

Contexto del Proyecto

Documentación oficial para la defensa del Sprint 3. Este archivo **README.md** es tu portada en Visual Studio Code.

Descargar README.md

Guía de Defensa (Los 10 Puntos)

- #1

Objetivo del Modelo
- #2

Descripción Dataset (X e y)
- #3

Preprocesamiento
- #4

División Train/Test
- #5

Selección Algoritmo
- #6

Entrenamiento .fit()
- #7

Predicciones .predict()
- #8

Métricas Evaluación
- #9

Modelo Implementado
- #10

Gráficos y Conclusiones

Tip para VSC: Descarga el README.md arriba. Contiene esta tabla mapeada a los archivos reales para que no te pierdas durante la presentación.

Variables del Modelo (X e y)

ROL	VARIABLE	SIGNIFICADO
y (Target)	is_fidelizado	Lo que queremos predecir (1=Fiel, 0=Ocasional)
X (Feature)	frequency	Cantidad de compras (Pista principal)
X (Feature)	recency_days	Días desde la última visita
X (Feature)	monetary_log	Total gastado (normalizado)



Resumen Ejecutivo

Generado por Gemini 2.5 Flash



Regenerar Reporte

Estimada Directiva,

Les presento un resumen ejecutivo de la situación actual de nuestra cartera de clientes, así como una estrategia táctica para optimizar su valor.

Estatus de Cartera

Nuestra cartera de clientes actual se compone de 67 consumidores. De estos, identificamos que 37 son clientes leales, lo que representa una base sólida y comprometida que supera la mitad de nuestros compradores activos. Los 30 clientes restantes son de carácter ocasional, un segmento con un potencial considerable para la conversión. El Ticket Promedio general de nuestra base de clientes se sitúa en \$ 39.573,39, un valor significativo que subraya la calidad y el impacto económico de cada transacción.

Hallazgo Clave

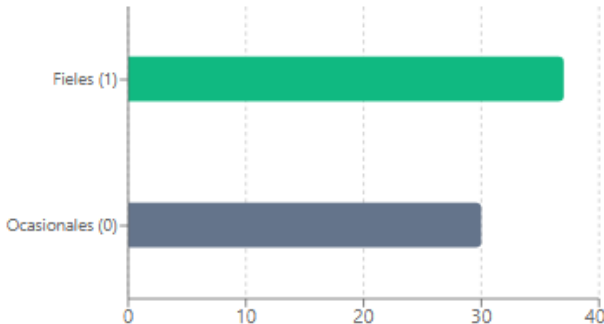
El elevado Ticket Promedio combinado con una base leal predominante nos posiciona favorablemente, pero la conversión de clientes ocasionales representa la vía más directa para un crecimiento estratégico y sostenido.

Estrategia Sugerida

Para capitalizar el valor actual de nuestros clientes y fomentar la recurrencia, proponemos implementar un programa de fidelización segmentado. Este deberá ofrecer beneficios exclusivos y personalización para nuestros clientes leales, buscando aumentar su frecuencia de compra y el valor de vida del cliente. Paralelamente, diseñaremos micro-campañas dirigidas a nuestros clientes ocasionales con incentivos atractivos para su segunda compra, transformándolos en compradores recurrentes y ampliando nuestra base fiel. La clave será una comunicación de valor que refuerce la relación y el beneficio de elegir Aurelion Retail.

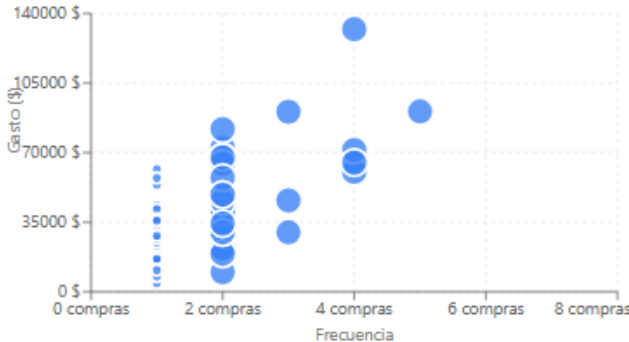
Distribución del Target

Proporción de clientes Fieles (1) vs Ocasionales (0)



Patrón: Frecuencia vs Gasto

Relación entre veces que compran y cuánto gastan



Dataset Procesado (RFM)

67 registros listos para el modelo



Descargar CSV

Ocultar Tabla

ID_CLIENTE	REGENCY_DAYS	FREQUENCY	MONETARY	CATEGORIA_PREFERIDA	IS_FIDELIZADO
1	94	2	\$ 72.448,00	ALIMENTOS	FIEL (1)
2	77	1	\$ 22.150,00	ALIMENTOS	OCASIONAL (0)
3	133	1	\$ 33.310,00	ALIMENTOS	OCASIONAL (0)
5	3	4	\$ 132.158,00	ALIMENTOS	FIEL (1)
6	136	2	\$ 48.878,00	ALIMENTOS	FIEL (1)
8	171	1	\$ 61.503,00	ALIMENTOS	OCASIONAL (0)
9	95	2	\$ 41.840,00	LIMPIEZA	FIEL (1)
10	9	1	\$ 25.860,00	LIMPIEZA	OCASIONAL (0)
12	60	2	\$ 49.518,00	ALIMENTOS	FIEL (1)

Simulador de Entrenamiento ML

Visualiza cómo el algoritmo aprende a separar clientes Fieles de Ocasionales.

✓ Entrenamiento Finalizado

📄 Descargar Informe Completo (.md)

ACCURACY GLOBAL

100%

¡Modelo Perfecto!

PRECISION (FIELES)

1.00

Sin errores en positivos

RECALL (FIELES)

1.00

Encontró a todos

TP

FP

FN

TN

Matriz de Confusión

⚠️ Guía Rápida: ¿Cómo leer la Matriz de Confusión?

TP Verdadero Positivo: La IA predijo "Fiel" y el cliente SÍ compró 2+ veces. ¡Éxito!

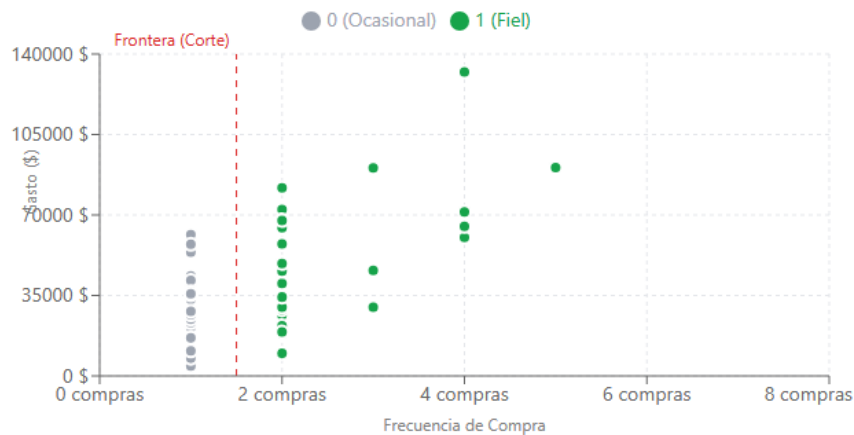
TN Verdadero Negativo: La IA predijo "Ocasional" y el cliente SÍ fue de 1 vez. ¡Ahorro!

FP Falso Positivo: Predijo "Fiel" pero era Ocasional. (Gastaste marketing por error).

FN Falso Negativo: Predijo "Ocasional" pero era Fiel. (Perdiste un cliente VIP).

📊 Frontera de Decisión Visual

Regla: Freq \geq 2



* Gráfico estilo Seaborn: Puntos con borde blanco, Grilla gris suave.

🎓 GLOSARIO DE DEFENSA

🔍 ¿Qué son las Iteraciones?

Es cuántas veces lee el modelo el libro entero (dataset). **100 Iteraciones** = lo leyó 100 veces. Si son muchas, memoriza (overfitting); si son pocas, no aprende.

📈 ¿Por qué la curva S?

Porque predecimos **Probabilidad** (0 a 1). La "S" (Sigmoide) aplasta valores locos (ej: 1.5) para que siempre den un % válido.

🔧 ¿Qué es el Optimizador?

Es el método para encontrar la solución. **liblinear** es ideal para datasets pequeños.

📋 CONCLUSIÓN EJECUTIVA

"Matemáticamente, la lealtad se decide en la **2da compra**. El monto gastado es irrelevante para la clasificación."

➡ Llévalo a Producción

Esta simulación fue visual. Descarga el código Python real (Scikit-Learn) para ejecutarlo en tu VS Code y presentarlo.

📄 Descargar entrenamiento_modelo_aurelion.py

Tu Código Python para el Sprint 3

Este código ahora incluye la generación de gráficos con Matplotlib y Seaborn para que los visualices directamente en tu Notebook de VS Code.

• script_completo_sprint3.py

📄 Copiar Código

📄 Descargar .ipynb

```
# =====
# PROYECTO AURELION - SPRINT 3: CLASIFICACIÓN DE FIDELIDAD (MACHINE LEARNING)
# VERSIÓN: 1.4 (Con Gráficos Integrados)
# =====

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
```

VERSION ANTERIOR

3. Ficha Técnica del Modelo SPRINT 3

ALGORITMO

Regresión Logística

Librería: Scikit-Learn

HIPERPARÁMETROS

Tasa Aprendizaje: 0.01

Iteraciones: 100

Optimizador: liblinear

VALIDACIÓN

El modelo utiliza **validación cruzada** (Train/Test Split) para asegurar que las predicciones sean generalizables y no memorizadas.

Nota para Estudiantes: ¿Por qué no usamos Regresión Lineal?

Es común confundir ambos términos. Aquí tienes la justificación para tu profesor:

Regresión Lineal

Predice **números infinitos** (ej. Precio: \$500, \$1000). No sirve para "Sí/No" porque podría dar resultados como 1.5 o -0.2.

Regresión Logística

Predice **probabilidades** entre 0 y 1 usando una curva "S". Es perfecta para clasificar categorías (Fiel vs Ocasional).

5. Guía Visual: Matriz de Confusión

TP (Verdadero Positivo):
La IA dijo "Fiel" y acertó.
Resultado: Ganancia de Marketing.

FP (Falso Positivo):
La IA dijo "Fiel" pero se equivocó.
Resultado: Desperdicio de dinero en promociones.

TN (Verdadero Negativo):
La IA dijo "Ocasional" y acertó.
Resultado: Ahorro de recursos.

FN (Falso Negativo):
La IA dijo "Ocasional" pero era Fiel.
Resultado: ¡Cliente perdido! Error grave.

Ingeniería de Features (RFM)

Análisis visual de las variables predictoras

[Pantalla Completa](#)

Resumen Ejecutivo AI

Generado por Gemini 2.5 Flash

¿Necesitas un análisis rápido para tu presentación?

[Haz clic para generar insights automáticos de tus datos RFM](#)

[Generar Reporte](#)

Distribución del Target

Proporción de clientes Fieles (1) vs Ocasionales (0)

Categoría	Proporción
Fieles (1)	~37
Ocasionales (0)	~30

Patrón: Frecuencia vs Gasto


Relación entre veces que compran y cuánto gastan

Frecuencia	Gasto (\$)
1	~5000
1	~10000
1	~15000
1	~20000
1	~25000
1	~30000
1	~35000
1	~40000
1	~45000
1	~50000
1	~55000
1	~60000
1	~65000
1	~70000
1	~75000
1	~80000
1	~85000
1	~90000
1	~95000
1	~100000
1	~105000
1	~110000
1	~115000
1	~120000
1	~125000
1	~130000
1	~135000
1	~140000
2	~5000
2	~10000
2	~15000
2	~20000
2	~25000
2	~30000
2	~35000
2	~40000
2	~45000
2	~50000
2	~55000
2	~60000
2	~65000
2	~70000
2	~75000
2	~80000
2	~85000
2	~90000
2	~95000
2	~100000
2	~105000
2	~110000
2	~115000
2	~120000
2	~125000
2	~130000
2	~135000
2	~140000
3	~5000
3	~10000
3	~15000
3	~20000
3	~25000
3	~30000
3	~35000
3	~40000
3	~45000
3	~50000
3	~55000
3	~60000
3	~65000
3	~70000
3	~75000
3	~80000
3	~85000
3	~90000
3	~95000
3	~100000
3	~105000
3	~110000
3	~115000
3	~120000
3	~125000
3	~130000
3	~135000
3	~140000
4	~5000
4	~10000
4	~15000
4	~20000
4	~25000
4	~30000
4	~35000
4	~40000
4	~45000
4	~50000
4	~55000
4	~60000
4	~65000
4	~70000
4	~75000
4	~80000
4	~85000
4	~90000
4	~95000
4	~100000
4	~105000
4	~110000
4	~115000
4	~120000
4	~125000
4	~130000
4	~135000
4	~140000
5	~5000
5	~10000
5	~15000
5	~20000
5	~25000
5	~30000
5	~35000
5	~40000
5	~45000
5	~50000
5	~55000
5	~60000
5	~65000
5	~70000
5	~75000
5	~80000
5	~85000
5	~90000
5	~95000
5	~100000
5	~105000
5	~110000
5	~115000
5	~120000
5	~125000
5	~130000
5	~135000
5	~140000
6	~5000
6	~10000
6	~15000
6	~20000
6	~25000
6	~30000
6	~35000
6	~40000
6	~45000
6	~50000
6	~55000
6	~60000
6	~65000
6	~70000
6	~75000
6	~80000
6	~85000
6	~90000
6	~95000
6	~100000
6	~105000
6	~110000
6	~115000
6	~120000
6	~125000
6	~130000
6	~135000
6	~140000
7	~5000
7	~10000
7	~15000
7	~20000
7	~25000
7	~30000
7	~35000
7	~40000
7	~45000
7	~50000
7	~55000
7	~60000
7	~65000
7	~70000
7	~75000
7	~80000
7	~85000
7	~90000
7	~95000
7	~100000
7	~105000
7	~110000
7	~115000
7	~120000
7	~125000
7	~130000
7	~135000
7	~140000
8	~5000
8	~10000
8	~15000
8	~20000
8	~25000
8	~30000
8	~35000
8	~40000
8	~45000
8	~50000
8	~55000
8	~60000
8	~65000
8	~70000
8	~75000
8	~80000
8	~85000
8	~90000
8	~95000
8	~100000
8	~105000
8	~110000
8	~115000
8	~120000
8	~125000
8	~130000
8	~135000
8	~140000

Dataset Procesado (RFM)

67 registros listos para el modelo

[Ver Datos Detallados](#)

**Dataset Procesado (RFM)**
67 registros listos para el modelo

Ocultar Tabla ^

[Descargar CSV Completo](#)

ID_CLIENTE	REGENCY_DAYS	FREQUENCY	MONETARY	CATEGORIA_PREFERIDA	IS_FIDELIZADO
1	94	2	\$72448.00	ALIMENTOS	FIEL (1)
2	77	1	\$22150.00	ALIMENTOS	OCASIONAL (0)
3	133	1	\$33310.00	ALIMENTOS	OCASIONAL (0)
5	3	4	\$132158.00	ALIMENTOS	FIEL (1)
6	136	2	\$48878.00	ALIMENTOS	FIEL (1)
8	171	1	\$61503.00	ALIMENTOS	OCASIONAL (0)
9	95	2	\$41840.00	LIMPIEZA	FIEL (1)
10	9	1	\$25860.00	LIMPIEZA	OCASIONAL (0)
12	60	2	\$49518.00	ALIMENTOS	FIEL (1)
13	59	1	\$13188.00	ALIMENTOS	OCASIONAL (0)

Simulador de Entrenamiento ML

Visualiza cómo el algoritmo aprende a separar clientes Fieles de Ocasionales.



ALGORITMO
Regresión Logística



TASA DE APRENDIZAJE
0.01



ITERACIONES
100



¿Listo para entrenar?

Al hacer clic, el modelo leerá tus 120 clientes y buscará la regla matemática perfecta usando la Tasa de Aprendizaje de 0.01.

 Ejecutar Entrenamiento

Simulador de Entrenamiento ML

Visualiza cómo el algoritmo aprende a separar clientes Fieles de Ocasionales.



Entrenamiento Finalizado



Descargar Informe Completo (.lmd)

ACCURACY GLOBAL

100%

¡Modelo Perfecto!

PRECISION (FIELES)

1.00

Sin errores en positivos

RECALL (FIELES)

1.00

Encontró a todos

MATRIZ CONFUSIÓN

TP

FP

FN

TN

(Pasa el mouse)



Guía Rápida: ¿Cómo leer la Matriz de Confusión?

TP

Verdadero Positivo: La IA predijo "Fiel" y el cliente SÍ compró 2+ veces. ¡Éxito!

TN

Verdadero Negativo: La IA predijo "Ocasional" y el cliente SÍ fue de 1 vez. ¡Ahorro!

FP

Falso Positivo: Predijo "Fiel" pero era Ocasional. (Gastaste marketing por error).

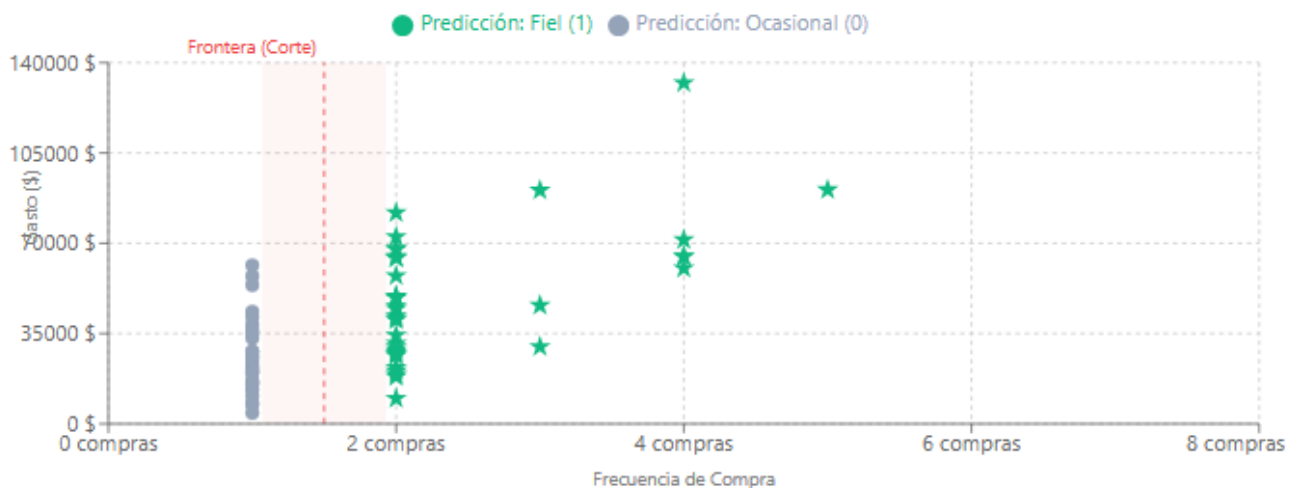
FN

Falso Negativo: Predijo "Ocasional" pero era Fiel. (Perdiste un cliente VIP).



Frontera de Decisión Visual

Regla: Freq \geq 2



* La zona sombreada roja indica la separación matemática calculada.



GLOSARIO DE DEFENSA



¿Qué son las Iteraciones?

Es cuántas veces lee el modelo el libro entero (dataset). **100 Iteraciones** = lo leyó 100 veces.

Si son muchas, memoriza (overfitting); si son pocas, no aprende.



¿Por qué la curva S?

Porque predecimos **Probabilidad** (0 a 1). La "S" (Sigmoide) aplasta valores locos (ej: 1.5) para que siempre den un % válido.

>_ ¿Qué es el Optimizador?

Es el método para encontrar la solución. **liblinear** es ideal para datasets pequeños.

📁 CONCLUSIÓN EJECUTIVA

"Matemáticamente, la lealtad se decide en la **2da compra**. El monto gastado es irrelevante para la clasificación."

>_ Llévalo a Producción

Esta simulación fue visual. Descarga el código Python real (Scikit-Learn) para ejecutarlo en tu VS Code y presentarlo.

📄 Descargar Script .py

Tu Código Python para el Sprint 3

Este es el código completo que cubre: Limpieza de Categorías (Coca-Cola/Alimentos), Cálculo RFM, y el Pipeline de Entrenamiento.

• script_completo_sprint3.py

📄 Copiar Código

📄 Descargar .ipynb

```
# =====
# PROYECTO AURELION - SPRINT 3: CLASIFICACIÓN DE FIDELIDAD (MACHINE LEARNING)
# VERSIÓN: 1.2 (Actualizada: Regla de Negocio Frecuencia >= 2)
# =====

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline

# 1. CARGA Y LIMPIEZA DE DATOS (Corrección de Categorías)
# -----
# Simulamos la carga. En tu caso: df = pd.read_csv('ventas_master.csv')
# Aquí definimos la lógica de "Fuente de Verdad" para corregir errores (ej. Coca Cola en Limpieza)
```


MATRIZ DE CONFUSION

Matriz de confusión

✓ Verdadero Positivo (TP): el modelo acierta al predecir un caso positivo.

✓ Verdadero Negativo (TN): el modelo acierta al predecir un caso negativo.

✗ Falso Positivo (FP): el modelo dice "positivo", pero era negativo.

✗ Falso Negativo (FN): el modelo dice "negativo", pero era positivo.

		Predicción	
		Positivos	Negativos
Observación	Positivos	Verdaderos Positivos (VP)	Falsos Negativos (FN)
	Negativos	Falsos Positivos (FP)	Verdaderos Negativos (VN)

Ejemplo clasificación binaria

5. Guía Visual: Matriz de Confusión

TP (Verdadero Positivo):
La IA dijo "Fiel" y acertó.
Resultado: Ganancia de Marketing.

TN (Verdadero Negativo):
La IA dijo "Ocasional" y acertó.
Resultado: Ahorro de recursos.

FP (Falso Positivo):
La IA dijo "Fiel" pero se equivocó.
Resultado: Desperdicio de dinero en promociones.

FN (Falso Negativo):
La IA dijo "Ocasional" pero era Fiel.
Resultado: ¡Cliente perdido! Error grave.

La matriz tiene 4 casillas. Las **Verdes** son aciertos y las **Rojas** son errores.

- **TP (Verdadero Positivo) = "El Acierto VIP"**
 - 🤖 **IA:** "Este cliente es FIEL".
 - 🛒 **Realidad:** El cliente compró 5 veces.
 - ✅ **Resultado:** ¡Perfecto! Identificamos a un buen cliente.
- **TN (Verdadero Negativo) = "El Acierto de Ahorro"**
 - 🤖 **IA:** "Este cliente es OCASIONAL".
 - 🛒 **Realidad:** El cliente compró 1 sola vez y se fue.
 - ✅ **Resultado:** ¡Bien! No gastamos dinero de marketing en él.
- **FP (Falso Positivo) = "La Falsa Alarma" (Error Tipo 1)**
 - 🤖 **IA:** "¡Es FIEL!".
 - 🛒 **Realidad:** Solo compró una vez.
 - ❌ **Problema:** Le regalaste un cupón de descuento y perdiste dinero porque no volvió.
- **FN (Falso Negativo) = "La Oportunidad Perdida" (Error Tipo 2)**
 - 🤖 **IA:** "Bah, es ocasional, ignóralo".
 - 🛒 **Realidad:** ¡Era un cliente que compraba muchísimo!
 - ❌ **Problema:** Lo trataste mal y se fue a la competencia. Es el error más grave en negocios.

¿Por qué tu gráfico dice 100%?

En tu simulador, verás que **TP** y **TN** están llenos, y **FP/FN** están vacíos (o en cero).

Esto pasa porque usamos una regla matemática exacta (Frecuencia ≥ 2) para crear la "Realidad" (Target). Como el modelo aprendió esa misma regla matemática, **no se equivoca nunca**. En la vida real, siempre habrá algunos errores en los cuadros rojos.

Explicación rápida para ti ahora:

La matriz compara **Lo que predijo la IA** (Columnas) vs **La Realidad** (Filas).

1. **Cuadrante Verde (TP - Verdadero Positivo):**
 - *Tu IA dijo:* "Este cliente es Fiel".
 - *La Realidad:* El cliente compró 5 veces.
 - *Resultado:* **¡Bingo!** Le mandas un cupón y lo aprovecha.
2. **Cuadrante Verde (TN - Verdadero Negativo):**
 - *Tu IA dijo:* "Este es Ocasional".
 - *La Realidad:* Compró una vez y nunca volvió.
 - *Resultado:* **Ahorro.** No gastaste dinero de marketing en alguien que no valía la pena.
3. **Cuadrante Rojo (FP - Falso Positivo) "El Desperdicio":**
 - *Tu IA dijo:* "¡Es Fiel!".

- *La Realidad:* Era Ocasional.
- *Resultado:* Le diste un descuento VIP y perdiste ese dinero porque no era VIP.

4. Cuadrante Rojo (FN - Falso Negativo) "El Pecado Capital":

- *Tu IA dijo:* "Es Ocasional, ignóralo".
- *La Realidad:* ¡Era un cliente Fiel!
- *Resultado:* Lo ignoraste y se fue a la competencia. Es el peor error.