Tecnológico de Monterrey

Maestría en Inteligencia Artificial

Equipo 14
A01794502 Lucero Guadalupe Contreras Hernández
A01686824 Manuel Alejandro Ambriz Baca
A01797072 Angel Adrian Morales Aldaco
A00739034 Alberto Cortés Murillo
A01795348 Myriam Beatriz Aguirre Pérez

Proyecto - Fase 1: Predicción de riesgo crediticio

Materia: Operaciones de aprendizaje automático

Repositorio principal:
https://github.com/A00739034/mlops-eq14

Notebook en Collaboratory:

https://github.com/ManuelAmbrizTec/riesgo_crediticio_equipo14/blob/main/Riesgo_crediticio_fase1_equipo14%20(1).ipynb

Video

https://youtu.be/SeTKOnNZbpg

12 de Octubre 2025

Análisis del problema: Predicción de riesgo crediticio

Introducción y objetivo

La gestión del riesgo crediticio es un elemento central en la banca. Su finalidad es estimar la probabilidad de que un prestatario incumpla sus obligaciones y cuantificar el impacto económico asociado a ese incumplimiento. Esto permite a las instituciones financieras fijar líneas de crédito, establecer provisiones regulatorias y definir precios (tasas de interés y comisiones) que cubran las posibles pérdidas. La **predicción del riesgo crediticio** consiste en elaborar modelos estadísticos o de aprendizaje automático que, a partir de información sobre el solicitante (demografía, historial de pagos, riqueza y estabilidad), estimen la probabilidad de impago y las pérdidas esperadas.

Para ejemplificar los conceptos técnicos de este documento, se utiliza el conjunto de datos **South German Credit**, un estrato de 1 000 operaciones de crédito otorgadas por un banco regional en Alemania entre 1973 y 1975. El conjunto contiene 700 operaciones clasificadas como “buen riesgo” y 300 como “mal riesgo” (Damrongkitkanwong, 2024). Incluye 20 características, la mayoría categóricas (estado de la cuenta corriente, duración del crédito, historial de crédito, propósito del préstamo, empleo, tipo de vivienda, etc.) y tres variables numéricas (monto del crédito, duración y edad) (Damrongkitkanwong, 2024). Aunque el ejemplo está basado en datos europeos, la metodología es adaptable a cualquier región siempre que el modelo se reentrene con datos locales.

El objetivo de este proyecto es construir un modelo de **aprendizaje automático para predecir el riesgo crediticio**, analizar su impacto para distintos tipos de instituciones (banca tradicional y fintech) y proponer métricas de negocio y de modelo que permitan evaluar su desempeño. Además, se discuten las implicaciones regulatorias, la integración de datos alternativos y las consideraciones de equidad.

Equipo del Proyecto

A continuación, se presenta a los integrantes del equipo, su rol y un resumen de sus contribuciones generales durante esta fase.

Lucero - Data Scientist: Lideró el análisis exploratorio de los datos, la interpretación de los resultados de los modelos y la extracción de conclusiones de negocio para entender el perfil de riesgo de los clientes.

Alberto - DevOps Engineer: Se encargó de la infraestructura de versionamiento, asegurando la trazabilidad y reproducibilidad de los datos, artefactos y experimentos mediante el uso de Git y DVC.

Manuel - Data Engineer: Fue responsable de la ingesta inicial, limpieza, validación y preparación del dataset crudo, transformándolo en un conjunto de datos limpio y estructurado.

Adrian - ML Engineer: Diseñó y ejecutó el pipeline de entrenamiento de los modelos, desde la creación del modelo base hasta la optimización de hiperparámetros del modelo avanzado.

Myriam - Software Engineer: Aseguró la calidad y cohesión del código del proyecto, estructurando el notebook, creando funciones de ayuda y garantizando que el flujo de trabajo fuera coherente y libre de errores.

1. Análisis de Requerimientos: documentado usando MLCanvas

Propuesta de valor

¿Qué estamos intentando hacer?
Desarrollar un modelo de Machine Learning capaz de predecir la probabilidad de que un solicitante de crédito cumpla o no con sus obligaciones financieras, clasificándose como “bueno” o “malo”.

¿Por qué es importante?
La predicción temprana del riesgo crediticio es crítica para las instituciones financieras, ya que:

- Reduce la probabilidad de préstamos incobrables (Non-Performing Loans – NPLs).
- Permite optimizar las provisiones de capital y mejorar la rentabilidad.
- Automatiza la evaluación crediticia, reduciendo tiempos de aprobación.
- Aumenta la inclusión financiera al permitir otorgar crédito responsable a clientes con poca información crediticia (thin-file).

¿Quién lo va a usar o estará impactado?

- **Analistas de riesgo y oficiales de crédito** (toman decisiones informadas con base en las predicciones).
- **Comités de crédito y dirección financiera** (monitorean el riesgo agregado y el desempeño de la cartera).
- **Clientes** (experimentan procesos de aprobación más rápidos y personalizados).

- **Unidades de innovación o fintechs** (aprovechan el modelo como núcleo de su motor crediticio automatizado).

Aprender

Fuentes de datos

¿Qué fuentes de datos en bruto podemos utilizar?

El conjunto de datos **South German Credit**, que contiene 1,000 registros históricos con 20 variables predictoras (cuentas bancarias, historial crediticio, monto solicitado, duración del préstamo, edad, estado civil, empleo, ahorros, etc.).
Variable objetivo: credit_risk (bueno = 1, malo = 0).

Recolección de datos

¿Cómo obtenemos nuevos datos de los que aprender? (tanto entradas como salidas)

Para este proyecto, el conjunto de datos ya está recolectado. En un sistema en producción, la recolección sería un proceso continuo:

- Entradas: Cada nueva solicitud de crédito que se procesa se convierte en un nuevo punto de datos con sus respectivas características.
- Salidas (Etiquetas): El resultado del préstamo (si fue pagado a tiempo o si entró en default) se registra a lo largo del tiempo. Este ciclo de retroalimentación es fundamental para reentrenar y actualizar el modelo periódicamente.

Características

Entradas al modelo a conseguir a partir de las fuentes de datos en bruto

Las 20 variables predictoras del conjunto de datos South German Credit, que incluyen:

- Financieras: Estado de la cuenta corriente (status), historial de crédito (credit_history), monto del crédito (amount), ahorros (savings).
- Laborales y personales: Duración del empleo (employment_duration), tipo de empleo (job), edad (age), estado personal y sexo (personal_status_sex).
- Relacionadas con el crédito: Duración del préstamo (duration), propósito (purpose), tasa de cuota (installment_rate).
- Contextuales: Si es trabajador extranjero (foreign_worker), si tiene teléfono (telephone).

Construcción de modelos

¿Cuándo creamos o actualizamos los modelos con nuevos datos de entrenamiento? ¿De cuánto tiempo disponemos?

Entrenamiento inicial con el dataset limpio (Fase 1).
Actualizaciones periódicas planificadas al incorporar nuevas observaciones.
Tiempo estimado de entrenamiento: minutos a horas, según complejidad del modelo y tamaño del dataset.

Predecir

Tareas Machine Learning

Cuáles son las entradas, cuál la salida a predecir y qué tipo de algoritmo (clasificación, regresión…)

- Entradas: Un vector con las características del solicitante de crédito (las 20 variables mencionadas).
- Salida a predecir: Una predicción binaria: "buen pagador" (0) o "mal pagador" (1). Más importante aún, el modelo debe generar una probabilidad de incumplimiento calibrada (un valor entre 0 y 1).
- Tipo de algoritmo: Clasificación binaria supervisada. Se evaluarán algoritmos como Regresión Logística (como línea de base) y modelos más avanzados como XGBoost y LightGBM, que son conocidos por su alto rendimiento en datos tabulares.

Decisiones

Cómo se usan las predicciones para tomar las decisiones que aportan valor

- Aprobar, rechazar o revisar manualmente una solicitud de crédito.
- Ajustar la tasa de interés o los límites de crédito según el riesgo.
- Segmentar clientes por nivel de riesgo para campañas personalizadas o gestión preventiva.

Hacer predicciones

Cuándo hacemos las predicciones sobre nuevas entradas y de cuánto tiempo disponemos

- Cuándo: Las predicciones se realizan en tiempo real, en el momento en que un cliente completa y envía una solicitud de crédito a través de los canales del banco (online, sucursal, etc.).
- Tiempo disponible: El tiempo de respuesta debe ser muy bajo (de segundos a pocos minutos) para no afectar negativamente la experiencia del cliente y mantener la eficiencia del proceso.

Evaluación offline

De qué métricas y predicciones disponemos para evaluar el modelo antes del paso a producción

Estas métricas evalúan la capacidad del modelo para distinguir correctamente entre solicitantes de alto y bajo riesgo.

- PR-AUC (Área bajo la Curva de Precisión-Recall): Será nuestra métrica de discriminación principal. A diferencia del ROC-AUC, el PR-AUC es mucho más informativo y fiable en conjuntos de datos con clases muy desbalanceadas, ya que se enfoca en el rendimiento sobre la clase minoritaria (los "malos pagadores"), que es la que más nos interesa identificar correctamente.
- Recall (Sensibilidad) para la clase "malo": Mide la capacidad del modelo para identificar a todos los que realmente incumplirán. Esta métrica es de máxima prioridad, ya que el costo financiero de un Falso Negativo (aprobar a un cliente que terminará en default) es significativamente mayor que el de un Falso Positivo.
- Puntuación F1 (F1-Score): Es la media armónica de Precisión y Recall. Proporciona una única puntuación que equilibra la necesidad de capturar a los malos pagadores (Recall) sin generar una cantidad excesiva de falsas alarmas (Precisión). Es ideal para una evaluación global del balance del modelo.

Evaluar
Monitorización y evaluación en vivo

De qué métricas y predicciones disponemos para evaluar el modelo ya en producción

Una vez en producción, el rendimiento se monitorea a través de dos lentes:

- Métricas de negocio (KPIs):
    - Ratio de Préstamos No Productivos (NPL Ratio): El indicador clave que se espera reducir.
    - Costo del Riesgo (Cost of Risk): Mide el costo de las provisiones que el banco debe hacer.
    - Retorno de la Inversión (ROI): Compara el beneficio financiero (pérdidas evitadas) con el costo de desarrollo y mantenimiento del proyecto.
- Métricas de rendimiento del modelo:

- Se monitorea la precisión, PR-AUC y otras métricas técnicas sobre los nuevos datos para detectar una posible degradación del modelo (model drift).
- Se analizan las distribuciones de las variables de entrada para identificar si están cambiando con el tiempo, lo que podría afectar el rendimiento del modelo.

2. Manipulación y preparación de datos

Esta fase consistió en la ingesta del dataset crudo (`german_credit_modified.csv`) y la aplicación de las primeras capas de limpieza para corregir errores estructurales y de contenido.

Contribuciones del equipo:

- Manuel (Data Engineer): Escribió el código para cargar los datos, forzar los tipos a numérico, y utilizó los diccionarios `VALID` y `RANGE` para identificar y marcar como nulos todos los valores que no cumplían con las reglas de negocio (códigos inválidos, edades o montos fuera de rango).
- Myriam (Software Engineer): Creó la función `brief()` para generar resúmenes rápidos y estandarizados del estado del Data Frame en cada paso, mejorando la legibilidad y el monitoreo del proceso de limpieza.

	laufkont	laufzeit	moral	verw	hoehe	sparkont	beszeit	rate	famges	buerge	...	verm	alter	weitkred	wohn	bishkred	beruf	pers	telef	gastarb	kredit
0	1.0	18.0	4.0	2.0	1049.0	1.0	2.0	4.0	2.0	1.0	...	2.0	21.0	3.0	1.0	1.0	3.0	2.0	1.0	2.0	1.0
1	1.0	9.0	4.0	0.0	2799.0	1.0	3.0	2.0	3.0	1.0	...	1.0	36.0	3.0	1.0	2.0	3.0	1.0	1.0	2.0	1.0
2	2.0	12.0	2.0	9.0	841.0	2.0	4.0	2.0	2.0	1.0	...	1.0	23.0	3.0	1.0	1.0	2.0	2.0	1.0	2.0	1.0

3 rows × 21 columns

Celda de verificación previa al EDA

```
path = "/content/drive/MyDrive/OperacionesML/Fase1/processed/german_credit_clean.csv"

df = pd.read_csv(path)
print("Shape:", df.shape)
print("Columnas:", df.columns.tolist())
print("\nValores nulos por columna (top 10):")
print(df.isna().sum().sort_values(ascending=False).head(10))
print("\nDistribución target_bad:")
print(df["target_bad"].value_counts(normalize=True).round(3))
```

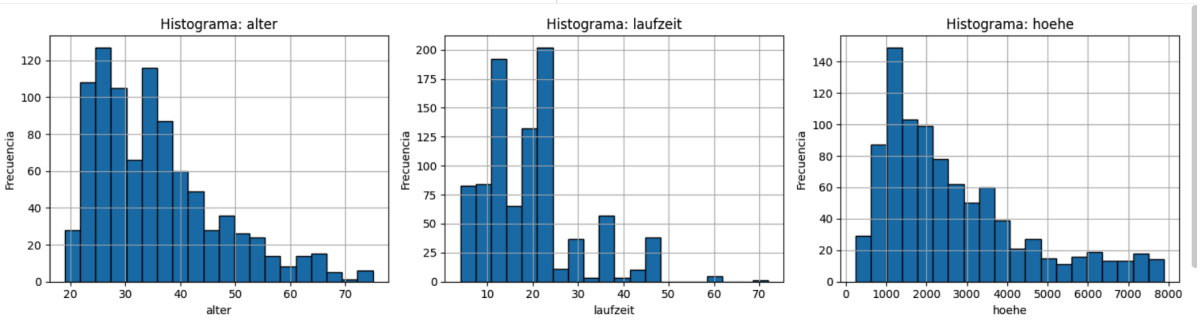
```
Shape: (923, 21)
Columnas: ['laufkont', 'laufzeit', 'moral', 'verw', 'hoehe', 'sparkont', 'beszeit', 'rate', 'famges', '

Valores nulos por columna (top 10):
target_bad    23
laufzeit       0
moral          0
verw           0
laufkont       0
hoehe          0
sparkont       0
rate           0
beszeit        0
buerge         0
dtype: int64

Distribución target_bad:
target_bad
0.0    0.717
1.0    0.283
Name: proportion, dtype: float64
```

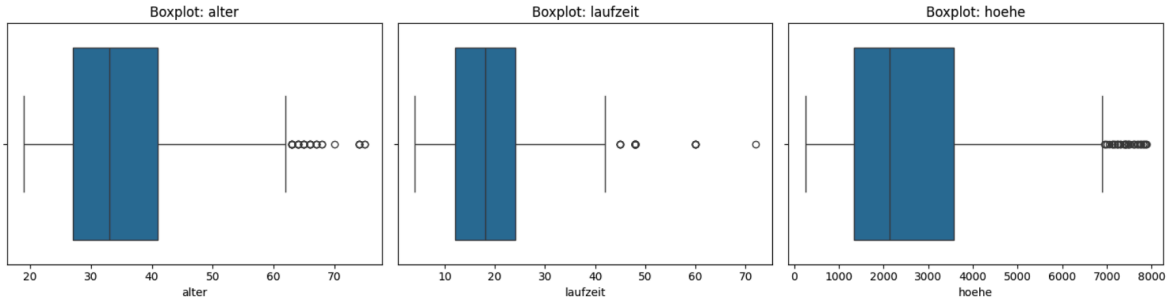
3. Exploración y preprocesamiento de datos

Una vez con un dataset estructuralmente limpio, se procedió a realizar un Análisis Exploratorio de Datos (EDA) para entender las características de los solicitantes y se aplicaron las transformaciones finales para preparar los datos para el modelado.

Contribuciones del equipo:

- Lucero (Data Scientist): Realizó el EDA, generando e interpretando los histogramas, boxplots y tablas de frecuencia para construir el "perfil del solicitante típico". También dirigió el análisis de outliers con el método IQR y documentó las conclusiones clave para el modelo.
- Manuel (Data Engineer): Implementó el código para la imputación de valores nulos (usando mediana y moda) y la eliminación de duplicados exactos y conflictivos, entregando un dataset 100% completo.
- Adrian (ML Engineer): Escribió el código para la transformación final de la variable objetivo (`kredit` a `target_bad`), el escalado de las variables numéricas con `StandardScaler`, y la división de los datos en conjuntos de entrenamiento y prueba.


Histograma: alter
Frecuencia
alter
Histograma: laufzeit
Frecuencia
laufzeit
Histograma: hoehe
Frecuencia
hoehe


Boxplot: alter
alter
Boxplot: laufzeit
laufzeit
Boxplot: hoehe
hoehe

```
{
  "alter": {
    "low": 18.0,
    "high": 62.0,
    "outliers": 34
  },
  "laufzeit": {
    "low": 4.0,
    "high": 42.0,
    "outliers": 44
  },
  "hoehe": {
    "low": 250.0,
    "high": 6915.75,
    "outliers": 36
  }
```


sparkont	
1.0	568
5.0	160
2.0	93
3.0	57
4.0	45

```python
# --- Imputación medianas para continuas, moda para categóricas ---
for col in df2.columns:
    if col == "target_bad":
        continue
    if col in cont_cols:          # continuas: mediana
        df2[col] = df2[col].fillna(df2[col].median())
    else:                         # categóricas: moda
        moda = df2[col].mode(dropna=True)
        if not moda.empty:
            df2[col] = df2[col].fillna(moda.iloc[0])
```

Muestra de código: Se rellenaron los valores nulos con la mediana para continuas y moda para categóricas.

```python
# --- Split ---
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.25, stratify=y, random_state=RANDOM_STATE
)
print("Train:", X_train.shape, " Test:", X_test.shape)
print("Distribución y:", y.value_counts(normalize=True).round(3).to_dict())
```

El 25% de los datos se utilizaron para la evaluación final.

4. Versionado de datos

Para garantizar la reproducibilidad y trazabilidad del proyecto, se implementó un sistema de versionamiento utilizando Git para el código y DVC para los datos y artefactos.

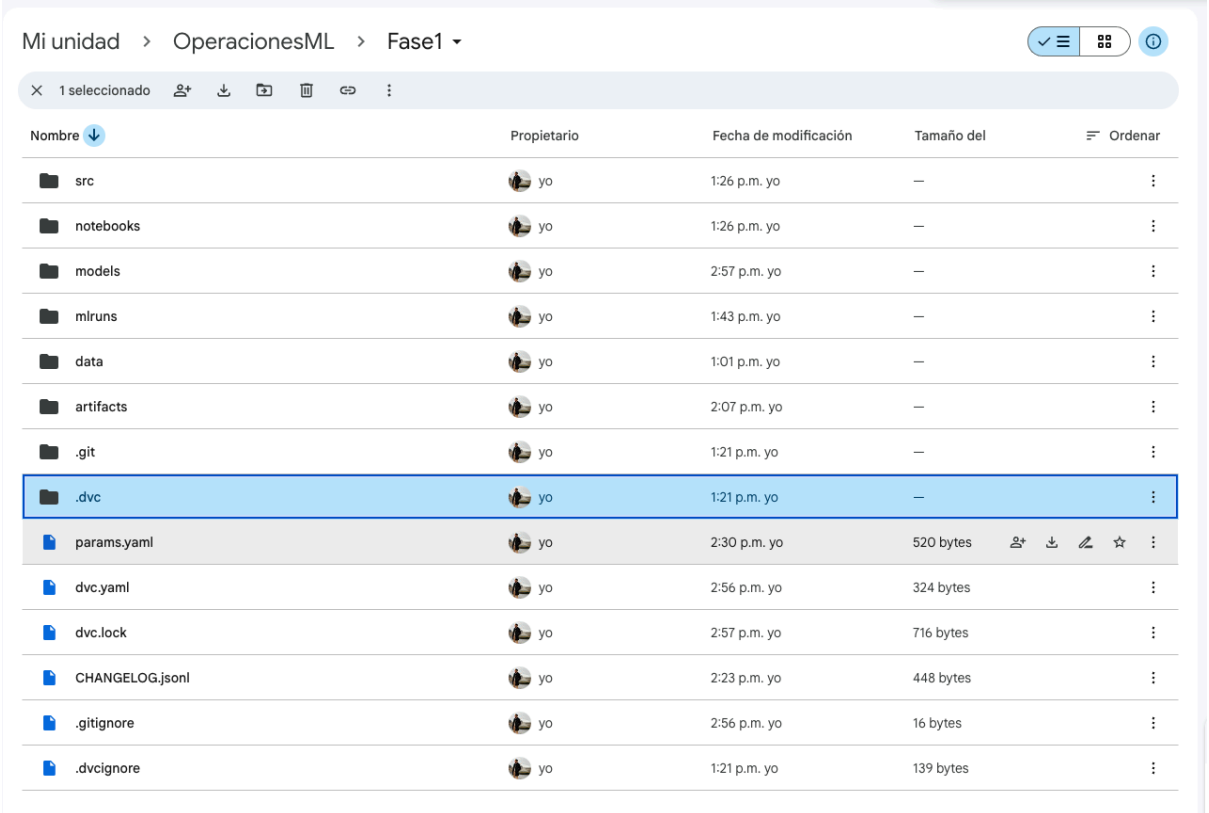
Contribuciones del equipo:

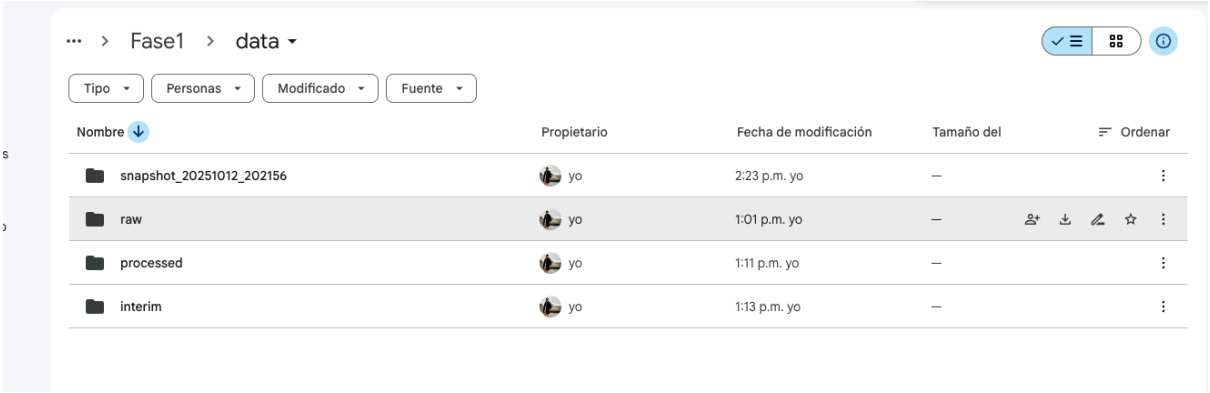
- Manuel (DevOps Engineer): Inicializó `Git` y `DVC` en el repositorio, definió la estrategia de versionamiento y ejecutó los comandos (`dvc add, git commit`) para versionar el dataset final. También configuró el `CHANGELOG.jsonl` y la Model Card como parte del pipeline de documentación.
- Myriam (Software Engineer): Escribió el código Python que prepara los "snapshots" de datos en carpetas únicas (`snapshot_{RUN_ID}`), facilitando el trabajo de versionamiento de Alberto.

Se adjunta evidencia de las ejecuciones que se realizaron para configurar el dvc

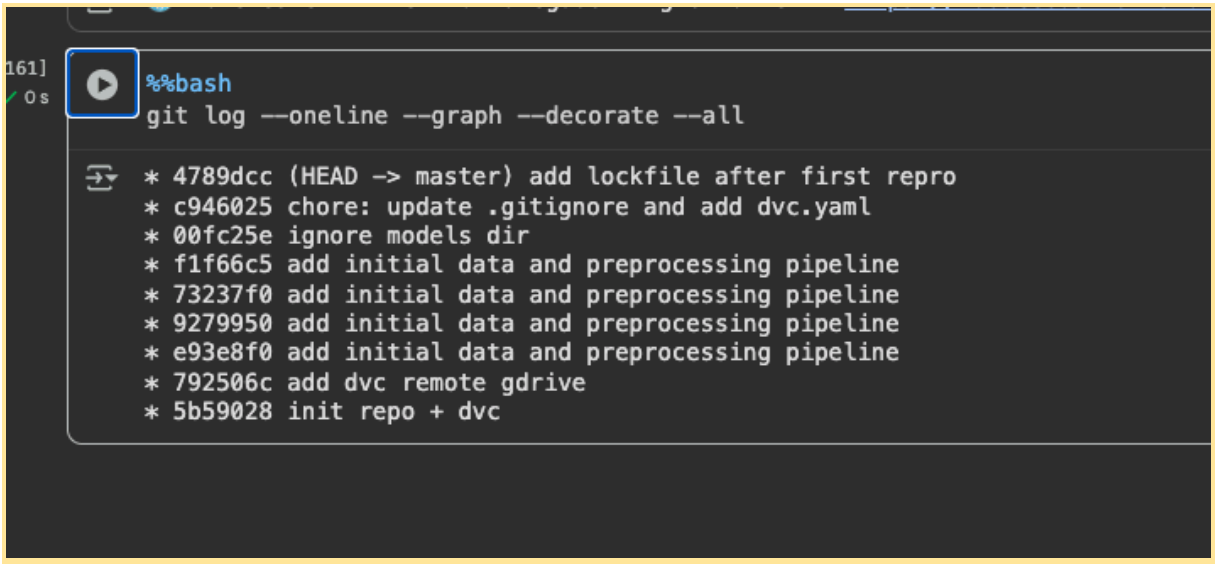
```
%%bash
dvc push

Process is terminated.
```

```
%%bash
git add dvc.lock
git commit -m "add lockfile after first repro"

[master 4789dcc] add lockfile after first repro
 1 file changed, 30 insertions(+)
 create mode 100644 dvc.lock
```

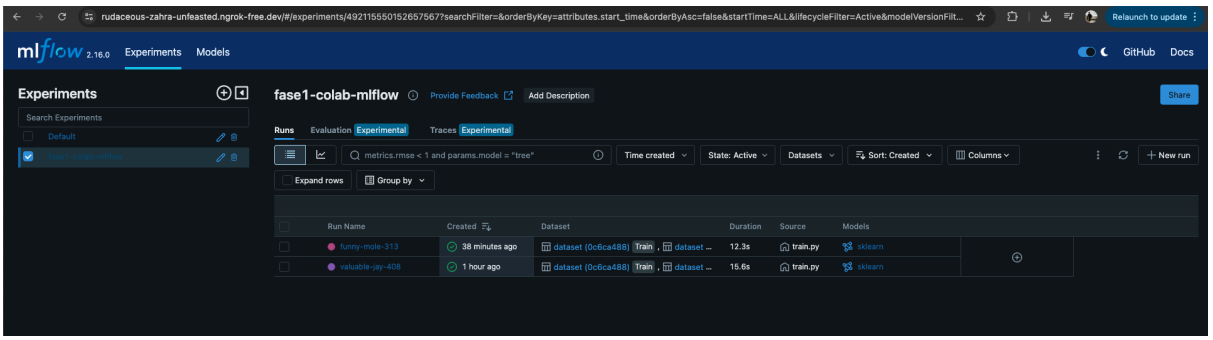




Cabe mencionar que todo por el momento se ha realizado en google colab y drive,



Por otro lado, también se ha configurado el primer avance para la ejecución de MLFlow.



Para esto al igual usamos google Colab, y para exponerlo a internet usamos la herramienta ngrok.

5. Construcción, ajuste y evaluación de modelos de Machine Learning

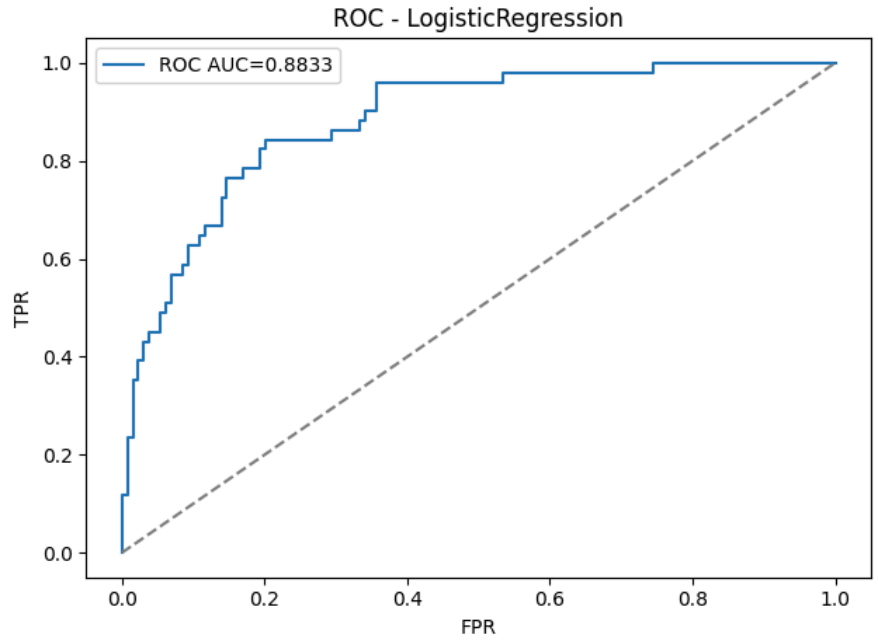
En esta fase final, se construyó un modelo base para establecer un punto de referencia y un modelo avanzado que fue optimizado para obtener el mejor rendimiento posible.

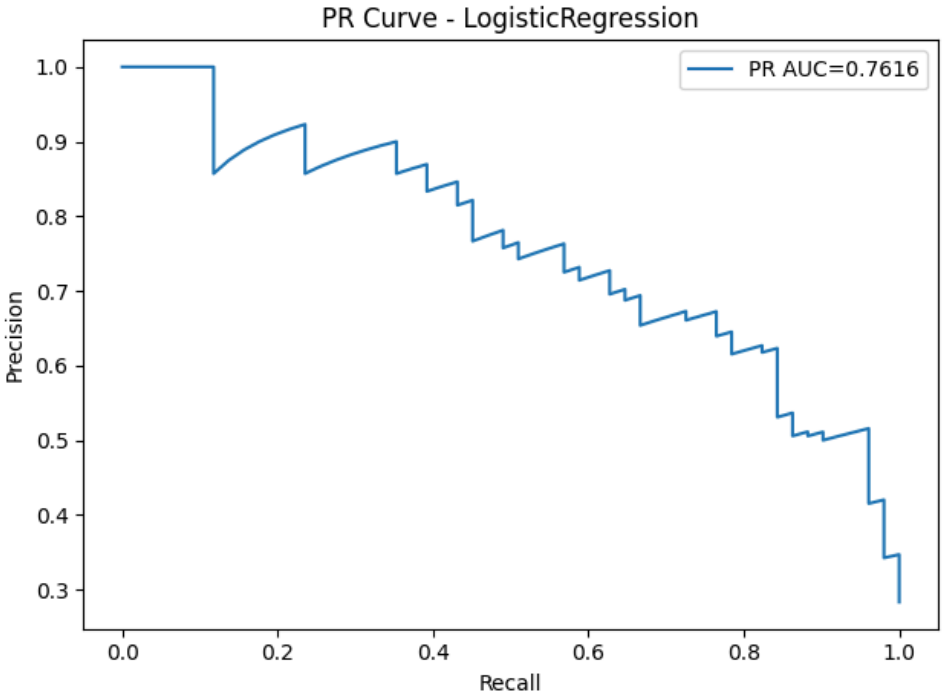
Contribuciones del equipo:

- Manuel (ML Engineer): Implementó el código para entrenar y evaluar tanto la Regresión Logística (baseline) como el Random Forest. Configuró y lanzó el `GridSearchCV` para encontrar los mejores hiperparámetros y guardó los modelos finales (`.joblib`).
- Lucero (Data Scientist): Analizó en profundidad las métricas y las curvas de rendimiento (ROC y PR) de ambos modelos. Fue responsable de redactar las conclusiones, interpretar el "trade-off" de Precision-Recall y declarar el modelo ganador basado en la evidencia.
- Alberto (DevOps Engineer): Se aseguró de que todos los artefactos de esta fase (modelos, métricas en JSON, figuras) se guardaran en la carpeta de ejecución única (`run_{RUN_ID}`) para mantener la trazabilidad.

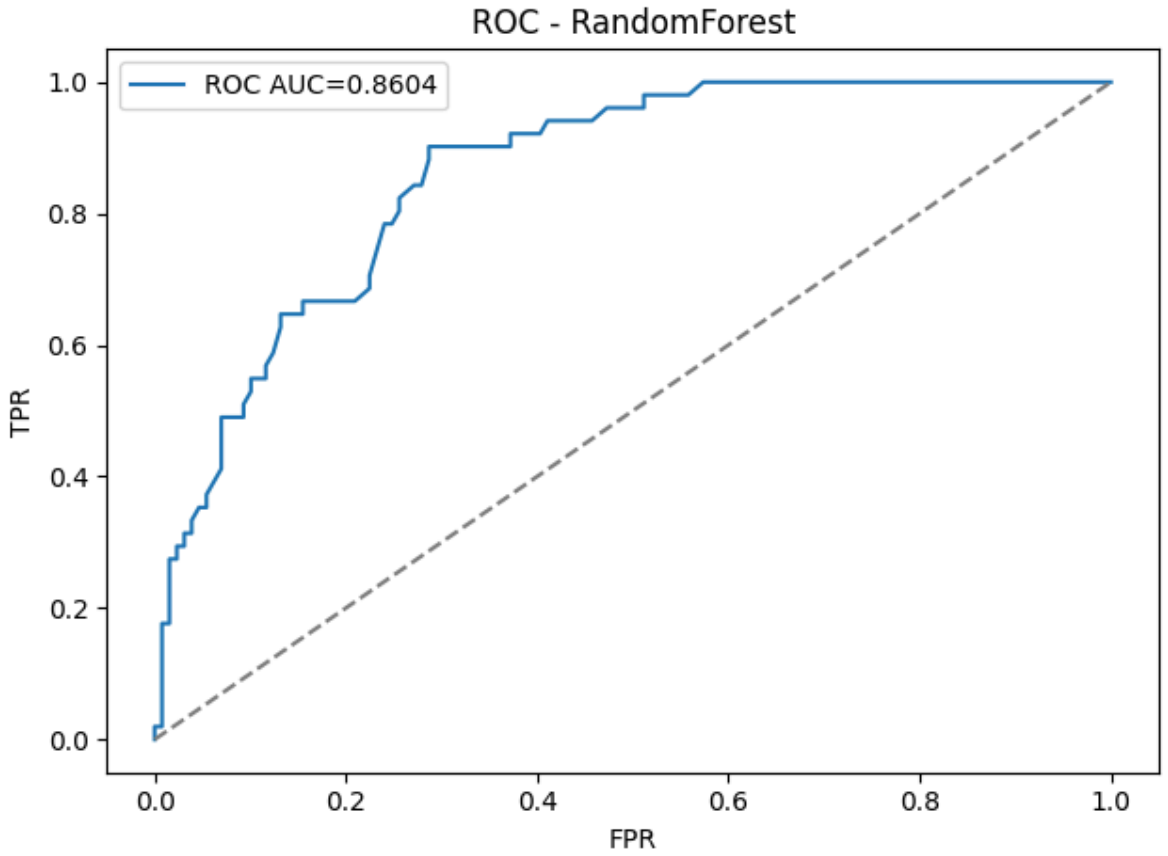
Imágenes

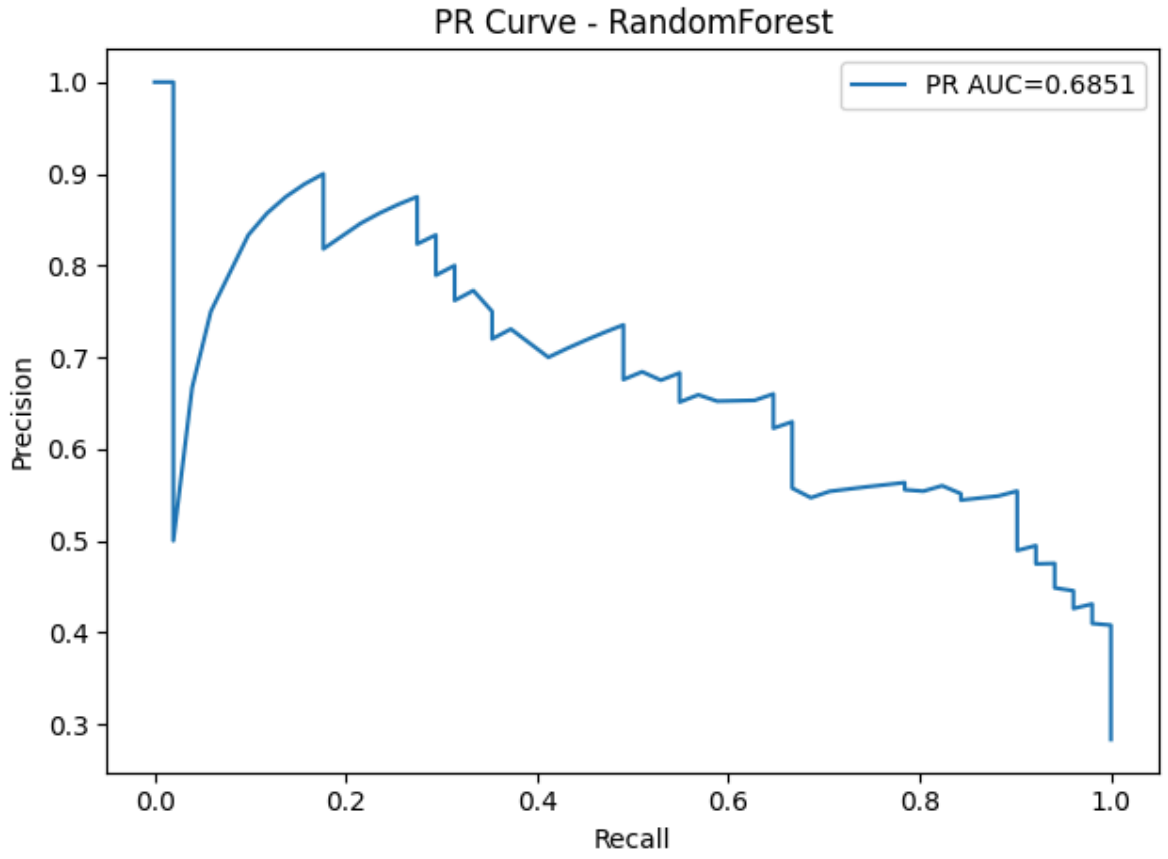
1. La **Curva ROC** y la **Curva Precision-Recall** del modelo de Regresión Logística.





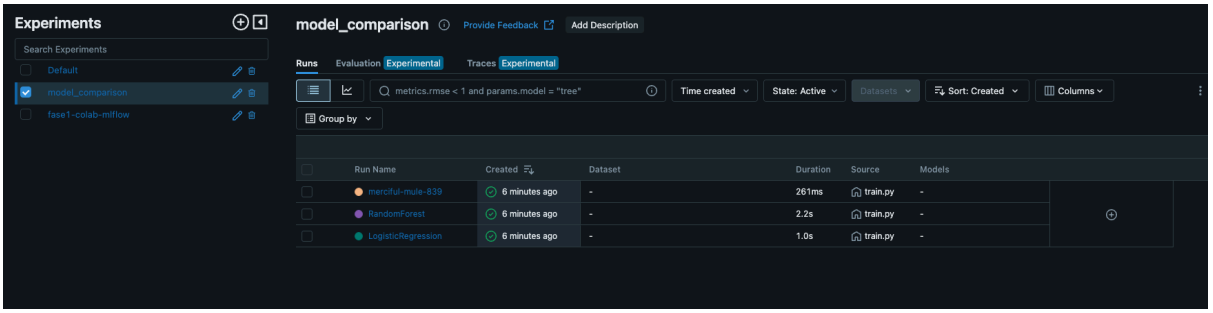
2. La **Curva ROC** y la **Curva Precision-Recall** del modelo Random Forest.





Tablas

1. La **tabla final de comparación** de modelos que muestra el `roc_auc` y `pr_auc` de cada uno y declara al ganador.



Conclusiones

En la primera fase del proyecto se permitió comprender de manera integral el proceso de predicción de riesgo crediticio desde la preparación de los datos hasta la evaluación de los modelos. A través del uso del conjunto de datos South German Credit, se lograron aplicar técnicas de limpieza, análisis exploratorio y modelado, demostrando la importancia de una correcta ingeniería de datos para obtener resultados confiables y reproducibles. Este enfoque garantizó la calidad del conjunto de datos y sentó las bases para un modelado sólido y escalable.

La comparación entre modelos evidenció que las técnicas basadas en árboles, como Random Forest, ofrecen un adecuado equilibrio entre precisión y sensibilidad frente a otros modelos predictivos, especialmente en escenarios con clases desbalanceadas. Sin embargo, se observó que el rendimiento del modelo de regresión lineal presentó una precisión mayor en las métricas elegidas, como el recall para los clientes de alto riesgo y el área bajo la curva PR (PR-AUC).

Con el resultado del proyecto, se resalta la relevancia del versionamiento y la trazabilidad en entornos de ciencia de datos aplicada. El uso de herramientas como Git y DVC permitió asegurar la reproducibilidad de los experimentos y la transparencia del proceso, aspectos fundamentales para proyectos en instituciones financieras donde la gobernanza de modelos y el cumplimiento regulatorio son críticos.

Finalmente, la implementación de este modelo abre oportunidades para futuras mejoras, como la integración de datos alternativos, el monitoreo del model drift en producción y la adopción de técnicas de aprendizaje más avanzadas. Con ello, las instituciones financieras podrán no solo reducir pérdidas por incumplimiento, sino también promover una inclusión crediticia más justa, eficiente y basada en evidencia.

Referencias

Anzai, Y. (2012). Pattern Recognition and Machine Learning. Morgan Kaufmann.

Damrongkitkanwong, A. (2024). Cost-sensitive Classification of Credit Risk [Dataset]. In Medium. https://medium.com/@aumdamrong/cost-sensitive-classification-of-credit-risk-8b0ecc87ceb8

Géron, Aurélien (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd Edition. O'Reilly.

Jafari, R. (2022). Hands-On Data Preprocessing in Python. Packt Publishing.

Kumar M, S., & Ahmed, U. (2020). Hands-On Exploratory Data Analysis with Python. Packt Publishing.

Lauchande N. (2021). Machine Learning Engineering with MLFlow. O'Reilly Media.

McKinney, W. (2022). Python for Data Analysis (3rd ed.). O´Reilly Media.

Peck, E. A., & Vining, G. G. (2021). *Introduction to linear regression analysis / Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining* (Sixth edition). John Wiley & Sons, Inc.

Zare, M. (2025). Forecasting market returns using machine learning: evidence from Random Forest models. *Applied Economics Letters*, 1–5. https://doi.org/10.1080/13504851.2025.2567614