la práctica de desarrollar software de manera asistida por modelos de lenguaje de gran escala (LLM) mediante instrucciones en lenguaje natural, ha transformado la forma en que los investigadores interactúan con herramientas de codificación (Becker et al., 2023). En este trabajo se hace uso de Claude Code (Anthropic, 2025), basado en el modelo claude-sonnet-4-6 de Anthropic, como entorno de desarrollo asistido por IA para automatizar y agilizar las etapas del proceso de analítica predictiva.

El objetivo principal de este trabajo es realizar un estudio comparativo de seis algoritmos de aprendizaje supervisado aplicados a un dataset institucional, con el fin de identificar el modelo predictivo más adecuado para anticipar el desempeño estudiantil en materias de programación de primer nivel. La variable dependiente denominada RENDIMIENTO adopta tres categorías: BAJO (nota inferior a 4), MEDIO (nota 4 ó 5) y ALTO (nota mayor o igual a 6). El proceso de modelado es asistido íntegramente mediante Vibe Coding con Claude Code, integrando así una innovación metodológica en el campo de la analítica educativa.

Este artículo se enmarca en una línea de investigación que articula el análisis de datos académicos con técnicas de inteligencia artificial aplicada a la gestión institucional. Los resultados esperados contribuirán a la toma de decisiones en materia de política académica y a la implementación de sistemas de alerta temprana en el contexto de universidades públicas argentinas.

2. Proceso de Analítica Predictiva Asistido por Vibe Coding

El proceso de analítica predictiva se llevó a cabo siguiendo una metodología estructurada en catorce etapas, donde cada una fue asistida mediante interacciones en lenguaje natural con Claude Code (claude-sonnet-4-6). A continuación, se describe brevemente cada etapa:

2.1. Definición del problema

Se delimitó el objetivo predictivo, el cual consistió en clasificar el rendimiento de los estudiantes en el primer parcial de las materias AED, Matemática Discreta y Lógica y Estructuras Discretas (planes 2008 y 2023) en tres categorías (BAJO, MEDIO, ALTO). Se identificaron los actores involucrados, los requerimientos institucionales y las variables de interés. Mediante prompts en lenguaje natural, Claude Code asistió en la formalización del problema como una tarea de clasificación multiclase supervisada.

2.2. Recolección e integración de datos

El dataset fue construido a partir de registros académicos y administrativos de la universidad. Se integraron fuentes heterogéneas que incluyen datos personales, geográficos y académicos, obteniendo un conjunto de 2335 registros y 33 atributos. Claude Code asistió en la escritura de scripts de extracción y consolidación de datos desde múltiples fuentes en formato tabular.

2.3. Comprensión inicial de datos

Se realizó un análisis descriptivo del dataset para comprender la naturaleza, distribución y calidad de las variables disponibles. Con la asistencia de Claude Code se generaron resúmenes estadísticos, análisis de frecuencias y visualizaciones iniciales que permitieron identificar variables candidatas y posibles anomalías en los datos.

2.4. Limpieza y preparación

Se detectaron y trataron valores faltantes, outliers y registros inconsistentes. Claude Code fue utilizado para generar código de limpieza, imputación de valores ausentes con estrategias estadísticas y estandarización de categorías. Esta etapa garantizó la integridad y consistencia del dataset previo al modelado.

2.5. Análisis exploratorio

Se profundizó el análisis de las relaciones entre variables predictoras y la variable objetivo RENDIMIENTO. A través de Vibe Coding se generaron visualizaciones de distribuciones, correlaciones y análisis bivariados. Esta etapa permitió identificar patrones preliminares y fundamentar las decisiones de ingeniería de características.

2.6. Ingeniería de características

Se transformaron, codificaron y seleccionaron atributos relevantes para maximizar la capacidad predictiva de los modelos. Con apoyo de Claude Code se automatizó la codificación de variables categóricas (one-hot encoding, label encoding), la creación de variables derivadas y la selección de características mediante análisis de importancia.

2.7. División train/validation/test

El dataset fue particionado en conjuntos de entrenamiento (70%), validación (15%) y prueba (15%), garantizando la representatividad proporcional de las clases mediante estratificación. Claude Code generó el código correspondiente y verificó el balance de clases en cada subconjunto.

2.8. Pre-procesamiento integrado

Se construyeron pipelines de pre-procesamiento que integran la normalización, escalado y transformación de variables de manera reproducible. El uso de Claude Code permitió diseñar pipelines con scikit-learn que garantizan la consistencia entre los conjuntos de entrenamiento y evaluación, evitando fugas de datos.

2.9. Selección de algoritmos

Se seleccionaron seis algoritmos de aprendizaje supervisado representativos de distintas familias metodológicas: Árbol de Decisión, Random Forest, Gradient Boosting, k-Nearest Neighbors (k-NN), Support Vector Machine (SVM) y Naive Bayes. La selección fue fundamentada con asistencia de Claude Code, que sugirió la combinación de algoritmos lineales, basados en árboles, instancias y redes para garantizar diversidad en el benchmark.

2.10. Entrenamiento

Cada algoritmo fue entrenado sobre el conjunto de entrenamiento utilizando los hiperparámetros por defecto en primera instancia. Claude Code automatizó la generación del código de entrenamiento, la gestión de dependencias y el registro de los resultados en cada iteración, reduciendo significativamente los tiempos de implementación.

2.11 Ajuste de hiperparámetros.

Se aplicaron técnicas de optimización de hiperparámetros. Claude Code asistió en la definición de los espacios de búsqueda y en la interpretación de los resultados de optimización para cada algoritmo.

2.12. Evaluación

Los modelos entrenados fueron evaluados sobre el conjunto de prueba utilizando métricas de desempeño multiclase: exactitud (accuracy), precisión (precision), recall, F1-Score macro y matriz de confusión. Claude Code generó los reportes de clasificación y las visualizaciones comparativas de las métricas obtenidas.

2.13. Comparación de modelos

Se elaboró un cuadro comparativo de los seis modelos, ordenados según su desempeño en las métricas clave. Claude Code asistió en la generación de tablas de resultados, gráficos de barras comparativos y la identificación del modelo con mejor balance entre rendimiento predictivo y complejidad computacional.

2.14. Interpretación de resultados

Se analizaron los resultados en función del contexto educativo, identificando los atributos de mayor influencia sobre el rendimiento estudiantil mediante técnicas de importancia de características (feature importance). Claude Code facilitó la generación de visualizaciones interpretativas y la redacción de conclusiones basadas en los hallazgos del proceso de modelado.

3. Resultados Preliminares

Finalizado el proceso de analítica predictiva se obtuvieron algunos resultados respecto de la validación de los modelos donde se aplicaron los 6 algoritmos de aprendizaje supervisado. En la Fig. 1 a continuación, se pueden observar los valores obtenidos para las métricas de exactitud, precisión, recall y F1-Score.

Modelo	Accuracy	F1 Weighted	F1 Macro	Precision W.	Recall W.
decision_tree	0.4458	0.4393	0.4069	0.4342	0.4458
* random_forest	0.4667	0.4605	0.4265	0.4561	0.4667
gradient_boosting	0.4958	0.4632	0.4229	0.4554	0.4958
knn	0.4750	0.4438	0.4079	0.4423	0.4750
svm	0.5125	0.4413	0.3920	0.5480	0.5125
naive_bayes	0.3125	0.2771	0.2937	0.4713	0.3125

Fig. 1. Tabla Comparativa de resultados de métricas

Por otro lado, en la Fig. 2 se pueden observar los resultados arrojados por las matrices de confusión de los 6 algoritmos.

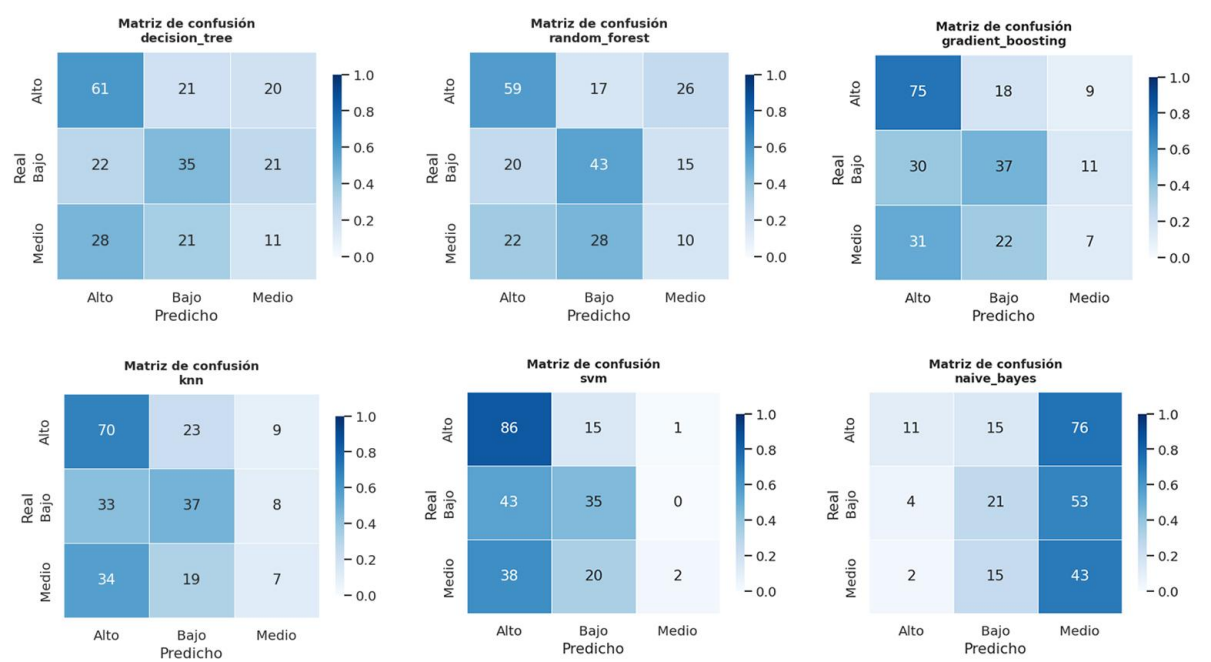


Fig. 2. Resultados de Matriz de Confusión

Las matrices de confusión muestran un patrón muy claro: todos los modelos tienen dificultades severas para clasificar la clase “Medio”.

Luego de realizar una interpretación de los resultados obtenidos, pudimos determinar que Random Forest emerge como la mejor alternativa porque ofrece el mejor equilibrio global, con el mayor F1 macro, un F1 weighted competitivo y una matriz de confusión más estable. Gradient Boosting queda muy cerca y podría considerarse una segunda opción seria, pero con mayor sesgo hacia la clase Alto.

Desde el punto de vista de la detección temprana del rendimiento estudiantil, el modelo seleccionado debe priorizar equilibrio, sensibilidad ante clases problemáticas y minimización de errores pedagógicamente costosos. Con la evidencia disponible, la elección técnicamente más defendible es Random Forest, aunque todavía requiere mejoras para lograr una detección más confiable de la clase Medio y una mejor generalización en test.

Finalmente, en relación con la incorporación de herramientas de generación de código asistidas por modelos de lenguaje (LLM) en el desarrollo del proceso de analítica, se observa una reducción significativa en los tiempos de construcción del pipeline, favoreciendo la automatización de tareas y la rápida iteración de modelos. No obstante, fue necesario realizar ajustes y validaciones manuales en distintas etapas del proceso, particularmente en la definición de pre-procesamientos, la correcta parametrización de los algoritmos y la interpretación de los resultados, lo que evidencia que este enfoque actúa como un facilitador del desarrollo, pero no reemplaza la supervisión experta.

4. Acciones Futuras

Como trabajo futuro, se prevé profundizar el análisis comparativo, incluyendo mejoras, como, por ejemplo:

- Mejorar la detección de la clase “Medio”.
- Ajustar hiperparámetros del modelo de Random Forest.
- Utilizar una validación más robusta, como, por ejemplo, validación cruzada estratificada k-fold.
- Incorporar al análisis otros algoritmos de aprendizaje supervisado, como, por ejemplo, Regresión Logística Multinomial y XGBoost, Redes Neuronales.
- Construir nuevas características que estén orientadas al dominio, para incrementar el espectro de atributos.
- Evaluar el uso de ensambles.
- Evaluar el uso de otras herramientas de desarrollo asistido por IA, como, por ejemplo, Antigravity.

Referencias

- Aldowah, H., Al-Samarraie, H., & Fauzy, W. M. (2019). Educational data mining and learning analytics for 21st century higher education: A review and synthesis. *Telematics and Informatics*, 37, 13–49. <https://doi.org/10.1016/j.tele.2019.01.007>
- Anthropic. (2025). Claude Sonnet 4.6 (modelo de lenguaje de gran escala) utilizado a través de Claude Code. <https://www.anthropic.com>
- Becker, B. A., Denny, P., Finnie-Ansley, J., Luxton-Reilly, A., Prather, J., & Santos, E. A. (2023). Programming is hard – or at least it used to be: Educational opportunities and challenges of AI code generation. *ACM SIGCSE Technical Symposium on Computer Science Education*, 500–506. <https://doi.org/10.1145/3545945.3569759>
- García-Martínez, I., Fernández-Batanero, J. M., Fernández Cerero, D., & León, S. P. (2023). Using artificial intelligence to improve the quality of higher education: A systematic review. **Education Sciences*, 13*(1), 6. <https://doi.org/10.3390/educsci13010006>
- Muñoz, A., & Rook, L. (2026). Leveraging generative AI through vibe coding: A case of simulation-based curriculum redesign in management education. **Education Sciences*, 16*(4), 558. <https://doi.org/10.3390/educsci16040558>.
- Waheed, H., Hassan, S. U., Aljohani, N. R., Hardman, J., Alelyani, S., & Nawaz, R. (2020). Predicting academic performance of students from VLE big data using deep learning models. *Computers in Human Behavior*, 104, 106189. <https://doi.org/10.1016/j.chb.2019.106189>