Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería Eléctrica
IEE3031 – Aprendizaje de máquina:
Técnicas y aplicaciones en series de tiempo y hardware

Proyecto Final

Análisis predictivo del riesgo de deserción estudiantil mediante un Sistema de Alerta Temprana al inicio de la carrera

Nombre: Ignacio Andrés Moraga Silva

1. Introducción

1.1. Contexto del proyecto

De acuerdo con [1] la deserción estudiantil en la educación superior puede ser definida como el abandono de un programa de este nivel de educación antes de obtener el título sin reincorporación inmediata. Constituye uno de los desafíos más críticos tanto para los estudiantes como para las Instituciones de Educación Superior (IES) a nivel global dado que no solo representa una pérdida económica que afecta a la estabilidad financiera de las IES, sino que también afecta el desarrollo profesional de los individuos y por tanto la prosperidad de los países al limitar la formación de capital humano calificado [2, 3].

En particular, la retención de primer año corresponde al porcentaje de personas que ingresan a una IES como estudiantes de primer año en el año t y que siguen matriculados en la misma institución como estudiantes de la misma cohorte en el año $t+1$, respecto del total de estudiantes de primer año que ingresaron en el año t [4], por tanto, la deserción correspondería al porcentaje complementario para lograr el 100 %. Esta lógica puede aplicarse para los años posteriores. Se ha constatado en [1, 5], que el primer año de educación superior es el periodo más crítico para la retención. Siendo esta etapa una de transición, este es el período de mayor vulnerabilidad para los estudiantes, ya sea por dificultades académicas, cambios de vivienda, lejanía con su núcleo familiar, nuevos gastos económicos, etc. En ese sentido, factores como el rendimiento académico previo (NEM y Ranking en Chile), las puntuaciones en pruebas de acceso estandarizadas (PAES en Chile o sus equivalentes internacionales) y las variables socioeconómicas juegan un rol determinante en la decisión de abandonar los estudios recientemente comenzados [5]. Por consiguiente, el primer año actúa como un "filtro", ya que los estudiantes que muestran señales de riesgo en esta etapa temprana tienen una alta probabilidad de no completar sus estudios en el futuro, independientemente de cuándo se formalice su deserción.

Ahora bien, la importancia de los factores antes mencionados varía en cada situación analizada y depende del contexto, pero la gran mayoría de los estudios, como [6], establecen que el rendimiento académico durante el primer semestre de carrera es el factor más determinante para predecir la deserción futura, superando a antecedentes socioeconómicos o demográficos una vez que el estudiante ya ha comenzado sus clases, sin embargo, esperar a tener esos datos implica perder la oportunidad de intervención temprana [1], lo cual valida el uso de variables menos predictivas pero disponibles "a priori", como las socioeconómicas. Este compromiso es fundamental en un Sistema de Alerta Temprana (SAT), al enfocarse en la antelación en la predicción.

A nivel internacional, países como España enfrentan situaciones donde uno de cada tres estudiantes abandona sin completar su grado, y en Estados Unidos aproximadamente el 30 % de los estudiantes de

primer año abandonan antes de comenzar su segundo año de estudios [2]. En el contexto chileno, por ejemplo, en general la actual tasa de retención de primer año de pregrado ha aumentado en relación con la del último año. Para la cohorte 2024, considerando todo tipo de IES (Centros de Formación Técnica, Institutos Profesionales y Universidades) y carrera fue de 77.3 %, cifra que representa el nivel más alto registrado desde la creación del Servicio de Información de Educación Superior en 2007 [4]. Esto da cuenta que cerca del 22.7 % de estudiantes desertan tras el primer año. En particular, dentro de carreras como Ingeniería Informática se han reportado tasas de deserción similares e incluso cifras drásticamente mayores hacia el tercer año, alrededor del 50 % [5]. Ello ratifica que este tipo de deserción es existente y significa una problemática real.

Es importante mencionar que no solo basta con estudiar las consecuencias de este fenómeno, sino que disponer de herramientas que permitan tomar decisiones informadas en cuanto a cómo abordarlo. En concreto, de las mayores complicaciones en la gestión de la retención es que las IES suelen reaccionar de manera tardía, basándose en el fracaso académico una vez que este ya ha ocurrido, por ejemplo, tras el primer semestre. Por tanto, es imperativo desarrollar un enfoque preventivo, y no reactivo, que permita perfilar el riesgo desde el momento de la matrícula. Esto faculta a las instituciones para activar protocolos de apoyo antes de que el estudiante caiga en un círculo vicioso de malos resultados y desmotivación.

1.2. Problema a resolver

Siguiendo lo descrito al final de 1.1, las IES enfrentan barreras no menores para desarrollar SATs para la deserción que sean realmente efectivos. Por un lado, existe dificultad en el acceso a datos privados debido a regulaciones de privacidad; información sensible de los estudiantes, como ingresos, situaciones de salud o dinámicas familiares, son normalmente privadas. Esta restricción hace imperativo maximizar la capacidad de inferencia con los datos disponibles, lo que requiere herramientas más sofisticadas que una simple estadística descriptiva, de tal modo de encontrar patrones subyacentes en información como la demografía y antecedentes académicos [2]. Adicionalmente, métodos "clásicos" estadísticos, como la regresión lineal, suelen depender de suposiciones estrictas sobre los datos, como relaciones lineales entre la variable independiente y dependiente e independencia en las observaciones [6].

Por consiguiente, modelos de Machine Learning (ML) que permiten explotar la inferencia con los datos existentes y no requieren estas suposiciones estrictas permiten modelar relaciones complejas y no lineales entre variables, por ejemplo entre las demográficas y las académicas, y así detectar patrones latentes de riesgo que no son evidentes y permitir perfilar al estudiante más allá de su simple promedio de notas. En este sentido, de acuerdo con [3, 7], se sugiere que los modelos predictivos que utilizan datos administrativos disponibles al momento de la matrícula (características "pre-entrada" como demografía y antecedentes escolares) tienen un valor significativo para superar el desconocimiento de la situación del estudiante por parte de la IES, y así permitir estimar una probabilidad de fracaso futuro o deserción (*target*) al usar exclusivamente la información del presente (matrícula), y desplegar estrategias de intervención oportunas [7].

De esta forma, estudios como [2] han demostrado la implementación de algoritmos de ML con alta precisión en este dominio. Concretamente, modelos basados en árboles de decisión (Decision Trees) como Random Forest y Gradient Boosting (XGBoost), así como Redes Neuronales (ANNs), han logrado predecir lo anterior. Por ejemplo, en este mismo estudio el uso de Gradient Boosting logró identificar correctamente a más del 72 % de los estudiantes que abandonarían sus estudios, incluso antes de que comenzaran las clases. No obstante, no hay que perder de vista que en ML no existe un modelo único universal, al dependerse mucho de los datos disponibles [1]. Esta variabilidad justifica la necesidad de comparar distintas arquitecturas para determinar cuál se ajusta mejor a la distribución de los datos de este problema.

Entonces, se plantea como problema la identificación de perfiles de riesgo de deserción en estudiantes de nuevo ingreso bajo condiciones de información limitada, de modo tal que se busca discriminar entre estudiantes que requieren apoyo inmediato y aquellos que no. Para ello se propone un enfoque preventivo mediante un modelo clasificador de ML que asigne una probabilidad a un estudiante sobre si abandonará o no sus estudios superiores, el que podrá ser usado como un SAT por sub-instituciones internas dentro de IES.

2. Metodología propuesta

En línea con lo visto en 1.2, se planteó el desarrollo de un modelo de ML de aprendizaje supervisado capaz de predecir si, en base a una probabilidad de "riesgo de deserción", un estudiante presenta un alto riesgo de abandonar sus estudios, esto, mediante el uso de datos disponibles al momento de su matrícula en la IES; por ejemplo, rendimiento académico previo, factores demográficos, socioeconómicos, etc. En ese sentido, como se buscaría categorizar a un estudiante como alguien que deserta/no deserta, la tarea principal de ML sería clasificación binaria, de modo tal de predecir, a partir de las características mencionadas si un estudiante particular pertenece a la clase 1 (desertor) o 0 (no desertor). Es decir, se busca detectar desde el momento de matrícula si el estudiante posee un perfil de riesgo asociado al abandono eventual de la carrera.

Dado que se buscaría lograr la anticipación del riesgo de deserción, el modelo permitiría disponer de una herramienta de decisión para que las IES logren identificar tempranamente a los estudiantes con mayor riesgo de abandono de sus estudios. Con ello, sub-instituciones que velan por el bienestar del alumnado, como unidades de apoyo, podrían redefinir y/o focalizar recursos de apoyo para estudiantes nuevos, ojalá antes de que muestren los primeros signos de complicaciones. Cabe aclarar que la decisión de sobre a qué estudiantes asignar dichos recursos se saldría del *scope* del proyecto.

Las variables predictoras, o características, con las que se entrenaría el modelo simularían la información que una IES tendría en el momento de la matrícula. Por nombrar algunas, corresponderían a variables demográficas (edad al matricularse, género, estado civil, nacionalidad), socioeconómicas (nivel educacional de los padres, tipo de financiamiento de los estudios superiores), antecedentes académicos previos y carrera (qué tipo de carrera, jornada diurna o vespertina, etc.).

Ahora bien, el foco del problema se definió en Chile, no obstante como en general acceder a microdatos sensibles y específicos de estudiantes en Chile es complejo, para efectos metodológicos se validó el uso de un *dataset* "proxy". Este fue construido con datos institucionales de estudiantes matriculados en cursos de pregrado en el Instituto Politécnico de Portalegre, Portugal. En concreto, los datos se refieren a registros de estudiantes matriculados entre los años académicos 2008/09 y 2018/2019 y de 17 diferentes carreras, tales como agronomía, diseño, educación, enfermería, periodismo, administración, servicios sociales y tecnologías [8]. Se precisan datos demográficos y macroeconómicos, datos académicos al momento de la matrícula del estudiante y también a finales del primer y segundo semestre de carrera [9].

Es así como el *dataset* recibe el nombre de "Predict students' dropout and academic success - Investigating the Impact of Social and Economic Factors" y puede ser encontrado en Kaggle [9]. Este se estructura como un archivo en formato CSV y consiste de 4424 (ejemplos/instancias) $\times$ 35 (34 características + 1 columna `Target`; `Graduate`, `Dropout`, `Enrolled`). Ya se encuentra limpio en cuanto a formato (ejemplo, característica) e impurezas como valores "N.A.". El *dataset* comparte las variables estructurales clave buscadas y por tanto permite entrenar el modelo deseado. En cuanto a privacidad, está completamente anonimizado.

Ahora bien, es importante notar que el *dataset* contiene variables que no aplican para el desarrollo del proyecto. Dado el enfoque preventivo y el interés de evaluar el riesgo de deserción desde el primer año de carrera, se considerarán las variables que están disponibles en el momento de la matrícula para predecir el resultado final, con tal de no llevar a *data leakage*. Por tanto, variables como `Debtor` y `Tuition fees up to date`, fueron descartadas al ser información que no se dispone en el primer día de clases. Siguiendo la misma lógica de usar solo lo que se dispone al inicio, las variables que hacen referencia al segundo semestre no pueden ser empleadas, como tampoco los resultados del primer semestre, es decir, `Curricular units 1st sem (evaluations/approved/grade/without evaluations)`. Asimismo, la variable categórica `Target`, que clasifica a estudiantes según su situación al final de la duración formal de su carrera, debe ser modificada para ser binaria. Como esta indica el "estado del estudiante al final de la duración normal de su carrera", con la posibilidad de pertenecer a tres clases; `Graduate`, `Dropout`, `Enrolled`, para el proyecto se consideró que:

$$y = \begin{cases} 1 & \text{si } \texttt{Target} = \texttt{Dropout} \\ 0 & \text{si } \texttt{Target} \in \{\texttt{Enrolled}, \texttt{Graduate}\} \end{cases} \tag{1}$$

De modo tal que 1 representa el fracaso del proceso educativo, mientras que 0 corresponde a estudiantes que persisten o completan exitosamente dicho programa. Entonces, se seleccionaron 22 de 34 variables del *dataset* original para el entrenamiento del modelo. Estas son:

Clase de la variable	Variable	Tipo
Datos demográficos	Marital status	Categórica
	Nacionality	Categórica
	Displaced	Categórica
	Gender	Categórica
	Age at enrollment	Numérica
	International	Categórica
Datos socioeconómicos	Mother's qualification	Categórica
	Father's qualification	Categórica
	Mother's occupation	Categórica
	Father's occupation	Categórica
	Educational special needs	Categórica
	Scholarship holder	Categórica
Datos macroeconómicos	Unemployment rate	Numérica
	Inflation rate	Numérica
	GDP	Numérica
Datos académicos al momento de matricularse	Application mode	Categórica
	Application order	Numérica
	Course (carrera escogida por el estudiante)	Categórica
	Daytime/evening attendance	Categórica
	Previous qualification (Nivel o tipo de estudios previos)	Categórica
Datos académicos para el 1° semestre	Curricular units 1st sem (credited) [cursos convalidados desde otra IES para el primer año]	Numérica
	Curricular units 1st sem (enrolled) [cursos inscritos para el primer año]	Numérica
Objetivo	Target	Categórica

Tabla 1: Descripción de las variables a utilizar del *dataset*.

Entonces, con esta reducción del *dataset* original, para encontrar el modelo final antes mencionado se entrenaron tres modelos: 1. Regresión Logística (RL), 2. Gradient Boosting (GB) y 3. Perceptrón Multicapa (MLP). Para evaluar los modelos en cuanto a mejor desempeño en la clasificación se usaron métricas como:

- Accuracy: Proporción de ejemplos correctamente clasificados.
- Recall: Para una clase positiva dada, proporción de casos reales positivos que fueron correctamente detectados.
- Precision: Para una clase positiva dada, proporción de casos predichos como positivos que realmente pertenecen a la clase.
- F1-score Macro: Media armónica entre Precision y Recall, **útil en clases desbalanceadas**. Se emplea Macro entendido como que primero se calcula el F1-score independientemente para cada clase, luego, se calcula el promedio no ponderado de estas dos puntuaciones, de modo que ambas clases se tratan con igual importancia, independientemente de su frecuencia en el *dataset*.
- Curva ROC y AUC: Evalúa la capacidad de distinguir entre clases en diferentes escenarios de *trade-off* entre verdaderos positivos (cuántos positivos reales fueron correctamente identificados) con la True Positive Rate (TPR) y falsos positivos (cuántos negativos reales fueron incorrectamente identificados como positivos) con la False Positive Rate (FPR).

A través de estas métricas de desempeño finalmente se escogerá el modelo definitivo para el SAT.

3. Experimentos computacionales y análisis

3.1. Preprocesamiento de datos y modelos

Se filtraron las 22 características relevantes del *dataset* original y se transformó el Target a uno binario. Tal y como se vio antes, se definió la clase positiva ($y = 1$) para la etiqueta `Dropout` y la clase negativa ($y = 0$) para la agrupación $\{\text{Enrolled}, \text{Graduate}\}$.

Previo al entrenamiento, se visualizó el desbalance en los datos, donde la clase minoritaria (Deserción, $y = 1$) representa aproximadamente el 32 % de la muestra total; 1421 ejemplares, frente a un 68 % de la clase mayoritaria ($y = 0$); 3003 ejemplares. Este desbalance validó la necesidad de reportar métricas sensibles a la clase positiva (como F1-score y Recall) en el entrenamiento e inferencia. El desbalance mencionado se puede ver a continuación:

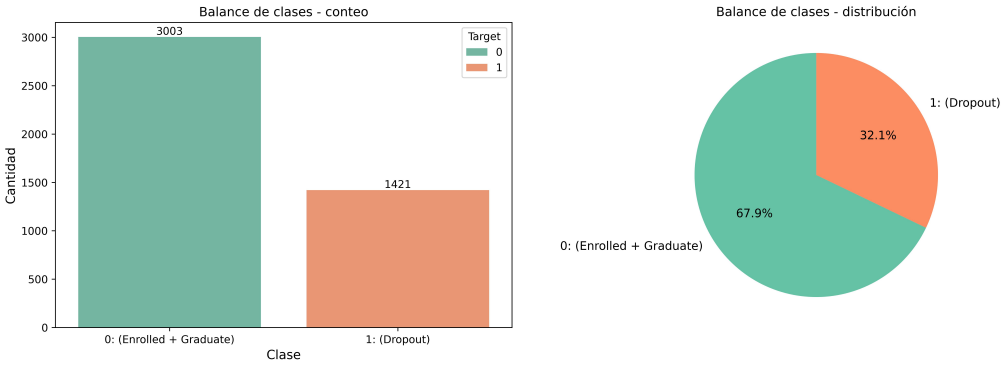

Figura 1: Desbalance de clases y su distribución en los datos considerados.

Como se empleó *cross-validation* para el entrenamiento y validación se dividió el *dataset* original en un conjunto de entrenamiento y validación correspondiente al 80 % de las muestras y en un conjunto de prueba con el 20 % restante [10]. La separación anterior se realizó con `train_test_split` de modo de implementar un muestreo estratificado tal que la proporción de clases se mantuviese constante tanto en ambos conjuntos. Para ello se fijó una semilla aleatoria para asegurar la reproducibilidad de la implementación.

Algunas de las características categóricas contempladas presentaron múltiples categorías, por tanto para que sean entendidas por todos los modelos a entrenar estas se codificaron mediante `OneHotEncoder`. Sin embargo esto induce un desafío crítico en cuanto a la alta dimensionalidad del vector de entrada a los modelos, dado que por ejemplo variables como `Mother's qualification` o `Father's qualification` tienen más de 20 categorías, por tanto, el vector *one-hot encoded* sería de al menos esa dimensión. Lo que es más, algunas de las categorías dentro de estas variables presentaron escasa representatividad muestral. Esto se identificó como una potencial causa de *overfitting* dado que el modelo intentaría aprender patrones a partir de un número muy acotado de observaciones. De hecho, realizando el *one-hot encoding* el número de características ascendió a 221, lo cual es gigantesco en comparación con las características iniciales.

Entonces, para mitigar lo anterior se implementó un transformador "RareLabelEncoder", que inspecciona, dentro del conjunto de entrenamiento, las variables con frecuencia marginal en sus categorías y aplica una regla de decisión basada en un umbral de tolerancia del 2 %. La elección de este umbral fue basado en las frecuencias de algunas categorías de variables vistas en el *dataset*. Las categorías minoritarias se agruparon bajo una categoría genérica `Other`, y con ello se logró compactar el espacio de características, concretamente, en 83 características. Posteriormente se aplicó la codificación *one-hot*, mientras que para las variables numéricas, se procedió con una estandarización mediante *Z-Score* con tal de obtener características numéricas comparables en magnitud. Las dos operaciones anteriores se encapsularon en el llamado "preprocesamiento" dentro de la clase "ColumnTransformer". Ello garantizó que todos los parámetros de transformación se aprendieran solo con el conjunto de entrenamiento, con tal de evitar cualquier forma de *data-leakage*.

Posteriormente, se definieron los tres modelos mencionados en 2, cada uno, encapsulado en *pipelines*. Primero, para el modelo de Regresión Logística se usó el solver `lbfgs` y regularización L_2 con un coeficiente $\lambda = 2$. Dado el desbalance de clases, se definió `class_weight = balanced` con tal de ajustar la función de pérdida al ponderar inversamente cada clase según su frecuencia muestral. Esto penaliza más los errores de clasificación en la clase minoritaria (Deserción) sin alterar cómo distribuyen los datos de entrada.

Segundo, se empleó Gradient Boosting mediante XGBoost. Se usó una función objetivo de pérdida `binary:logistic`. Nuevamente, dado el desbalance de clases, se calculó explícitamente el hiperparámetro `scale_pos_weight` como la razón entre instancias negativas y positivas (≈ 2.11). La profundidad máxima de los árboles se definió en 8 y se aplicó una regularización L_2 con `reg_lambda = 2.5` para prevenir sobreajuste.

Finalmente, se implementó un MLP con una capa oculta de 64 neuronas y función de activación ReLU. Para afrontar el problema del desbalance de clases y no sesgar a la red neuronal con la clase mayoritaria, se optó por implementar una estrategia de *oversampling* mediante Synthetic Minority Oversampling Technique (SMOTE) [11, 12]. Con ello se generaron instancias sintéticas de la clase minoritaria al tomar datos de esta, buscar sus vecinos más cercanos (*k-nearest neighbors*) en el espacio de características preprocesado (*one-hot encoded* y estandarizado, según corresponda) y crear un punto sintético en la recta que une a esos vecinos (en esencia, "interpolación lineal"). SMOTE se integró dentro de un `ImbPipeline` posterior al preprocesamiento (toda variable ahora es numérica) para asegurar que la generación de datos sintéticos ocurriese solo durante el entrenamiento de cada *fold*, evitando una vez más *data-leakage*.

3.2. Entrenamiento de modelos

Para el entrenamiento y validación se empleó *cross-validation* con $K = 5$ *folds* a través de `StratifiedKFold`. Con ello se particionó el conjunto de entrenamiento en K subconjuntos disjuntos preservando la proporción original de clases en cada uno. De este modo, en cada iteración el modelo se entrenó en $K - 1$ *folds* y se evaluó en el restante. Con ello, se obtuvieron los siguientes resultados:

Métrica	Regresión Logística	XGBoost	MLP
Accuracy	0.6937 (± 0.0339)	**0.7090 (± 0.0281)**	0.6982 (± 0.0251)
ROC-AUC	**0.7617 (± 0.0294)**	0.7481 (± 0.0277)	0.7503 (± 0.0265)
F1-score Macro	**0.6734 (± 0.0339)**	0.6662 (± 0.0287)	0.6700 (± 0.0249)
Recall (Clase 1: `Dropout`)	**0.6905 (± 0.0382)**	0.5462 (± 0.0348)	0.6314 (± 0.0343)
Precision (Clase 1: `Dropout`)	0.5189 (± 0.0409)	**0.5492 (± 0.0444)**	0.5266 (± 0.0359)
Neg Log-Loss	**-0.5798 (± 0.0262)**	-0.6279 (± 0.0489)	-0.5842 (± 0.0246)

Tabla 2: Resultados promedios de la validación cruzada ($K = 5$). Se destaca en negrita el mejor desempeño por métrica.

La Tabla 2 muestra que, si bien XGBoost obtuvo la mayor Accuracy, este resultado puede ser engañoso porque, dado el desbalance de clases, la clase mayoritaria tendrá más frecuencia y por tanto el modelo predecirá esta clase la mayor parte del tiempo. Esto se evidencia en el Recall de la clase minoritaria, que es el más bajo entre los tres modelos (de todos los desertores reales, detectó solo al $\approx 54.62\,\%$, predicción que casi puede ser comparada con lanzar una moneda). En el contexto del proyecto, un SAT debe poder detectar a tiempo un estudiante que va a desertar, por tanto, fallar en esto (levantar un falso negativo) y no intervenir significa un alto costo por lo visto en 1, mientras que creer que deserterá pero al final no lo hace (levantar un falso positivo) involucra un costo significativamente menor. En el fondo, como es preferible intervenir a un estudiante que finalmente no deserta, a ignorar a uno que sí lo hará, se busca priorizar la minimización de falsos negativos por sobre los falsos positivos.

Por el contrario, la Regresión Logística demostró el desempeño más robusto para los objetivos deseados. Superó a los otros modelos en todas las métricas excepto en Accuracy (aunque por lo visto antes se permite sacrificar esta métrica con tal de poder obtener un mayor Recall) y Precision. Sin embargo, frente a una baja Precision de $\approx 52\,\%$ (solo este porcentaje de los que predije que desertarían realmente lo harán) y un Recall de $\approx 69\,\%$ (detecté al $69\,\%$ de los verdaderos desertores) es preferible priorizar esta última métrica por el mismo argumento del párrafo anterior. En cuanto a la ROC-AUC, el valor reportado para este modelo indica que la Regresión Logística discrimina mejor entre estudiantes que desertarán de los que no lo harán. Respecto de F1-score Macro, el valor obtenido sugiere el mejor equilibrio entre Precision y Recall para ambas clases, en general. Además, si bien no es una métrica fundamental, el valor de Neg Log-Loss indica que las probabilidades y predicción entregadas por el modelo son más precisas que las de los otros (al preferirse un valor más cercano a 0 al minimizar la pérdida empleada).

Por último, el desempeño del MLP, aunque competitivo con respecto a la Regresión Logística, se situó en un punto intermedio sin justificar su mayor costo computacional. En conclusión, se seleccionó a la Regresión Logística como el modelo definitivo para la inferencia.

3.3. Inferencia

Se procedió a reentrenar el modelo utilizando la totalidad del conjunto de entrenamiento y validación, para luego evaluarlo sobre el conjunto de prueba reservado anteriormente (885 muestras). Los resultados obtenidos confirman la robustez del modelo:

Métrica	Valor
Accuracy	0.68
ROC-AUC	0.7754
F1-score Macro	0.67
Recall (Clase 1: `Dropout`)	**0.71**
Recall (Clase 0: `Enrolled/Graduate`)	0.67
Precision (Clase 1: `Dropout`)	0.50
Precision (Clase 0: `Enrolled/Graduate`)	0.83

Tabla 3: Métricas para inferencia realizada con modelo de Regresión Logística sobre el set de prueba.

De la Tabla 3 puede verse que las métricas, en general, son coherentes con las vistas en 2 para Regresión Logística. Esta consistencia entre las métricas de entrenamiento-validación con las de prueba es evidencia de que el desempeño del modelo en el conjunto de prueba es bueno al no identificarse *overfitting* y por tanto reportarse una buena capacidad de generalización. En definitiva, se destaca que el modelo, con un 71 % de Recall para la clase minoritaria, es capaz de detectar correctamente en el momento de la matrícula a aproximadamente 7 de cada 10 estudiantes que efectivamente desertarán en algún momento de su carrera.

Adicionalmente puede verse la curva ROC para el modelo, donde se ve su alta capacidad de discriminación al estar por encima de la diagonal (clasificador *random*):

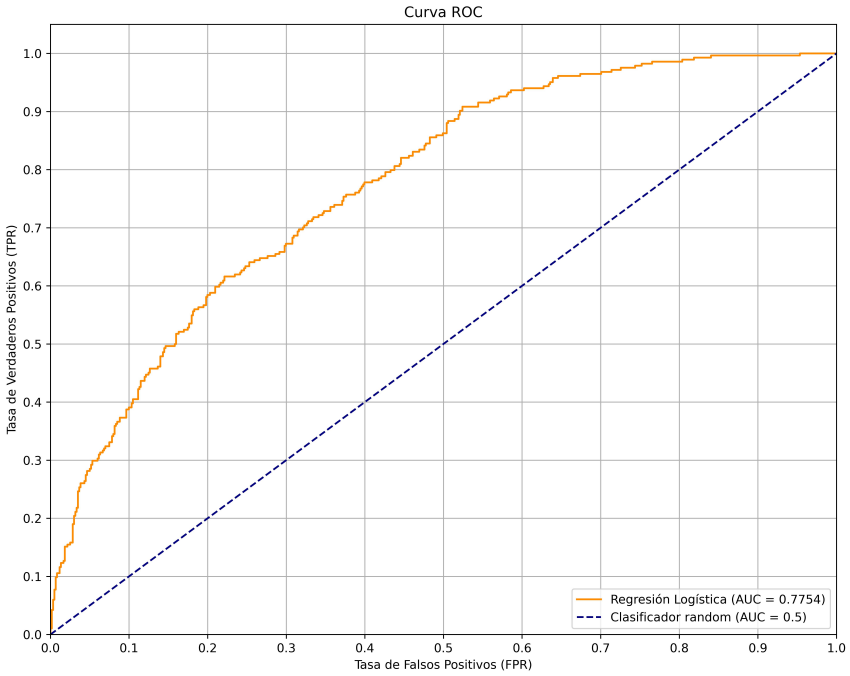

Figura 2: Curva ROC para el modelo de Regresión Logística en inferencia.

Para complementar las métricas anteriores se construyó la matriz de confusión, donde esta es estructurada como:

$$\begin{pmatrix} TN & FP \\ FN & TP \end{pmatrix} \tag{2}$$

Donde TN, FP, FN y TP corresponden a verdaderos negativos, falsos positivos, falsos negativos y verdaderos positivos. Se tiene entonces:

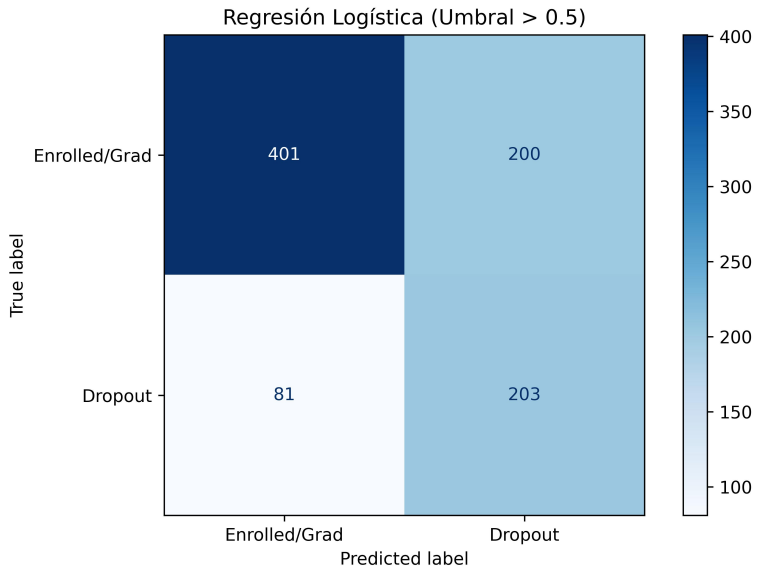

Figura 3: Matriz de confusión para inferencia en conjunto de prueba con modelo de Regresión Logística. Se clarifica que el umbral de decisión fue de una probabilidad del 50 %.

A partir de la Figura 2 puede verse que el modelo clasificó correctamente a $TN = 401$ estudiantes como "Enrolled/Grad" (no-desertores), clasificó incorrectamente a $FP = 200$ estudiantes como `Dropout` (desertores) cuando, en realidad, no desertaron, clasificó incorrectamente a $FN = 81$ estudiantes como `Enrolled/Grad` cuando en realidad sí desertaron, y por último clasificó correctamente a $TP = 203$ estudiantes como desertores. Como se mencionó antes, lo fundamental del SAT es minimizar los casos de FN, por tanto que el modelo se haya equivocado en predecir la deserción cuando el estudiante finalmente no lo hará, es decir, un FP, no es inaceptable, al ser relativamente efectivo para no ignorar desertores reales. Este bajo valor para FN es esperable dada la limitación de usar solo variables al momento de matrícula, sin considerar el rendimiento académico durante los semestres venideros, los cuales pueden ser factores críticos en si continuar o no una carrera.

En definitiva, para una IES el alto número de FP implica un gasto innecesario de recursos de apoyo al intervenir con estudiantes que habrían continuado sus estudios sin ayuda, sin embargo, el bajo número de FN (81) significa que se está fallando en intervenir en pocos casos de riesgo de deserción, los cuales son críticos pues implica perder a un alumno.

Por último, se extrajeron los coeficientes del clasificador lineal de modo tal de poder entender qué variables son las que más afectan (y de qué forma lo hacen) al objetivo de predicción. Para ello se obtuvo el *logit*:

$$\ln(\text{Odds}) = \ln\left(\frac{P(\text{Dropout})}{1 - P(\text{Dropout})}\right) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n \tag{3}$$

Donde Odds es la razón de probabilidades de ambas clases, θ corresponden a los coeficientes o parámetros del modelo entrenado y x a las características o variables dentro del vector de entrada. En particular, el signo del parámetro es útil para determinar la dirección del impacto, es decir, valores positivos indican un aumento en la probabilidad de la clase positiva (`Dropout`), mientras que los negativos actúan como factores protectores ante el riesgo de pertenecer a esta clase.

Entonces, se presenta a continuación los 12 factores/parámetros/coeficientes de entrada más determinantes del modelo. En rojo se visualizan los coeficientes positivos (aumentan el riesgo de deserción) y en verde los negativos (aminoran dicho riesgo):

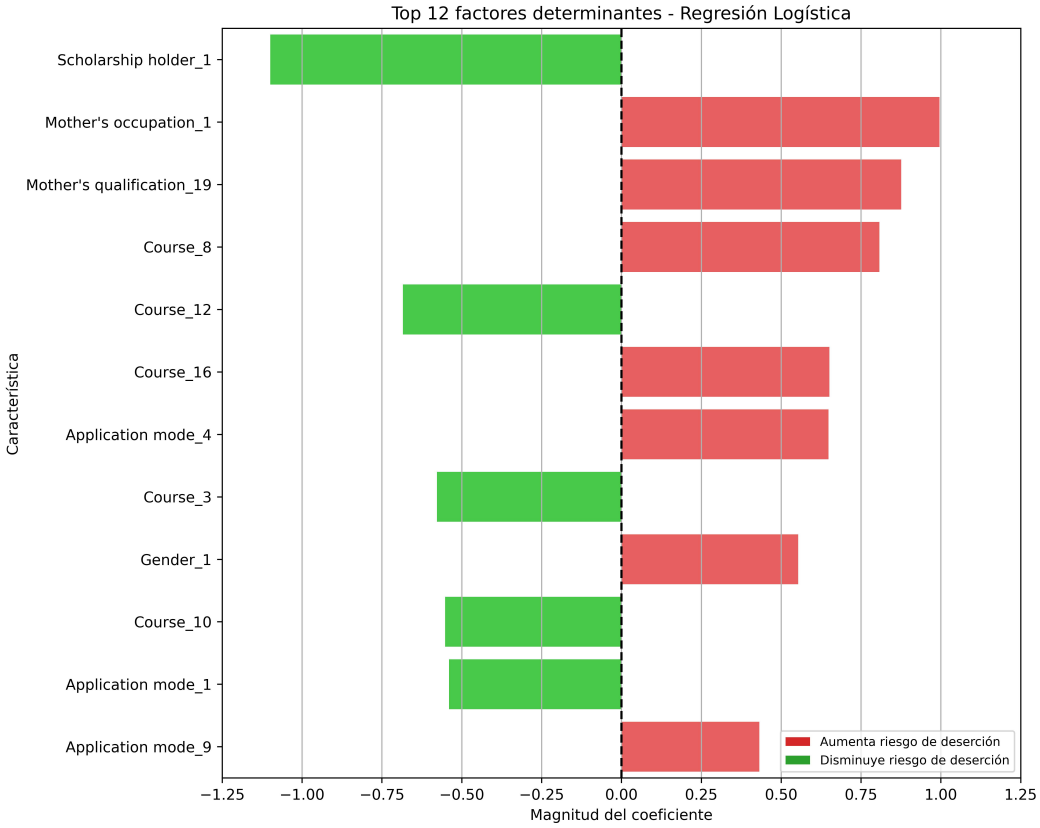

Figura 4: Los 12 coeficientes (y variables) más influyentes del modelo en la predicción realizada.

De la Figura 4 puede verse que, de los coeficientes positivos que aumentan el riesgo de deserción, los antecedentes socioeconómicos (ocupación y calificación de la madre) y la carrera elegida son los determinantes de riesgo más preponderantes en el modelo. Por su parte, de los coeficientes negativos que reducen el riesgo de deserción, el factor más "protector" es ser becario, aunque otros factores como la carrera escogida también impactan. Sin embargo, en términos cuantitativos no es mucha la información extraíble.

Para ver el impacto de la magnitud del parámetro, de acuerdo con [13, 14], se debe obtener el Odds Ratio para una determinada variable x_i asociada a un parámetro θ_i y por tanto para un determinado parámetro:

$$(\text{Odds Ratio})_i = \frac{\text{Odds}(x_i + 1)}{\text{Odds}(x_i)} = e^{\theta_i} = e^{\text{coeficiente}} \tag{4}$$

Que corresponde a un multiplicador que muestra cómo cambia la razón de probabilidades (Odds) por cada aumento de una unidad en el valor de la variable x_i. Con ello, es posible ver los Odds Ratios de cada coeficiente/parámetro visto antes. Se tiene entonces:

Tipo de Factor	Feature	Coeficiente/parámetro	Odds Ratio
Factores que aumentan riesgo de deserción	`Mother's occupation_1`	0.9960	2.7073
	`Mother's qualification_19`	0.8760	2.4014
	`Course_8`	0.8078	2.2430
	`Course_16`	0.6511	1.9177
	`Application mode_4`	0.6484	1.9124
Factores que reducen riesgo de deserción	`Scholarship holder_1`	-1.0997	0.3330
	`Course_12`	-0.6839	0.5046
	`Course_3`	-0.5773	0.5614
	`Course_10`	-0.5518	0.5759
	`Application mode_1`	-0.5400	0.5827

Tabla 4: Los 5 factores más importantes asociados a la deserción y a la permanencia. El signo del coeficiente y el Odds Ratio indican la dirección y magnitud del impacto de la variable asociada en la probabilidad de deserción (Clase 1), respectivamente.

Según [9] y la Tabla 4, de los factores que aumentan el riesgo de deserción, destaca `Mother's occupation_1 = Student` como la variable más crítica en cuanto al riesgo de deserción. Manteniendo las otras variables constantes, esta indica que estudiantes con una madre estudiante tienen ≈ 2.71 veces más riesgo de desertar. No se le queda atrás `Mother's qualification_19 = 2nd cycle of the general high school course`, dado que estudiantes en esta situación tienen 2.4 veces más riesgo de deserción que alguien en otra situación. También, estar matriculado en `Course_8 = Equiniculture` aumenta el riesgo de deserción en 2.24 veces.

Por su parte, de los factores que reducen el riesgo de deserción, destaca `Scholarship holder_1 = yes` como la variable más importante en la aminoración del riesgo de deserción. Manteniendo las otras variables constantes, ser becario reduce el riesgo de deserción en aproximadamente 66.7 % ($1 - \text{Odds Ratio} = 1 - 0.333$). También, estar matriculado en `Course_12 = Nursing`, `Course_3 = Social Service (evening attendance)`, `Course_10 = Social Service` da cuenta de una reducción del riesgo de deserción en el rango de $\approx [42\,\%, 50\,\%]$.

El análisis anterior es revelador. Es claro que el contexto económico es un predictor clave, dado que la ocupación y calificación de la madre son los mayores indicadores de riesgo de deserción, como también que la tenencia de una beca es el factor más determinante en reducir drásticamente el riesgo de abandono (reducción de más del 60 %). Se podría decir que la ayuda económica al estudiante es crucial y subraya que la institución debe prestar especial atención a los estudiantes con estos perfiles socioeconómicos más críticos para ofrecer apoyo.

4. Conclusiones

El presente proyecto presentó la aplicación de un modelo de ML para un SAT a implementar en IES dentro del contexto de la detección de deserción estudiantil, esto en cualquier momento de la duración formal de su carrera, y basado exclusivamente en información disponible al momento de la matrícula. A través de la experimentación con distintos algoritmos de aprendizaje supervisado, se demostró que es posible perfilar el riesgo de deserción sin necesidad de esperar a los primeros resultados académicos, lo cual permite una transición desde un enfoque institucional reactivo hacia uno preventivo. Si bien el uso de un *dataset* proxy proveniente de Portugal es una limitación de este proyecto, la metodología de preprocesamiento y selección de modelos es transferible al contexto chileno, siempre que se disponga de datos históricos locales equivalentes.

En cuanto a la selección del modelo, la Regresión Logística demostró ser la arquitectura más adecuada para los objetivos del SAT. Esta superó a modelos más complejos como XGBoost y MLP en Recall, la métrica crítica en el contexto educacional para detectar falsos negativos (no detectar a un estudiante que realmente abandonará sus estudios). En ese sentido, se destaca la capacidad del modelo para identificar correctamente al 71 % de los estudiantes que efectivamente desertarán y la decisión de sacrificar Accuracy en el modelo en favor de una mayor cobertura de detección de riesgo y así asegurar la retención del estudiante.

El análisis de los resultados del modelo reveló que los factores socioeconómicos son predominantes en la etapa de ingreso. Específicamente, la condición de becario fue el factor de reducción de riesgo de deserción más potente, al reducirlo en aproximadamente un 67 %. Mientras que antecedentes familiares, como la ocupación y calificación de la madre, actuaron como fuertes indicadores de riesgo. Ello sugiere que las políticas de retención por parte de las IES no deben limitarse al apoyo académico, sino que deben integrar robustos mecanismos de apoyo financiero y bienestar estudiantil desde el día uno, enfocándose en los perfiles demográficos identificados por el modelo.

Finalmente, es importante notar que el modelo presenta un número no menor de falsos positivos, lo que implica una asignación de recursos a estudiantes que no desertarían. Como trabajo futuro, se podría utilizar este modelo de Regresión Logística para una segmentación inicial al momento de la matrícula y, posteriormente, actualizar las probabilidades de riesgo tras las primeras evaluaciones. Ello permitiría refinar la precisión del SAT al reducir, eventualmente, el número de falsos positivos, para así optimizar la intervención de unidades de apoyo estudiantil a medida que avanza el semestre académico. Adicionalmente, podría variarse el umbral de decisión en validación y prueba a un valor menor a 50 % con tal de obtener un SAT más sensible en cuanto a la identificación de desertores.

Referencias

[1] D. Opazo, S. Moreno, E. Álvarez Miranda, and J. Pereira, "Analysis of first-year university student dropout through machine learning models: A comparison between universities," *Mathematics*, vol. 9, no. 20, 2021. [Online]. Available: https://www.mdpi.com/2227-7390/9/20/2599

[2] A. J. Fernández-García, J. C. Preciado, F. Melchor, R. Rodriguez-Echeverria, J. M. Conejero, and F. Sánchez-Figueroa, "A real-life machine learning experience for predicting university dropout at different stages using academic data," *IEEE Access*, vol. 9, pp. 133 076–133 090, 2021. [Online]. Available: 10.1109/ACCESS.2021.3115851

[3] J. Berens, K. Schneider, S. Gortz, S. Oster, and B. Julian, "Early Detection of Students at Risk - Predicting Student Dropouts Using Administrative Student Data from German Universities and Machine Learning Methods," *Journal of Educational Data Mining*, vol. 11, no. 3, pp. 1–41, 2019. [Online]. Available: https://jedm.educationaldatamining.org/index.php/JEDM/article/view/389

[4] Servicio de Información en Educación Superior [SIES], "Informe 2025, Retención de 1er año de pregrado, Cohortes 2020-2024," Subsecretaría de Educación Superior - Ministerio de Educación, Tech. Rep., 2025.

[5] F. A. Bello, J. Kóhler, K. Hinrechsen, V. Araya, L. Hidalgo, and J. L. Jara, "Using machine learning methods to identify significant variables for the prediction of first-year informatics engineering students dropout," in *2020 39th International Conference of the Chilean Computer Science Society (SCCC)*, 2020, pp. 1–5. [Online]. Available: 10.1109/SCCC51225.2020.9281280

[6] L. J. Rodríguez-Muñiz, A. B. Bernardo, M. Esteban, and I. Díaz, "Dropout and transfer paths: What are the risky profiles when analyzing university persistence with machine learning techniques?" *PLOS ONE*, vol. 14, no. 6, pp. 1–20, 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0218796

[7] O. Goren, L. Cohen, and A. Rubinstein, "Early prediction of student dropout in higher education using machine learning models," in *Proceedings of the 17th International Conference on Educational Data Mining*, B. PaaÃŸen and C. D. Epp, Eds. Atlanta, Georgia, USA: International Educational Data Mining Society, 2024, pp. 349–359. [Online]. Available: https://educationaldatamining.org/edm2024/proceedings/2024.EDM-short-papers.32/index.html

[8] M. V. Martins, D. Tolledo, J. Machado, L. M. T. Baptista, and V. Realinho, "Early prediction of student's performance in higher education: A case study," in *Trends and Applications in Information Systems and Technologies*, Á. Rocha, H. Adeli, G. Dzemyda, F. Moreira, and A. M. Ramalho Correia, Eds. Cham: Springer International Publishing, 2021, pp. 166–175.

[9] V. Realinho, J. Machado, L. Baptista, and M. V. Martins, "Predicting student dropout and academic success," *Data*, vol. 7, no. 11, 2022. [Online]. Available: https://www.mdpi.com/2306-5729/7/11/146

[10] Rathi, P., "What is the difference between cross validation and test set in validation of machine learning algorithm?" [Online]. Available: https://www.quora.com/What-is-the-difference-between-cross-validation-and-test-set-in-validation-of-machine-learning-algorithm

[11] Imbalanced Learn, "SMOTE." [Online]. Available: https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, p. 321–357, 2002. [Online]. Available: http://dx.doi.org/10.1613/jair.953

[13] U. (https://stats.stackexchange.com/users/182590/user1607), "logit - interpreting coefficients as probabilities," Cross Validated. [Online]. Available: https://stats.stackexchange.com/q/363806

[14] Wikipedia, "Logistic regression." [Online]. Available: https://en.wikipedia.org/wiki/Logistic_regression