TRABAJO FINAL BIG DATA

- **Integrantes**: Lucas Bole, Franco Lamperti, Facundo Macagno, Asuncion Leonard y Bautista Zanero

1. **Selección del Dataset asignado**: **Grupo 6 - Credit Card Fraud Detection Dataset 2023**

2. Definición del Problema: Detección de Fraudes en Transacciones de Tarjetas de Crédito

El problema a resolver consiste en identificar transacciones potencialmente fraudulentas dentro de un conjunto masivo de operaciones con tarjetas de crédito. Nuestro objetivo es predecir si una transacción es fraudulenta o legítima, utilizando variables asociadas al comportamiento del cliente, como el monto y otros atributos de la operación.

La detección temprana de fraudes es muy importante para las entidades financieras porque permite reducir pérdidas económicas derivadas de cargos fraudulentos, proteger la reputación y la confianza del cliente en el sistema financiero, y optimizar recursos de los equipos de prevención priorizando alertas con mayor probabilidad real de fraude.

En resumen, buscamos construir un modelo predictivo basado en Big Data que aprenda de los patrones históricos y logre distinguir, en tiempo real, operaciones legítimas de aquellas con comportamiento anómalo, contribuyendo a una gestión más eficiente del riesgo financiero.

3. Análisis Exploratorio de Datos:

Objetivo del EDA

Explorar el dataset *creditcard_2023* para entender la distribución de las variables, detectar outliers, valores nulos y observar diferencias entre transacciones fraudulentas y legítimas.

Descripción del Conjunto de Datos:

El conjunto de datos utilizado corresponde a transacciones con tarjetas de crédito realizadas por titulares europeos durante el año 2023. Cada registro representa una operación individual e incluye variables numéricas y categóricas que describen distintos aspectos de la transacción.

En total, el dataset contiene más de 550.000 registros, y los datos han sido anonimizados para proteger la identidad de los titulares. El objetivo principal de este conjunto es facilitar el desarrollo de algoritmos de detección de fraude, permitiendo

identificar operaciones potencialmente fraudulentas a partir de patrones históricos de comportamiento.

Luego de cargar el archivo con las transacciones de tarjetas de crédito en Orange, pudimos ver a través de una primera *Data Table* que el dataset cuenta con 568.630 instancias, 31 features y ninguna variable Target.

Features (31)

- ➔ **id**: Identificador único para cada transacción
- ➔ **V1-V28**: Variables anónimas que representan diversos atributos de una transacción (por ejemplo, tiempo, ubicación, etc.)
- ➔ **Amount**: Monto de transacción en unidades monetarias
- ➔ **Class**: Variable objetivo binaria indicando si la transacción es fraudulenta (1) o legítima (0)

Uso

- **Detección de Fraude con Tarjetas de Crédito**: Construir un modelo de aprendizaje (machine learning) para detectar y prevenir fraudes con tarjetas de crédito identificando transacciones sospechosas y prevenir fraudes en tiempo real basadas en las características proporcionadas.

Otras posibles aplicaciones del dataset:

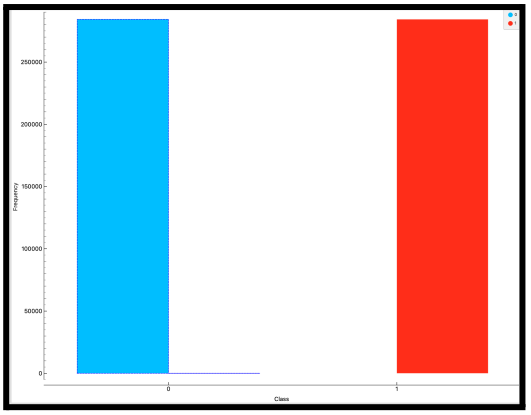
- **Análisis por categoría de comercio**: Evaluar si ciertos rubros o tipos de comercio presentan una mayor incidencia de fraudes.
- **Análisis por tipo de transacción**: Determinar si existen modalidades de operación más vulnerables al fraude que otras.

Análisis de Balance de Clases

Con el widget *Distributions* analizamos la variable objetivo (*Class*) la cual mostró un balance perfecto de clases.

De un total de 568,630 registros, la distribución es la siguiente:

- Clase 0 (No Fraudulenta): 284,315 registros (50.0%)
- Clase 1 (Fraudulenta): 284,315 registros (50.0%)

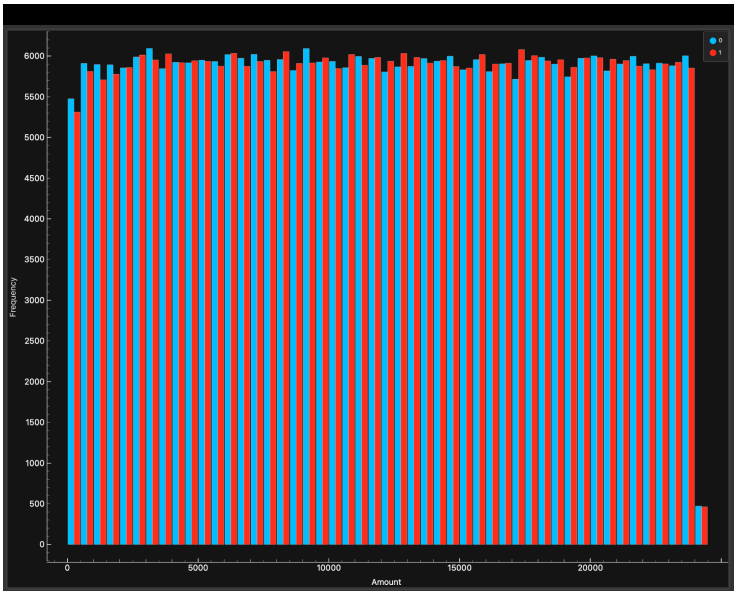


Formulacion de Hipotesis

1. Amount: Las transacciones fraudulentas a menudo tienen un patrón de monto diferente: o son muy pequeños (para probar la tarjeta) o muy grandes (para maximizar el beneficio), lo que crea una distribución distinta a las transacciones legítimas.
2. Features Vx: Estas features (V1, V2… V28) son resultado de una transformación de PCA (Análisis de Componentes Principales) que combina información clave (ubicación, tiempo, frecuencia, etc.). La hipótesis es que un subconjunto de estas features mostrará una diferencia de distribución o de dispersión mucho más marcada y significativa entre la Clase 1 y la Clase 0 que el 'Amount'.

Apoyo Gráfico y Estadístico para comprobar la hipótesis

Amount por Class (Distributions)



Hallazgo Clave (Refutación de Hipótesis)

- Las barras azules (Clase 0: No Fraude) y las barras rojas (Clase 1: Fraude) están casi perfectamente alineadas, desde el inicio hasta el final de la escala.
- Esto significa que la distribución del monto de las transacciones fraudulentas es casi idéntica a la distribución de las transacciones legítimas.
- Por lo tanto, la feature *Amount* es irrelevante para predecir el fraude. El monto de la transacción por sí solo no ayuda a distinguir si es fraudulenta o no, por lo que la detección de fraude dependerá en gran medida de las variables Vx.

Variables más importantes para la predicción (Rank)

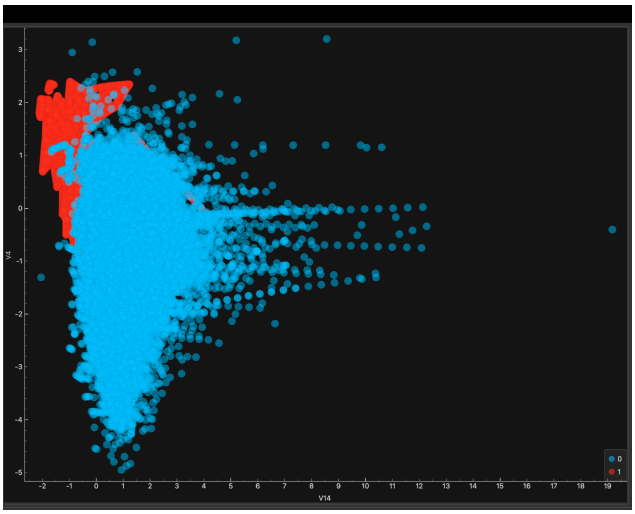
#		Gain ratio	Gini
1	V14	0.326	0.364
2	V4	0.282	0.320
3	V12	0.274	0.304
4	V10	0.265	0.301
5	V11	0.254	0.285
6	V3	0.231	0.262
7	V17	0.217	0.236
8	V16	0.208	0.232
9	V2	0.177	0.212
10	V7	0.177	0.205
11	V9	0.170	0.200
12	V18	0.127	0.149
13	V21	0.120	0.147
14	V27	0.117	0.141
15	V1	0.107	0.133
16	V6	0.106	0.130
17	V5	0.100	0.123
18	V28	0.086	0.110
19	V8	0.065	0.084
20	V19	0.054	0.070
21	V20	0.049	0.065
22	V23	0.016	0.022
23	V24	0.013	0.018
24	V26	0.007	0.010
25	V13	0.002	0.003
26	V15	0.002	0.002
27	V25	0.001	0.002
28	V22	0.001	0.001
29	Amount	0.000	0.000

Hallazgo Clave (Confirmación de Hipótesis)

- El análisis con *Rank* confirmó que la predicción del fraude depende principalmente de las **variables anonimizadas (Vx)**, destacándose V14, V4, V12, V10 y V11 como las más influyentes según los indicadores de Gain Ratio y Gini.

- En cambio, la variable **Amount**, como ya mencionamos, resultó ser **estadísticamente irrelevante**, con un valor de Gain Ratio cercano a *cero*. Estos resultados validan la hipótesis de que las variables internas del modelo son las que mejor explican los patrones anómalos asociados a fraudes, evidenciando que el monto de la transacción no contribuye significativamente a la detección.

Gráfico Dispersión V14 y V4 (Scatter Plot)



El gráfico compara las 2 features más relevantes V14 y V4 respectivamente, coloreando los puntos por Clase (Clase 0 azul y Clase 1 rojo), en base a este, pudimos sacar algunas conclusiones:

- Muestra una separación visual muy marcada entre la Clase 1 (rojo) y la Clase 0 (azul)
- Los puntos rojos (Fraude) se concentran fuertemente en una región específica:
    - Eje X (V14): Principalmente entre -2 y 1.
    - Eje Y (V4): Principalmente entre 1 y 3
- La gran mayoría de los puntos azules (No Fraude) se extienden por el resto del gráfico, mostrando la diversidad de transacciones legítimas.

Por lo tanto, este Scatter Plot confirma y valida la Hipótesis 2, que establece que las features Vx son la base de la predicción, en específico estas 2 features

4. Preprocesamiento y Selección de Variables:

a. Configuración del dataset y definición de variables

Utilizando el widget *Select Columns* definimos las variables.

- **Features (Características):** Seleccionamos las variables V1–V28 y *Amount* cómo *features* numéricas porque representan los distintos atributos de cada transacción. Estas variables serán las que el modelo utilice para aprender patrones y realizar predicciones.

- **Target (Variable objetivo):** Definimos *Class* como la variable *target* (objetivo) porque indica si una transacción es fraudulenta (1) o no fraudulenta (0). Al ser una variable binaria, es adecuada para un problema de clasificación supervisada, que busca predecir una de dos posibles categorías.

- **Meta (Metadatos):** Dejamos id como *meta attribute* (con select columns), ya que cumple solo la función de identificar cada transacción y no aporta información útil para el entrenamiento del modelo.

Esta configuración permite que los algoritmos de Orange comprendan qué variable deben predecir (*Class*) y qué atributos utilizar como entrada (*Features*), optimizando el proceso de detección de fraude.

b. Limpieza y Transformación de Datos

No se encontraron valores faltantes en el conjunto de datos, lo que eliminó la necesidad de imputación (y esto nos pareció lógico ya que al ser transacciones de tarjetas de crédito, sería muy raro que falten datos). A su vez nos encargamos de eliminar las variables tipo texto (ID) para que el modelo optimice su funcionamiento con variables únicamente numéricas.

c. **Estandarización de *Features***
El EDA mostró que las features V1–V28 y Amount tienen escalas muy diferentes, lo que podría desestabilizar los modelos. Por ello, se aplicó la Estandarización a todas las features numéricas mediante el widget *Preprocess* para asegurar que cada variable contribuyera equitativamente al proceso de aprendizaje.

d. **Dividir los datos**

Una vez configuradas y estandarizadas las variables, utilizamos el widget *Data Sampler* para dividir el dataset en dos subconjuntos: uno destinado al entrenamiento del modelo y otro para la evaluación.

- Se asignó un **70%** de los datos al entrenamiento y un **30%** al testeo, siguiendo la práctica habitual en los problemas de clasificación supervisada y así tratar de reducir el overfitting y underfitting.
- El conjunto de **entrenamiento** permite que el modelo aprenda los patrones y relaciones entre las variables que caracterizan las transacciones fraudulentas y las legítimas.
- Por su parte, el conjunto de **evaluación** se utiliza para medir la capacidad del modelo de generalizar lo aprendido y predecir correctamente sobre nuevos casos que no vio durante el entrenamiento.

5. Modelado:

El objetivo de esta etapa fue entrenar y evaluar múltiples algoritmos de clasificación supervisada para identificar patrones en los datos que permitieran distinguir una transacción fraudulenta (Clase 1) de una legítima (Clase 0).

Algoritmos de Clasificación Seleccionados
Se seleccionaron tres algoritmos de naturaleza diferente para evaluar qué tipo de enfoque se adapta mejor a la complejidad de los datos de detección de fraude:

1) Logistic Regression: Un modelo lineal simple que estima la probabilidad de que una instancia pertenezca a la clase objetivo. Es una base de referencia común por su velocidad y fácil interpretabilidad.
2) Tree: Un modelo no paramétrico que divide los datos basándose en reglas simples extraídas de las features. Es útil para identificar interacciones no lineales importantes.
3) Neural Network: Un modelo más complejo capaz de aprender relaciones intrincadas y no lineales en grandes volúmenes de datos, a menudo proporcionando un alto rendimiento en tareas de clasificación.

Configuración del Flujo de Modelado en Orange
El flujo de trabajo implementado en Orange garantiza que los modelos sean entrenados y evaluados sistemáticamente:

- Los tres modelos se conectaron al subconjunto de *Training* para aprender los patrones de fraude.
- Posteriormente, los modelos entrenados se conectaron al widget *Test and Score*, el cual utiliza el subconjunto de *Test* para calcular métricas de rendimiento y generar las Matriz de Confusión.

6. Evaluación del Modelo:

El objetivo de la evaluación es seleccionar el modelo más adecuado para el problema de negocio: Detección de Fraude. Esto implica priorizar la minimización de los errores más costosos.

Requisitos del Problema de Negocio

En la Detección de Fraude, priorizamos:

1. **Minimizar Falsos Negativos (FN)**: Evitar fallar en detectar un fraude real (Clase 1, pero se predice 0). El FN es el error más costoso, ya que representa una pérdida financiera directa. (Priorizar Recall).
2. Minimizar Falsos Positivos (FP): Evitar marcar una transacción legítima como fraude (Clase 0, pero se predice 1). Esto genera altos costos operativos y molesta a los clientes. (Priorizar Precisión). Pero sus consecuencias no terminan siendo tan graves como las de los FN
3. Maximizar F1-Score: Lograr el mejor equilibrio entre la Precisión y el Recall.

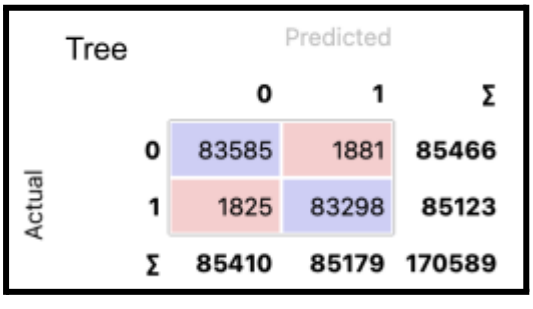
Predictions:

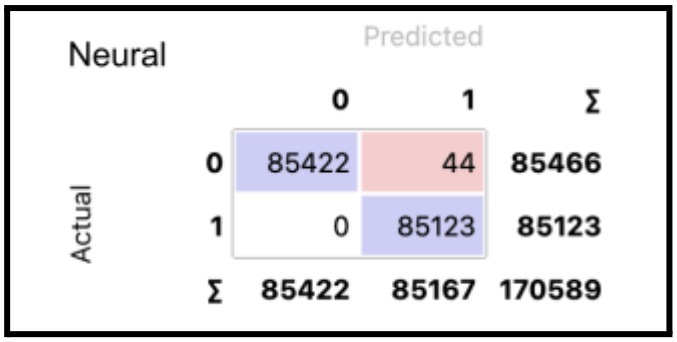
Model	AUC	CA	F1	Prec	Recall	MCC
Neural Network	1.000	1.000	1.000	1.000	1.000	1.000
Logistic Regression	0.993	0.964	0.964	0.965	0.964	0.929
Tree	0.981	0.978	0.978	0.978	0.978	0.957

Test and score:

Model	AUC	CA	F1	Prec	Recall	MCC
Neural Network	1.000	0.999	0.999	0.999	0.999	0.999
Logistic Regression	0.993	0.965	0.965	0.965	0.965	0.929
Tree	0.981	0.979	0.979	0.979	0.979	0.957

Matrices de Confusión





Neural Network: Este modelo exhibió un rendimiento prácticamente perfecto. Su métrica de Recall de 0.99 y, de forma más crítica, el conteo de Falsos Negativos (FN) de 0 en la Matriz de Confusión, indican que no dejó pasar ninguna transacción fraudulenta en el conjunto de prueba. Este resultado es ideal para la prevención de pérdidas.

Logistic Regression: Aunque el AUC fue alto (0.993), el F1 Score y el Recall (0.965) fueron los más bajos de los tres. En términos de fallos, la Matriz de Confusión muestra 4.215 Falsos Negativos, siendo el modelo menos efectivo para cumplir el requisito de no perder transacciones fraudulentas.

Árbol de Decisión (Tree): Este modelo mostró un rendimiento sólido con un F1 Score de 0.979. Sin embargo, a pesar de su alta precisión, registró 1.825 Falsos Negativos, lo que se traduciría en 1.825 fraudes perdidos en un escenario real, impactando negativamente en el objetivo de reducción de pérdidas.

Selección del Modelo Más Adecuado

El modelo seleccionado es Neural Network (NN)

1. **Rendimiento Superior:** Logró un rendimiento del 0.999 en F1-score, Precisión y Recall, superando consistentemente a sus competidores por un gran margen.
2. **Cumplimiento de Requisitos de Negocio:** Alcanzó 0 FN, satisfaciendo la prioridad crítica de no perder dinero por fraudes no detectados. Además, minimiza drásticamente las Falsas Alarmas (FP = 44) comparado con los otros modelos, lo que reduce los costos operativos y mejora la experiencia del cliente.
3. **Validación de la Hipótesis:** El éxito de la NN confirma que las relaciones complejas y no lineales entre las features V (como V14 y V4) fueron capturadas de manera efectiva, mientras que los modelos que son más simples (Tree y Regresión Logística) no pudieron trazar las fronteras de decisión con la misma precisión.

Si bien se notó una ligera sospecha de *overfitting* en la Red Neuronal (1.000 en Predictions vs 0.999 en Test), la abrumadora reducción de los errores críticos (FN y FP)

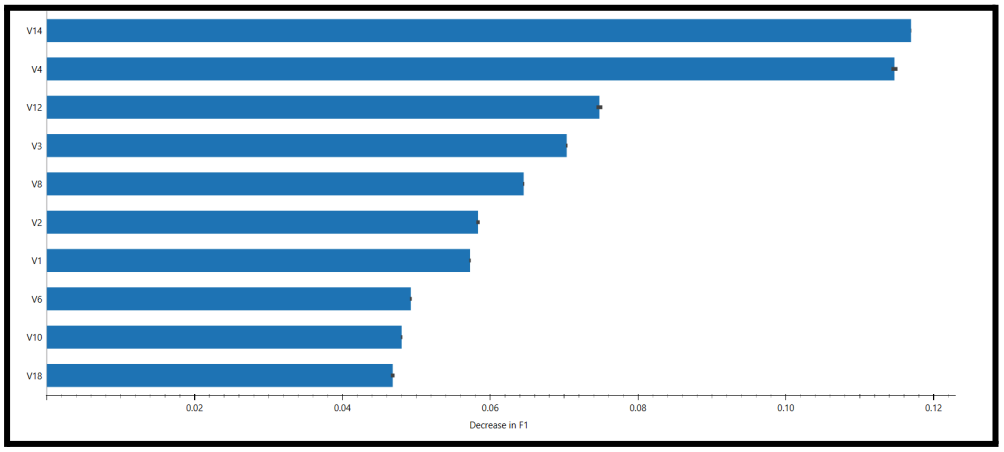
en el conjunto de prueba valida su elección como el modelo de mayor valor para el despliegue.

7. Interpretación de las Predicciones:

El objetivo de esta sección es analizar qué variables son consideradas más importantes por la **Red Neuronal** y cómo se relacionan con las hipótesis y hallazgos del EDA.

Para analizar el éxito del modelo se utilizó el widget *Feature Importance* con el F1-score como métrica de evaluación, ya que es la que mejor resume el equilibrio entre la detección de fraude (Recall) y la minimización de falsas alarmas (Precisión). Los resultados confirmaron el origen del alto rendimiento del modelo.

Variables Determinantes para el F1-Score en Red Neuronal



1. **Núcleo Predictivo V14 y V4:** La feature más importante fue V14, cuya eliminación provocaría una caída de 10.8 puntos porcentuales en el F1-Score. Le sigue V4, responsable de una caída de 8.8 puntos. Estas dos features, junto con V3 y V1, son el núcleo decisivo que la Red Neuronal explota para identificar el fraude.

2. **Validación de la Hipótesis:** Este ranking valida la hipótesis de que el fraude es distinguible por la estructura anómala capturada en las features transformadas Vx, y no por los montos. (En la foto no aparece, pero Amount está en la última posición indicando una variación de 0 puntos si se elimina)

3. **Justificación del FN = 0:** La NN logró su rendimiento perfecto (FN=0) porque estás features críticas (V14 y V4) son capaces de aislar el cluster de fraude (como se vio en el Scatter Plot) con una precisión que los modelos más simples no pudieron igualar. El modelo se apoya fuertemente en este subconjunto de variables para garantizar que en lo posible todas las transacciones anómalas sean identificadas.