**Minería de datos aplicada a la predicción de deserción académica en educación superior**

**Laura Sofia Jiménez Moreno**

**Sofia Mejia Rivas**

**Universidad Pontificia Bolivariana**

**Facultad de Ingeniería**

**Ingeniería en Ciencia de datos**

**Minería de datos estructurados**

**Medellín**

**2025-1**

1. **ENTENDIMIENTO DEL NEGOCIO:** 
   1. **DESCRIPCIÓN DEL NEGOCIO**:

La deserción estudiantil representa una **pérdida anual de hasta $16 mil millones de dólares** para las instituciones de educación superior en América Latina, según un informe del (Grupo Banco Mundial, 2022). Para mitigar este problema, universidades como la **Universidad Nacional Autónoma de México (UNAM)** han implementado sistemas de alerta temprana basados en aprendizaje automático, logrando **reducir la deserción en un 15%** en programas de ingeniería (UNAM, 2021).Este proyecto académico replica metodologías validadas (como CRISP-DM) utilizando el dataset de la **Universidad de Portugal (UCI Machine Learning Repository)**, que incluye variables clave avaladas por la **UNESCO** para predecir deserción: rendimiento académico, estatus socioeconómico y engagement estudiantil (UNESCO, 2022). Su viabilidad técnica se basa en herramientas de código abierto (Python, scikit-learn), alineadas con las recomendaciones de la **OECD** para instituciones con recursos limitados (OECD, 2023).

* 1. **DESCRIPCIÓN DEL PROBLEMA**:

La deserción estudiantil en la educación superior representa un problema crítico para las instituciones educativas, ya que impacta negativamente en la calidad del sistema educativo, la inversión pública y privada, y el desarrollo social del país. De acuerdo con el Ministerio de Educación Nacional de Colombia (Ministerio de Educación Nacional de Colombia, 2022) cerca del 45 % de los estudiantes abandonan sus estudios antes de finalizar su carrera, especialmente en los primeros semestres. Este fenómeno está asociado a múltiples causas como dificultades económicas, bajo rendimiento académico, falta de orientación vocacional y problemas personales o familiares. Comprender estos factores es esencial para enfrentar el problema de fondo y plantear soluciones efectivas desde la gestión institucional y las políticas públicas.

* 1. **OBJETIVOS DE LA MINERÍA**:
* Desarrollar modelos predictivos de deserción estudiantil mediante el entrenamiento y evaluación de siete algoritmos de machine learning: cuatro modelos supervisados (árbol de clasificación, red neuronal, SVM y KNN) y tres métodos de ensamble (voting soft, XGBoost y random forest).
* Seleccionar los tres mejores modelos mediante análisis de diferencia estadística (ANOVA y Tukey), seguido de una etapa de hiperparametrización con GridSearch y optimización mediante algoritmos genéticos o bayesiana, con el fin de la obtención de un modelo más preciso para predicción de la deserción estudiantil.
* Implementar el despliegue del modelo final mediante un pipeline que integre las transformaciones de datos y una interfaz gráfica (GUI) que facilite su uso por parte de las instituciones educativas.
  1. **DISEÑO DE SOLUCIÓN**:

Como resultado esperado, se busca que el modelo final alcance un **accuracy** superior al 70%, un **recall** 0.75 que permita detectar correctamente a los estudiantes que desertan, y un **F1-score** también por encima de **0.75**. Además, se espera un buen desempeño en el **AUC-ROC**, idealmente por encima de 0.80, para asegurar una adecuada capacidad de discriminación entre clases.



**1.5 RECURSOS PARA CREACIÓN DEL MODELO Y PARA DESPLIEGUE**:

El modelo fue desarrollado en **Jupyter** **notebooks usando Python 3.10 for data science**, utilizando librerías especializadas como pandas, scikit-learn, matplotlib, seaborn, imbalanced-learn, entre otras. El procesamiento y entrenamiento se realizó en máquinas virtuales en Deepnote, cuaderno colaborativo basado en la nube, con 2 vCPU (Basic) de 5GB de RAM.

Para el **despliegue**, se utilizó **Streamlit**, una herramienta que permite construir interfaces web interactivas en Python. La aplicación fue publicada gratuitamente mediante la conexión entre **GitHub y Streamlit**. Para esto, se creó un repositorio en GitHub.

La aplicación queda disponible como una **app web accesible desde cualquier navegador**, sin necesidad de instalaciones locales, ideal para instituciones educativas que utilicen computadores de escritorio o portátiles con conexión a internet.

**2. ENTENDIMIENTO DE LOS DATOS:**

**2.1 CICLO DE LOS DATOS**:

Los datos fueron generados en un entorno académico, específicamente por una institución de educación superior en Portugal, que recopiló información de estudiantes de los programas de licenciatura en Ciencias de la Computación y Sistemas de Información. Estos datos fueron recolectados de los registros institucionales y no provienen de un entorno en la nube, sino de fuentes internas de la universidad.

El almacenamiento original fue realizado de forma local y posteriormente compartido de forma estática a través del repositorio de UCI Machine Learning. En este contexto, los datos no son modificados en tiempo real ni cuentan con un flujo continuo de actualización. No hay un responsable actual que mantenga los datos activos o en constante modificación; se trata de un conjunto histórico.

Dado que los datos no se actualizan de forma periódica (no hay nuevas versiones cada seis meses o cada cierto tiempo), no es posible establecer un cronograma de reentrenamiento automático. Sin embargo, si este modelo se llevara a producción en una institución educativa real, se recomendaría hacer un reentrenamiento **al final de cada semestre académico**, coincidiendo con la disponibilidad de datos nuevos sobre rendimiento y permanencia estudiantil.

**2.2 DICCIONARIO DE DATOS**:

|  |  |  |
| --- | --- | --- |
| **Variable** | **Descripción** | **De qué tipo es (categórica, Numérica, string, float)** |
| **Marital status** | Estado civil del estudiante | Categórica (Codificada en entero) |
| **Application mode** | Modalidad de ingreso a la universidad | Categórica (Codificada en entero) |
| **Application** **order** | Orden de preferencia de la carrera | Numérica (Codificada en entero) |
| **Course** | Código de la carrera matriculada | Categórica (Codificada en entero) |
| **Daytime/evening attendance** | Turno de estudios | Binaria (Codificada en entero) |
| **Previous qualification** | Nivel educativo previo | Categórica (Codificada en entero) |
| **Previous qualification (grade)** | Nota de la titulación previa | Numérica (float) |
| **Nacionality** | Nacionalidad del estudiante | Categórica (Codificada en entero) |
| **Mother's qualification** | Nivel educativo de la madre | Categórica (Codificada en entero) |
| **Father's qualification** | Nivel educativo del padre | Categórica (Codificada en entero) |
| **Mother's occupation** | Ocupación de la madre | Categórica (Codificada en entero) |
| **Father's occupation** | Ocupación del padre | Categórica (Codificada en entero) |
| **Admission grade** | Nota de admisión a la universidad | Numérica (float) |
| **Displaced** | Indica si es estudiante desplazado | Binaria (Codificada en entero) |
| **Educational special needs** | Necesidades educativas especiales | Binaria (Codificada en entero) |
| **Debtor** | Indica si tiene deudas pendientes | Binaria (Codificada en entero) |
| **Tuition fees up to date** | Indica si tiene las tasas al día | Binaria (Codificada en entero) |
| **Gender** | Género del estudiante | Binaria (Codificada en entero) |
| **Scholarship holder** | Indica si es beneficiario de beca | Binaria (Codificada en entero) |
| **Age at enrollment** | Edad al momento de matricularse | Numérica (Entero) |
| **International** | Indica si es estudiante internacional | Binaria (Codificada en entero) |
| **Curricular units 1st sem (credited)** | Asignaturas convalidadas en 1er semestre | Numérica (Entero) |
| **Curricular units 1st sem (enrolled)** | Asignaturas matriculadas en 1er semestre | Numérica (Entero) |
| **Curricular units 1st sem (evaluations)** | Evaluaciones realizadas en 1er semestre | Numérica (Entero) |
| **Curricular units 1st sem (approved)** | Asignaturas aprobadas en 1er semestre | Numérica (Entero) |
| **Curricular units 1st sem (grade)** | Nota media del 1er semestre | Numérica (float) |
| **Curricular units 1st sem (without evaluations)** | Asignaturas no evaluadas en 1er semestre | Numérica (Entero) |
| **Curricular units 2nd sem (credited)** | Asignaturas convalidadas en 2do semestre | Numérica (Entero) |
| **Curricular units 2nd sem (enrolled)** | Asignaturas matriculadas en 2do semestre | Numérica (Entero) |
| **Curricular units 2nd sem (evaluations)** | Evaluaciones realizadas en 2do semestre | Numérica (Entero) |
| **Curricular units 2nd sem (approved)** | Asignaturas aprobadas en 2do semestre | Numérica (Entero) |
| **Curricular units 2nd sem (grade)** | Nota media del 2do semestre | Numérica (float) |
| **Curricular units 2nd sem (without evaluations)** | Asignaturas no evaluadas en 2do semestre | Numérica (Entero) |
| **Unemployment rate** | Tasa de desempleo nacional | Numérica (float) |
| **Inflation rate** | Tasa de inflación nacional | Numérica (float) |
| **GDP** | Producto Interno Bruto del país | Numérica (float) |
| **Target** | Resultado académico del estudiante | Categórica (Object) |

**2.3 REGLAS DE CALIDAD**:

|  |  |
| --- | --- |
| **Variable** | **Regla de calidad (valores válidos)** |
| Marital status | 1 (soltero), 2 (casado), 3 (viudo), 4 (divorciado), 5 (unión de hecho), 6 (separado) |
| Application mode | Enteros 1-57 (códigos de modalidad de ingreso) |
| Application order | Enteros 0-9 (0=primera opción) |
| Course | Códigos específicos (ej. 9119=Ingeniería Informática) |
| Daytime/evening attendance | 0 (vespertino), 1 (diurno) |
| Previous qualification | Enteros 1-44 (niveles educativos) |
| Previous qualification (grade) | 0-200 (nota) |
| Nacionality | Códigos 1-109 (ej. 1=portugués) |
| Mother's qualification | Enteros 1-44 |
| Father's qualification | Enteros 1-44 |
| Mother's occupation | Códigos 0-194 |
| Father's occupation | Códigos 0-195 |
| Admission grade | 0-200 (nota) |
| Displaced | 0 (no), 1 (sí) |
| Educational special needs | 0 (no), 1 (sí) |
| Debtor | 0 (no), 1 (sí) |
| Tuition fees up to date | 0 (no), 1 (sí) |
| Gender | 0 (mujer), 1 (hombre) |
| Scholarship holder | 0 (no), 1 (sí) |
| Age at enrollment | ≥17 años |
| International | 0 (no), 1 (sí) |
| Curricular units 1st sem (credited) | ≥0 |
| Curricular units 1st sem (enrolled) | ≥0 |
| Curricular units 1st sem (evaluations) | ≥0 |
| Curricular units 1st sem (approved) | ≥0 |
| Curricular units 1st sem (grade) | 0-20 (nota) |
| Curricular units 1st sem (without evaluations) | ≥0 |
| Curricular units 2nd sem (credited) | ≥0 |
| Curricular units 2nd sem (enrolled) | ≥0 |
| Curricular units 2nd sem (evaluations) | ≥0 |
| Curricular units 2nd sem (approved) | ≥0 |
| Curricular units 2nd sem (grade) | 0-20 (nota) |
| Curricular units 2nd sem (without evaluations) | ≥0 |
| Unemployment rate | ≥0 (%) |
| Inflation rate | ≥0 (%) |
| GDP | ≥0 |
| Target | 1 (abandono), 2 (matriculado), 3 (graduado) |

**3. PREPARACIÓN DE DATOS:**

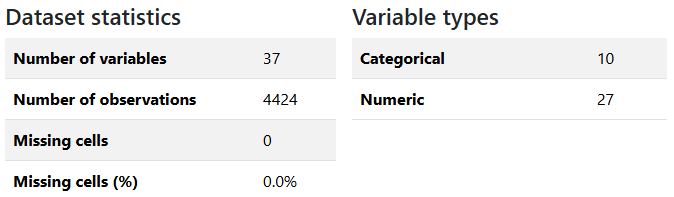
**3.1 INTEGRACIÓN**:

No es necesario realizar una integración de los datos ya que se encuentran todos en un mismo archivo estructurado.

**3.2 SELECCIÓN DE VARIABLES**:

No se realiza borrado de variables no relevantes, mediante una primera evaluación se llega a que todas las variables predictoras iniciales pueden contribuir a la predicción de la variable objetivo

**3.3 DESCRIPCIÓN ESTADÍSTICA**:

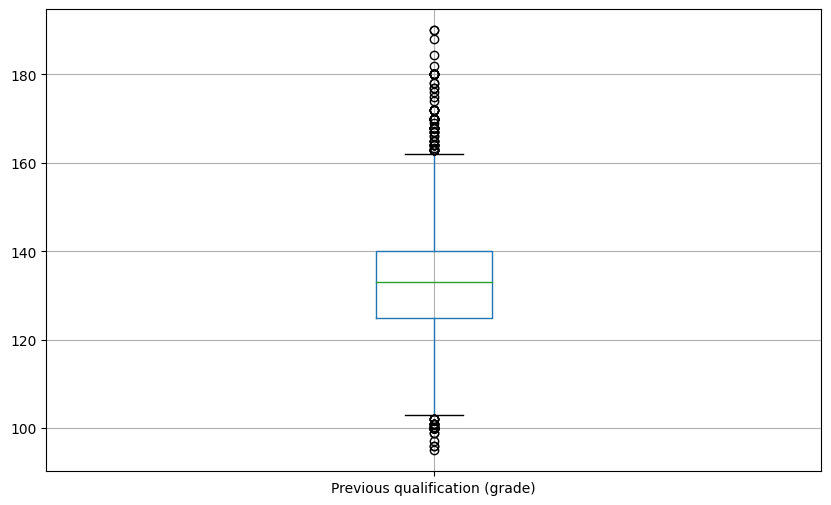


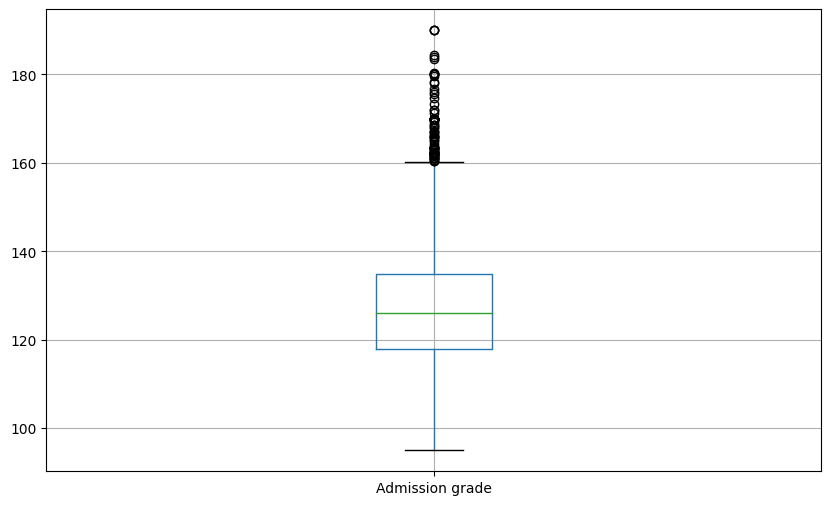
**3.4 LIMPIEZA DE ATÍPICOS**:

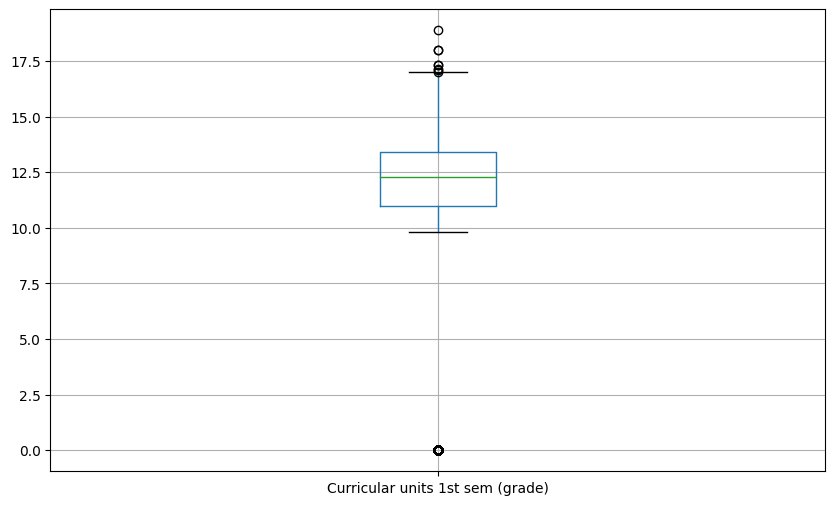
Para realizar la limpieza de datos atípicos se consulta en la página de donde se obtuvo el dataframe las reglas de calidad de cada variable, de las cuales se encuentran solamente para cuatro variables numéricas.

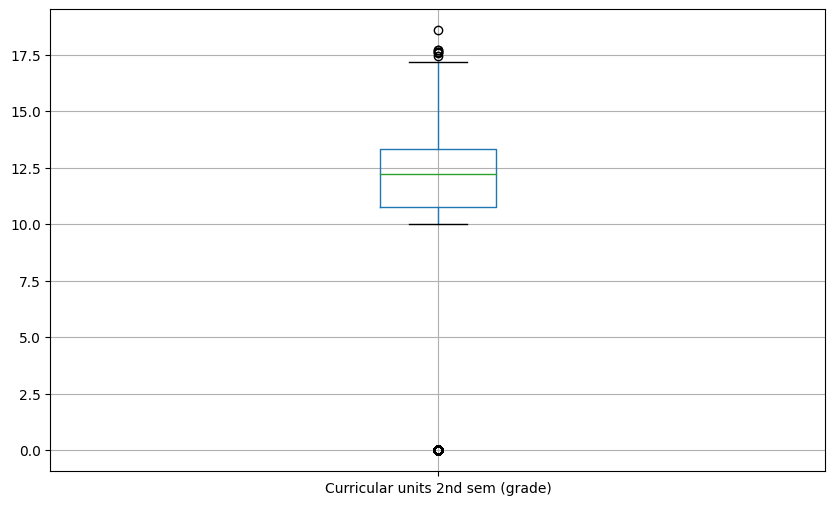
* Previous qualification (grade) rango: [0, 200]
* Admission grade rango: [0, 200]
* Curricular units 1st sem (grade) rango: [0,20]
* Curricular units 2nd sem (grade) rango: [0,20]

Por lo que se procede a revisar si existen valores que incumplen sus respectivas normas, mediante boxplots y consultando si existen valores que no se encuentren en estos rangos.

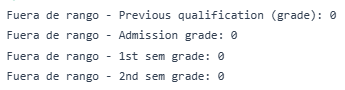








Además, se consulta mediante los rangos establecidos si existen datos que se encuentren por fuera:



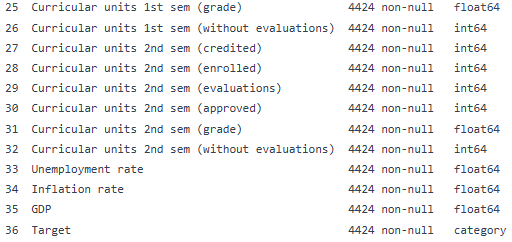
Se puede evidenciar tanto por diagramas de cajas y bigotes, como por valores fuera del rango que los datos considerados outliers cumplen con las reglas de calidad que se conocen, por lo que no se deben eliminar.

**3.5 LIMPIEZA DE NULOS:**

Se toma como estrategia:

* Eliminar registros con más de 30% de nulos.
* Eliminar columnas con más de 15%-20% de nulos.
* Imputar por media, moda, mediana, vecinos cercanos. No se puede imputar más allá del 15%-20% de los datos.
* Para casos especiales se crea modelo predictivo.





No se presentan datos nulos debido a que el dataset ya ha sido sometido a preprocesamiento. Pero si fuese necesario, se imputarían por su promedio, que no altera la media muestral; o por otras imputaciones más detalladas como imputación por valor anterior o siguiente, o por vecinos más cercanos.

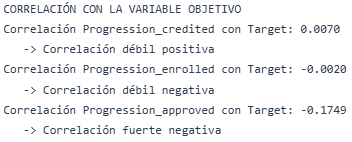
Explicación sobre el pre-procesamiento de datos presente en la página donde se extrajeron los datos:



**3.6 CREACIÓN DE NUEVAS VARIABLES**:

Se analiza la creación de variables que evidencien el cambio respecto a los créditos entre primer y segundo semestre, ya que puede ser útil para capturar patrones de mejora o deterioro académico, de la siguiente manera:

1. **Progresión en créditos convalidados**: Progression\_credited = Curricular units 2nd sem (credited) - Curricular units 1st sem (credited)
2. **Progresión en créditos matriculadas:** Progression\_enrolled = Curricular units 2nd sem (enrolled) - Curricular units 1st sem (enrolled)
3. **Progresión en créditos aprobados;** Progression\_approved = Curricular units 2nd sem (approved) - Curricular units 1st sem (approved)

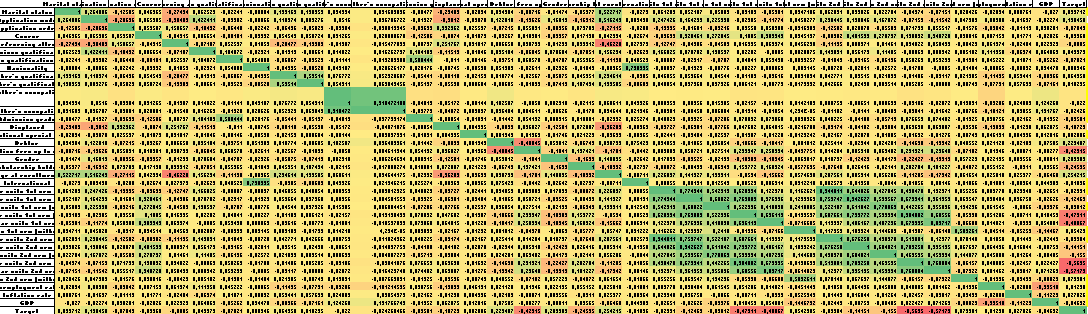


Dado que, mediante el análisis de correlaciones para irrelevancia, Progression\_credited y Progression\_enrolled se eliminarían, ya que están por debajo del umbral de 0.01. Se decide no continuar con estas variables progresivas

**3.7 ANÁLISIS DE CORRELACIONES PARA REDUNDANCIA**:

Para el análisis de correlaciones se realizó una matriz utilizando seaborn, sin embargo, debido a sus grandes dimensiones la visualización se dificultó, por lo que se exporta a Excel y se aplica la herramienta de formato condicional para escalas de color.





Como resultado de verificar alta correlación entre variables predictoras, se obtiene:

Variables con alta correlación entre sí (> 0.8):

- International y Nacionality\_Portuguese tienen correlación -1.00, considerar eliminar una.

- Curricular units 1st sem (credited) y Curricular units 2nd sem (credited) tienen correlación 0.94, considerar eliminar una.

- Curricular units 1st sem (enrolled) y Curricular units 2nd sem (enrolled) tienen correlación 0.94, considerar eliminar una.

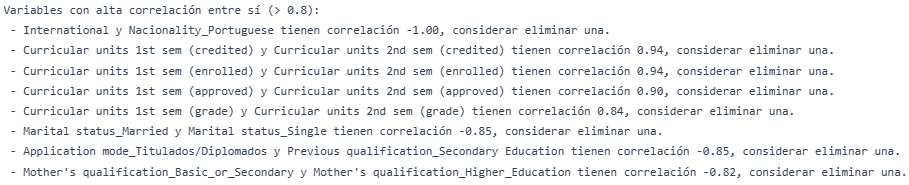
- Curricular units 1st sem (approved) y Curricular units 2nd sem (approved) tienen correlación 0.90, considerar eliminar una.

- Curricular units 1st sem (grade) y Curricular units 2nd sem (grade) tienen correlación 0.84, considerar eliminar una.

- Marital status\_Married y Marital status\_Single tienen correlación -0.85, considerar eliminar una.

- Application mode\_Titulados/Diplomados y Previous qualification\_Secondary Education tienen correlación -0.85, considerar eliminar una.

- Mother's qualification\_Basic\_or\_Secondary y Mother's qualification\_Higher\_Education tienen correlación -0.82, considerar eliminar una.



Para decidir qué variables se deben mantener para evitar problemáticas como la maldición de la dimensionalidad y la colinealidad en los modelos a realizar. Además, para mantener la variable que más información le proporciona a “*Target*” se elabora el análisis de correlaciones con la variable de salida.

**3.8 ANÁLISIS DE CORRELACIONES PARA IRRELEVANCIA**:

Para tomar la decisión de eliminar variables con baja correlación con la variable objetivo se toma un valor extremo de 0.01 ya que existen estudios realizados anteriormente sobre factores/variables que influyen en la deserción educativa, por lo que tomar el valor predeterminado 0.1 podría causar que no se analicen relaciones importantes. Por ejemplo, el nivel de estudio de los padres, que inicialmente no demostraba una buena correlación, ya ha sido estudiada en algunos artículos:

*"Factores socioeconómicos que con mayor frecuencia se han presentado en los diferentes casos, como la carencia de fuentes de financiamiento para continuar sus estudios, ya sea de la universidad o de entidades financieras, las deficientes bases económicas de la familia y la incompatibilidad laboral, proporcionan otro elemento significativo frente a las decisiones que toman los alumnos”* (Gasca, 2009).

*"Por su parte, la creciente vulnerabilidad socioeconómica se ve reflejada tanto en el nivel de ingresos familiares reportado por los nuevos estudiantes, como en la proporción de jóvenes que se encontraban trabajando al momento de presentar el Examen de Estado para Ingreso a la Educación Superior del ICFES, y el tipo de propiedad de la vivienda. El nivel educativo de la madre también ilustra una recomposición importante en este sentido (...). Se observa con claridad una correlación directa entre mayores ingresos y resultados académicos."*  (Guzmán, 2009)

Manteniendo la anterior idea, se obtienen los siguientes resultados:

Variables con baja correlación con el target (< 0.01):

- Educational special needs tiene correlación 0.00 con el target, considerar eliminarla. - Marital status\_Widower tiene correlación 0.00 con el target, considerar eliminarla.

- Nacionality\_Angolan tiene correlación 0.01 con el target, considerar eliminarla.

- Nacionality\_Brazilian tiene correlación 0.01 con el target, considerar eliminarla.

- Nacionality\_Cape Verdean tiene correlación 0.00 con el target, considerar eliminarla.

- Nacionality\_Guinean tiene correlación 0.01 con el target, considerar eliminarla.

- Nacionality\_Mexican tiene correlación 0.01 con el target, considerar eliminarla.

- Nacionality\_Russian tiene correlación 0.01 con el target, considerar eliminarla.

- Nacionality\_Spanish tiene correlación 0.00 con el target, considerar eliminarla.

- Nacionality\_Ukrainian tiene correlación 0.00 con el target, considerar eliminarla.

- Father's qualification\_Higher\_Education tiene correlación 0.00 con el target, considerar eliminarla.

- Father's qualification\_Technical\_Education tiene correlación 0.01 con el target, considerar eliminarla.

- Mother's occupation\_Professionals tiene correlación 0.01 con el target, considerar eliminarla.

- Father's occupation\_Unskilled Workers tiene correlación 0.01 con el target, considerar eliminarla.

Teniendo en cuenta lo anterior, se eliminan las siguientes variables, con su debida explicación:

* Mother's occupation y Father's occupation tienen alta correlación entre sí y tienen la misma correlación con la objetivo, por lo que se puede eliminar cualquiera de las dos. Se elimina Father's occupation
* Curricular units 1st sem (credited) y Curricular units 2nd sem (credited) tienen baja correlación con la variable objetivo, -0,029 y -0,033 respectivamente, a su vez tienen una alta correlación (0.94) entre ellas. Se mantiene Curricular units 2st sem (credited) se deja esa
* Curricular units 1st sem (enrolled) y Curricular units 2nd sem (enrolled) tienen correlación 0.94. Con la variable target, Curricular units 1st sem (enrolled) tiene una baja correlación -0,12 y Curricular units 2nd sem (enrolled) -0,141514926. Se elimina Curricular units 1st sem (enrolled)
* Curricular units 1st sem (approved) y Curricular units 2nd sem (approved) tienen alta correlación (0.90). Con la variable target, Curricular units 1st sem (approved) -0,479112034 y Curricular units 2nd sem (approved) -0,569500244. Se elimina Curricular units 1st sem (approved)
* Curricular units 1st sem (grade) y Curricular units 2nd sem (grade) tienen correlación 0.84. Con la variable de salida -0,480669065 y -0,571792203, por tanto se elimina Curricular units 1st sem (grade)
* International(0,0103596549088543) con target y Nacionality\_Portuguese -0,0103596549088547, se elimina International
* Marital status\_Married y Marital status\_Single tienen correlación alta -0.85. Con la variable objetivo tienen -0,0990478494184202 0,113858754518879 respectivamente. Se elimina Marital status\_Married
* Application mode\_Titulados/Diplomados tiene correlación -0,047451611988853 con 'target' y Previous qualification\_Secondary Education 0,064781704240618. Entre ambas variables hay correlación de -0.85. Se elimina Application mode\_Titulados/Diplomados
* Mother's qualification\_Basic\_or\_Secondary tiene correlación 0,0563642174849397 con 'target', Mother's qualification\_Higher\_Education 0,0121123249505718. Entre ambas variables hay correlación de -0.82. Se elimina Mother's qualification\_Higher\_Education

**3.9 REDUCCIÓN DE DIMENSIÓN**:

Para el caso del presente proyecto, se exponen algunas razones por las cuales no se aplicó PCA:

1. **Pérdida de la interpretabilidad**, en un caso como la deserción estudiantil, es importante entender **qué factores específicos** influyen en la deserción, y mediante políticas institucionales, educadores y administradores requieren identificar variables concretas.
2. El dataset tiene una **naturaleza de variables mixtas**: variables categóricas (estado civil, nacionalidad, ocupación de padres), variables numéricas (calificaciones, edad, tasas económicas)
3. Ya se ha realizado un **análisis de correlaciones para redundancia e irrelevancia**, por lo que se ha solucionado el problema de multicolinealidad.

**3.10 BALANCEO**:

**3.11 TRANSFORMACIONES:** (Ingeniería de características)

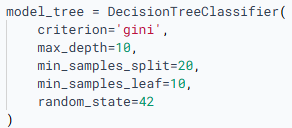
El balanceo y la ingeniería de características es realizada en el notebook de “*Creación modelos.ipynb*” debido a la diferencia en los requerimientos, expuestos en “*1.4 Diseño de solución*”

**4. MODELAMIENTO, EVALUACIÓN E INTERPRETACIÓN**

**4.1 CONFIGURACIÓN MÉTODOS DE MACHINE LEARNING**:

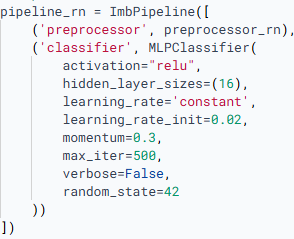
* **Modelos de aprendizaje supervisado:**

1. **Árbol de decisión:**



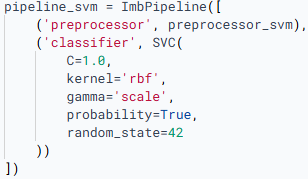
* **criterion='gini'**: Es más eficiente computacionalmente que entropy, y suele dar buenos resultados para clasificación binaria como en este caso (deserta / no deserta).
* **max\_depth=10**: Reduce el overfitting, un árbol muy profundo memoriza el entrenamiento, pero no generaliza bien.
* **min\_samples\_split=20 y min\_samples\_leaf=10**: Estas restricciones obligan al modelo a no dividir nodos a menos que haya suficientes datos, lo que evita divisiones irrelevantes en ramas pequeñas y reduce el riesgo de overfitting.
* **random\_state=42**: Semilla, para que los resultados sean reproducibles.

**2. Red Neuronal**



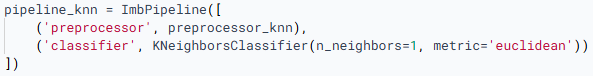
* **activation="relu"**: Por su eficiencia computacional y buen rendimiento para evitar un gradiente que desvanece. Además, funciona bien con relaciones no lineales.
* **hidden\_layer\_sizes=(16)**: 1 capa oculta con 16 neuronas. Sirve para capturar patrones relevantes sin llegar al sobreentrenamiento.
* **learning\_rate='constant'**: Brinda estabilidad al entrenamiento
* **learning\_rate\_init=0.02**: Tasa de aprendizaje para acelerar convergencia, ideal para los datos ya normalizados.
* **momentum=0.3**: Ayuda a acelerar el entrenamiento y evitar atascamiento en mínimos locales.
* **max\_iter=500**: Un máximo de suficientes ciclos de entrenamiento para la convergencia sin un entrenamiento excesivo.
* **random\_state=42**: Semilla, para la reproducibilidad del experimento.
* **verbose=False**: Evita la salida de logs durante el training.

**3. SVM**



* **kernel='rbf'**: Permite modelar relaciones no lineales entre las variables predictoras y la deserción. Esto es clave porque factores como edad, notas, becas y condiciones sociales no se combinan de forma lineal para predecir deserción.
* **C=1.0**: Valor moderado que busca equilibrio entre margen amplio y errores en entrenamiento. C más alto puede dar mejor ajuste, pero mayor overfitting.
* **gamma='scale'**: Se adapta correctamente a la varianza de los datos.
* **probability=True**: Se activa porque el modelo participa en soft voting, que necesita las probabilidades.

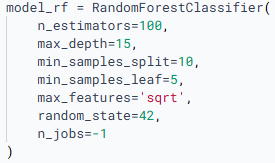
1. **KNN**



* **pipeline\_knn**: Contiene el preprocesador (preprocessor\_knn) y el clasificador KNN configurado con
* **n\_neighbors=1**: Es el número de vecinos más cercanos que el algoritmo KNN considera para clasificar o predecir una nueva instancia.
* **metric='euclidean'**: Es la función que mide la distancia entre dos puntos (instancias) en el espacio de características

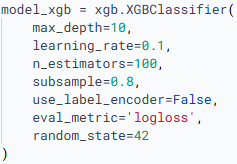
**Modelos de ensamble:**

1. **Random forest:**



* **n\_estimators=100**: Proporciona estabilidad al modelo sin ser demasiado costoso computacionalmente. En este problema con muchos predictores, se requieren varios árboles para capturar patrones robustos.
* **max\_depth=15**: Limita el crecimiento de cada árbol. Esto mejora la generalización.
* **min\_samples\_split=10, min\_samples\_leaf=5**: Evita que los árboles se vuelvan muy específicos de subconjuntos pequeños de datos, buscando eludir el overfitting.
* **max\_features='sqrt'**: Para cada división, solo se considera una raíz cuadrada del número total de características. Esto introduce aleatoriedad que disminuye la correlación entre árboles, lo cual mejora el rendimiento del ensamble.
* **n\_jobs=-1**: Utiliza todos los núcleos del procesador para acelerar el entrenamiento.

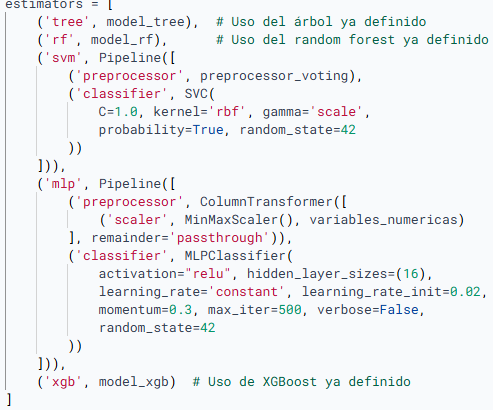
1. **XGBoost:**



* **Max\_depth**: Profundidad máxima de cada árbol de decisión en el modelo. Controla qué tan complejas pueden ser las reglas generadas por cada árbol.
* **Learning rate**: Controla cuánto contribuye cada árbol al modelo final. Es un factor de escalado para los pesos de las predicciones.
* **N\_estimators**: Número de árboles que se construirán en el modelo.
* **Subsample**: Fracción de datos de entrenamiento usados para entrenar cada árbol. Introduce aleatoriedad para evitar *overfitting*.
* **Use\_label\_encoder**: Indica si se debe usar un codificador automático de etiquetas.

#### **eval\_metric='logloss':** Métrica usada para evaluar el rendimiento durante el entrenamiento.

1. **Soft Voting:**



Combina las fortalezas de los distintos modelos. Algunos como SVC o redes neuronales capturan patrones no lineales; otros como árboles o random forest capturan relaciones jerárquicas e interacciones.

* **voting='soft'**: Usa las probabilidades de predicción de cada modelo, no solo las etiquetas. Esto permite hacer un promedio ponderado y suele ser más preciso y estable.

Se reutilizan modelos ya entrenados con buenas configuraciones y se integran en un pipeline mixto, con normalización para los modelos que lo requieren.

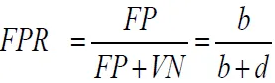
**4.2 ANÁLISIS DE MEDIDAS DE CALIDAD**:

Como se menciona en “*1.4* *DISEÑO DE SOLUCIÓN*” se usan las medidas de calidad:

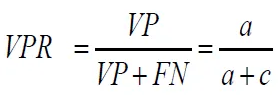
* **Accuracy:** Proporción de todas las clasificaciones que fueron correctas, ya sean positivas o negativas. [8] <https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall?hl=es-419>



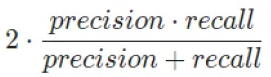
* **Área ROC:** Se calcula a partir de la tasa/razón (porcentaje) de verdaderos positivos y falsos positivos. Sólo funciona en clasificación binaria.
* **Definición de falsos positivos:**



* **Definición de verdaderos positivos:**



* **F1:** Combina precisión y cobertura, representa la calidad del modelo.

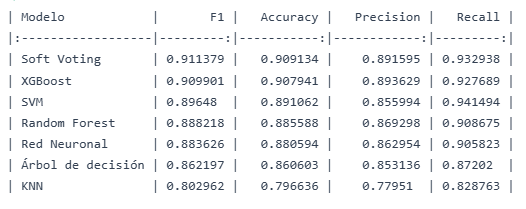


* **Recall:** Lo que el modelo fue capaz de identificar (cobertura), ayuda a controlar falsos negativos.

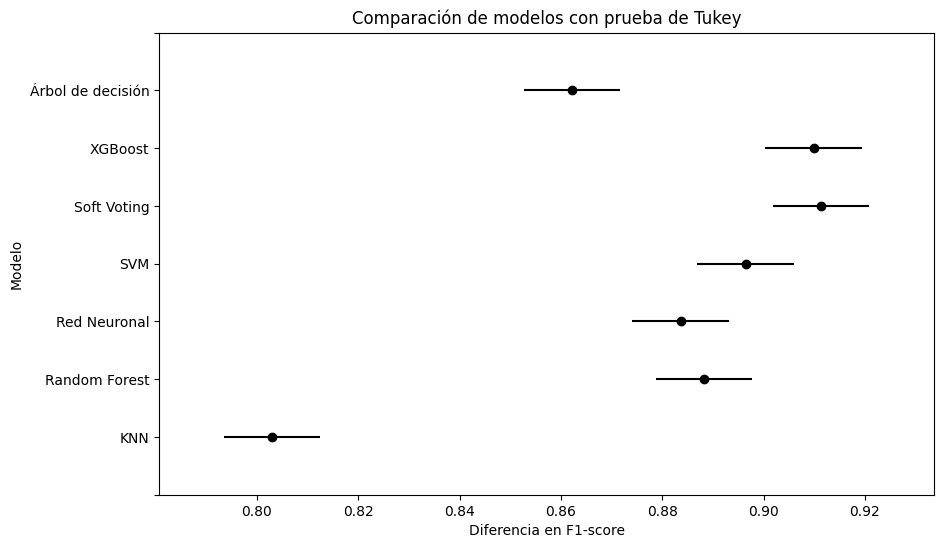


**4.3 SELECCIÓN DEL MEJOR MODELO**:

Para definir el modelo final del proyecto, se aplicó un proceso sistemático de evaluación y optimización. Inicialmente, se entrenaron y evaluaron siete modelos de clasificación utilizando validación cruzada estratificada. Las métricas consideradas fueron F1-score, accuracy, precisión y recall, lo que permitió establecer un primer ranking de desempeño.



Posteriormente, se realizó un análisis estadístico con ANOVA sobre los valores de F1-score. El resultado mostró un valor de **p < 0.05**, lo cual indica que existen diferencias significativas entre al menos algunos de los modelos evaluados. Para identificar exactamente cuáles modelos diferían entre sí, se aplicó una prueba post hoc de **Tukey HSD**.



Para determinar si existían diferencias significativas en el desempeño de los modelos, se aplicó un análisis ANOVA sobre los valores de F1-score obtenidos mediante validación cruzada.

* El resultado del ANOVA arrojó un valor de **p < 0.05**, lo que indica que **al menos un par de modelos tienen diferencias estadísticamente significativas** en su desempeño.
* Ante esto, se procedió con una **prueba post hoc de Tukey HSD** para identificar específicamente qué pares de modelos diferían entre sí.

#### **Resultados relevantes de la prueba de Tukey:**

* **KNN fue significativamente inferior** a todos los demás modelos (p-adj < 0.001).
* **XGBoost mostró diferencias significativas** respecto a modelos con menor rendimiento, como el Árbol de decisión, la Red Neuronal y KNN.
* **No se encontraron diferencias significativas** entre los modelos con mejor desempeño entre sí, como **XGBoost, SVM y Soft Voting**, lo cual sugiere un rendimiento estadísticamente comparable entre ellos.
* **Random Forest** se ubicó como un modelo intermedio, mostrando diferencias frente a algunos modelos (como Árbol de decisión), pero no frente a SVM.

Además del análisis estadístico y las métricas de rendimiento, se tuvo en cuenta el **costo computacional** de cada modelo, es decir, el tiempo de entrenamiento, la complejidad algorítmica y la escalabilidad en entornos reales.

#### **Criterios considerados:**

* **Tiempo de entrenamiento durante validación cruzada y optimización**
* **Demanda de recursos** (memoria y paralelización)
* **Robustez frente a sobreajuste**
* **Interpretabilidad y facilidad de despliegue**

Con base en estos criterios, se determinó que los siguientes modelos ofrecían el **mejor equilibrio entre desempeño predictivo y eficiencia computacional**:

* **XGBoost**:
  + Presentó el **mejor F1-score (~0.91)**
  + Mayor costo computacional, pero justificado por su precisión y capacidad de generalización
* **Random Forest**:
  + Desempeño competitivo con **bajo tiempo de entrenamiento**
  + Buena interpretabilidad y estabilidad
* **SVM**:
  + Buen rendimiento en tareas de clasificación binaria
  + Costo medio, especialmente en conjuntos de datos medianos

Tras seleccionar los tres modelos con mejor balance entre desempeño y costo computacional (XGBoost, Random Forest y SVM), se realizó un proceso de **optimización de hiperparámetros** utilizando dos enfoques complementarios:

#### **1. Búsqueda exhaustiva (GridSearchCV):**

* Se definieron espacios de búsqueda específicos para cada modelo.
* Se evaluaron múltiples combinaciones de parámetros utilizando validación cruzada estratificada.
* Esta técnica garantiza una exploración completa, aunque puede requerir mayor tiempo de cómputo.

#### **2. Optimización bayesiana (BayesSearchCV):**

* Se utilizó un enfoque probabilístico más eficiente para buscar configuraciones óptimas.
* Permite explorar espacios de búsqueda amplios con menor número de iteraciones.
* Resulta especialmente útil en modelos con múltiples hiperparámetros continuos.

### **Resultado de la Optimización**

Una vez comparados los mejores resultados obtenidos por cada modelo en ambas técnicas, se concluyó que:

**El mejor modelo fue XGBoost optimizado mediante GridSearch**,ya que alcanzó el **F1-score más alto**, manteniendo estabilidad y buen tiempo de ejecución.

Este modelo fue almacenado en un pipeline y preparado para la fase de despliegue.

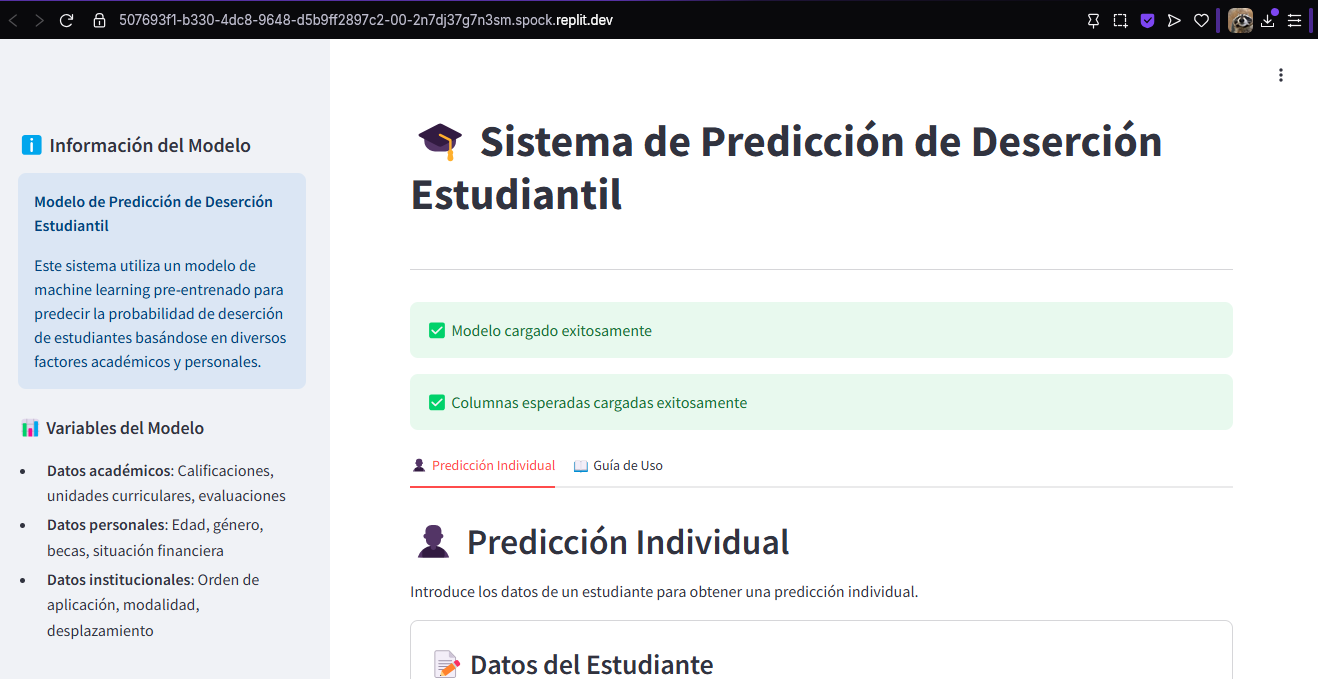
**5. DESPLIEGUE:**

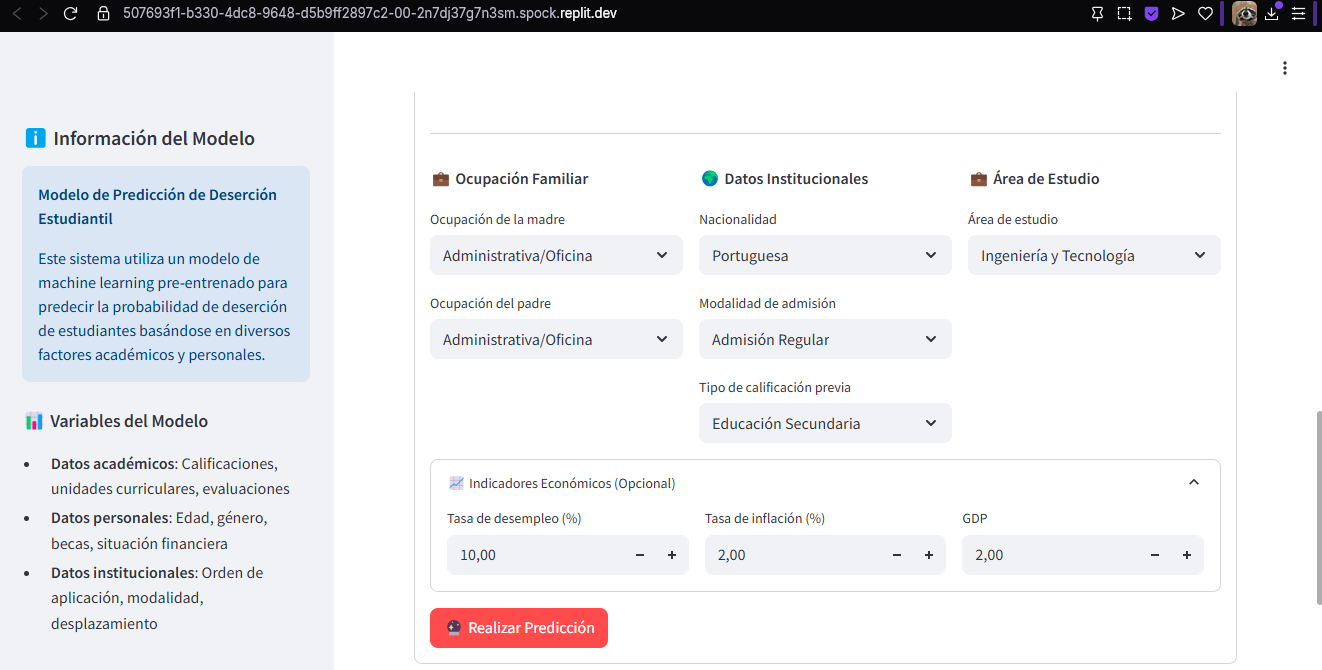
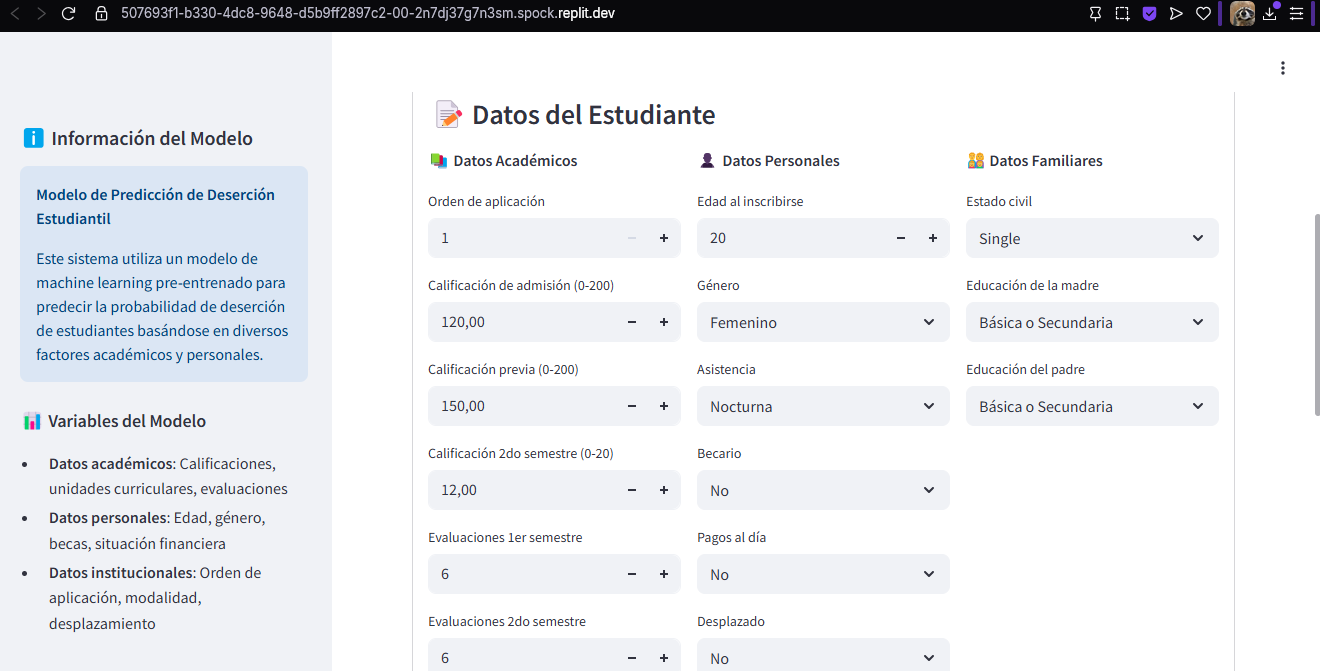
5.1 **PREDICCIÓN DE DATOS FUTUROS**:

Para el despliegue del modelo se construyó un pipeline completo que encapsula tanto las transformaciones de preprocesamiento como el clasificador XGBoost final. Este pipeline incluye el escalado selectivo de variables numéricas mediante MinMaxScaler y un ColumnTransformer, y el clasificador fue configurado con los hiperparámetros óptimos previamente encontrados mediante técnicas de optimización bayesiana.

Una vez entrenado con el conjunto final de datos, el pipeline fue almacenado utilizando la librería joblib en un archivo llamado pipeline\_final\_desercion.pkl, que puede ser cargado en cualquier momento sin necesidad de volver a entrenar. Para garantizar compatibilidad futura, también se almacenó la lista de columnas esperadas en columnas\_esperadas.pkl. Esta práctica facilita la integración en sistemas de predicción y su despliegue como aplicación web interactiva usando Streamlit, impulsado por la herramienta Replit, el despliegue puede ser ejecutado desde la siguiente URL:

<https://507693f1-b330-4dc8-9648-d5b9ff2897c2-00-2n7dj37g7n3sm.spock.replit.dev>





5.2 **MONITOREO**:

Para garantizar la vigencia del modelo predictivo, se recomienda un sistema de monitoreo que supervise continuamente el rendimiento del modelo, mediante el análisis de las métricas clave como F1-score, accuracy, recall y AUC-ROC. Además de la evaluación de posibles desviaciones en la distribución de variables de entrada, que podrían indicar un cambio de comportamiento (concept drift). Este monitoreo se puede integrar a futuro como una función dentro de la aplicación web de despliegue, alimentada por los registros de uso y los resultados reales obtenidos por las instituciones educativas.

5.3 **CRONOGRAMA DE MANTENIMIENTO**:

Dado que los datos estudiantiles se actualizan con cada semestre académico, se establece un cronograma de reentrenamiento semestral. Al finalizar cada período, se incorporarán los nuevos datos recolectados al conjunto histórico, se reentrenará el modelo con el top 3 de los mejores modelos (XGBoost, SVM, Random Forest) y se comparará su rendimiento con el modelo anterior. De ser favorable, el nuevo modelo será desplegado, manteniendo así la precisión y utilidad del sistema predictivo a lo largo del tiempo.

**Referencias bibliográficas**

Gasca, S. &. (2009). *Factores intervinientes en la deserción escolar de la Facultad de Psicología*. Obtenido de Fundación Universitaria Los Libertadores, Tesis Psicológica: https://www.redalyc.org/pdf/1390/139013586006.pdf

Grupo Banco Mundial. (2022). Obtenido de https://www.bancomundial.org/ext/es/home: https://www.bancomundial.org/ext/es/home

Guzmán, C. e. (2009). *Deserción estudiantil en la educación superior colombiana: Metodología de seguimiento, diagnóstico y elementos para su prevención*. Obtenido de https://www.mineducacion.gov.co/sistemasdeinformacion/1735/articles-254702\_libro\_desercion.pdf

Ministerio de Educación Nacional de Colombia. (2022). *Deserción escolar en Colombia: análisis, determinantes y política de acogida, bienestar y permanencia*. Obtenido de https://www.mineducacion.gov.co/1780/articles-363488\_recurso\_34.pdf

OECD. (2023). *Digital Tools for Education*. Obtenido de https://www.oecd.org

UNAM. (2021). Obtenido de https://www.unam.mx

UNESCO. (2022). *Educación superior en Iberoamérica*. Obtenido de www.unesco.org