



LET'S GO!/>

# Predicción de Fuga de Clientes en el Sector Financiero

PIM4 - Data Science



→ [soyhenry.com](https://soyhenry.com)



# Proyecto Integrador

## Contexto del proyecto y rol del estudiante:

FinanceGuard es un banco digital que ha experimentado un crecimiento exponencial en los últimos años, alcanzando 10000 clientes activos. Sin embargo, la dirección ha detectado un incremento preocupante en la tasa de abandono de clientes (churn), que actualmente alcanza el 20% anual, resultando en pérdidas millonarias.

Como Científico de Datos Junior, tu misión es desarrollar un modelo de Machine Learning completo que prediga qué clientes tienen mayor probabilidad de abandonar el banco. Este modelo permitirá al equipo de retención implementar estrategias personalizadas para retener a los clientes en riesgo, con el objetivo de reducir el churn al 15%.





# Objetivos del PI

- Implementar y comparar múltiples algoritmos de aprendizaje supervisado
- Optimizar modelos mediante validación cruzada y búsqueda de hiperparámetros
- Aplicar técnicas de ensamble para mejorar el rendimiento predictivo
- Manejar datos desbalanceados típicos en problemas de churn
- Implementar algoritmos básicos de aprendizaje no supervisado
- Entender e implementar los principios de aprendizaje profundo





<Proyecto Integrador>

# Entregable

# Final





# Proyecto Integrador

El entregable debe ser un **link a una carpeta de drive** con la siguiente identificación:

**ProyectoM4\_NombreEstudiante**

Recuerda habilitar los accesos a la carpeta



Contenido:

→ **Carpeta principal:** ProyectoM4\_NombreEstudiante/

## 1. Notebooks

- | — 1\_EDA\_RegresionLogistica.ipynb
- | — 2\_GradientBoosting\_Optimizacion.ipynb
- | — 3\_AprendizajeNoSupervisado.ipynb
- | — 4\_Extra\_credit.ipynb

## 2. Documentación

- | — Reporte\_Modelos.pdf: Comparación de modelos



**Se sugiere realizar los entregables en Google Colab, una herramientas gratuita en la nube para ejecutar código Python.**





<Proyecto Integrador>

# Detalle de **Avances**



# Proyecto Integrador



## Detalle de Avance#1:



### 1. Comprensión del problema y análisis exploratorio básico:

- Investigar qué es el churn bancario
- Carga y exploración inicial del dataset (50,000 clientes)
- Variables demográficas: edad, género, ubicación, antigüedad
- Variables financieras: saldo promedio, productos contratados, transacciones
- Variable objetivo: churn (1 = abandonó, 0 = activo)
- Análisis de desbalanceo de clases (típicamente 80-20)

### 1. Preparación de datos para regresión logística:

- Tratamiento de valores faltantes básico
- Encoding de variables categóricas (One-Hot, Label Encoding)
- Escalamiento de variables numéricas (StandardScaler)
- Split básico: train (80%), test (20%)
- Identificación de multicolinealidad

# Proyecto Integrador



## Detalle de Avance#1:

### 3. Implementación de Regresión Logística:

- **Regresión Logística Simple:**
  - Implementación desde cero (opcional)
  - Uso de scikit-learn
  - Interpretación de la función sigmoide
- **Análisis de coeficientes:**
  - Interpretación de pesos/coeficientes
  - Odds ratios y su significado
  - Intervalos de confianza
- **Evaluación específica:**
  - Matriz de confusión
  - Curva ROC y AUC
  - Precision, Recall, F1-Score

El estudiante debe entregar el notebook

**1\_EDA\_RegresionLogistica.ipynb** donde realice un análisis exploratorio (EDA) básico, un procesamiento de los datos básico, y los resultados de la implementación de la regresión logística



#### Conocimientos necesarios:

- Fundamentos de regresión logística
- Conceptos de odds y log-odds
- Métricas de clasificación binaria
- Interpretabilidad de modelos lineales



#### Tech Stack necesario:

- Python 3.8+
- Pandas, NumPy
- Scikit-learn (LogisticRegression)
- Matplotlib, Seaborn
- Jupyter Notebooks



**Notas extra:** La regresión logística será tu modelo baseline. Enfócate en entender cómo funciona el algoritmo, el procesamiento de las variables, y la interpretación de los resultados. Este modelo debe ser tu referencia para comparar modelos más complejos.

# Proyecto Integrador



## 1. Random Forest y Gradient Boosting:

- **Random Forest y XGBoost:**
  - Parámetros clave: learning\_rate, max\_depth, n\_estimators
  - Early stopping y otros parámetros para evitar overfitting
  - Feature importance y gain
  - Regularización alpha y lambda
- **LightGBM:**
  - Optimización para velocidad
  - Parámetros específicos: num\_leaves, min\_data\_in\_leaf
  - Categorical feature handling
- **CatBoost:**
  - Manejo automático de categóricas

## 2. Validación cruzada y métricas especializadas:

- **Estrategias de CV:**
  - StratifiedKFold para datos desbalanceados
  - GroupKFold para datos agrupados
  - StratifiedGroupKFold para datos desbalanceados y agrupados
  - TimeSeriesSplit para datos temporales
- **Métricas de evaluación:**
  - Accuracy, Recall, y Precision
  - PR-AUC para datos desbalanceados
  - ROC-AUC para datos desbalanceados
  - Business metrics personalizadas





# Proyecto Integrador

## Detalle de Avance#2:

### 3. Stacking y Blending:

- **Stacking:**
  - Nivel 1: XGBoost, LightGBM, CatBoost
  - Meta-learner: Regresión Logística regularizada
  - Cross-validation para evitar overfitting
- **Blending:**
  - Holdout set para meta-learner
  - Ponderación óptima de modelos

### 4. Optimización de hiperparámetros avanzada:

- **Bayesian Optimization con Optuna:**
  - Objective function personalizada
  - Visualización de importancia de hiperparámetros
- **Grid Search y Random Search:**
  - Comparación de eficiencia
  - Nested cross-validation



# Proyecto Integrador

## Detalle de Avance#2:

El estudiante debe entregar el notebook **2\_GradientBoosting\_Optimizacion.ipynb** donde compare los resultados de la ejecución de los algoritmos de Random Forest, Gradient Boosting (xGBoost, LightGBM, y CatBoost), y un ensamble de modelos (solo Stacking). El estudiante debe aplicar optimización de hiperparámetros solamente al modelo **XGBoost**, utilizando Grid Search. La optimización utilizando Optuna es opcional, siendo bueno mencionar que hoy en día es ampliamente utilizada. Es caso de decidir hacerla utilizar 50 trials.

### → Conocimientos necesarios:

- Algoritmos de Gradient Boosting
- Técnicas de ensemble (Stacking, Blending)
- Optimización bayesiana
- Validación cruzada
- Métricas de evaluación para modelos de clasificación

### → Tech Stack necesario:

- XGBoost, LightGBM, CatBoost
- Optuna, Random/Grid Search
- Scikit-learn (ensemble methods)



# Proyecto Integrador

## Detalle de Avance#3:



### 1. Clustering básico para segmentación:

- **K-Means clustering:**
  - Conceptos fundamentales: centroides, iteraciones
  - Selección del número de clusters K:
    - Método del codo (Elbow method)
    - Coeficiente de silueta (Silhouette score)
  - Implementación paso a paso
  - Interpretación de centroides
  - Visualización de clusters en 2D
- **DBSCAN (Density-Based clustering):**
  - Conceptos: core points, border points, noise
  - Parámetros básicos: eps (epsilon) y min\_samples
  - Ventajas: detecta outliers, no requiere definir K
  - Comparación con K-means

### 2. Reducción de dimensionalidad:

- **PCA (Principal Component Analysis):**
  - Conceptos básicos: componentes principales
  - Varianza explicada por cada componente
  - Selección del número de componentes
  - Visualización de datos en 2D y 3D
  - Interpretación de los componentes principales
- **t-SNE básico:**
  - Visualización no lineal de datos
  - Parámetro perplexity (concepto básico)
  - Diferencias con PCA
  - Limitaciones y cuidados en interpretación



# Proyecto Integrador

## Detalle de Avance#3:

### 3. Aplicación al problema de churn:

- **Segmentación de clientes:**
  - Aplicar K-means al dataset de clientes
  - Identificar 3-5 segmentos principales
  - Analizar características de cada segmento
  - Tasa de churn por segmento identificado
- **Perfiles de clientes por cluster:**
  - Características demográficas y comportamiento por cluster
  - Crear features derivadas del clustering

El estudiante debe entregar el notebook **3\_AprendizajeNoSupervisado.ipynb** donde implemente técnicas de clustering (K-Means y DBSCAN) para segmentar la base de clientes, y aplique reducción de dimensionalidad (PCA y t-SNE) para visualizar patrones en los datos.



# Proyecto Integrador

## Detalle de Avance#3:



### Conocimientos necesarios:

- Conceptos básicos de clustering (K-means, DBSCAN)
- Reducción de dimensionalidad (PCA, t-SNE básico)
- Métricas de evaluación no supervisada



### Tech Stack necesario:

- Scikit-learn (KMeans, DBSCAN, PCA)
- Matplotlib, Seaborn para visualizaciones
- Pandas, NumPy para manipulación de datos



**Notas extra:** La segmentación no supervisada puede revelar patrones “ocultos” en el comportamiento de los clientes. Usa estos insights para mejorar el feature engineering, y crear modelos más específicos.

# Proyecto Integrador

## Detalle de Avance#4:



### 1. Síntesis de resultados por avance:

- **Avance 1 - Regresión Logística:**
  - Performance del modelo baseline
  - Interpretabilidad y coeficientes más importantes
  - Fortalezas y limitaciones identificadas
- **Avance 2 - Gradient Boosting:**
  - Mejor modelo de boosting identificado
  - Feature importance del mejor modelo
  - Ganancia en performance vs modelo baseline
- **Avance 3 - Aprendizaje No Supervisado:**
  - Segmentos de clientes identificados
  - Insights de negocio por cluster
  - Features derivadas del clustering

### 1. Lecciones aprendidas:

- ¿Cuándo usar modelos supervisados vs no supervisados?
- Consideraciones para futuros proyectos de churn



# Proyecto Integrador

## Detalle de Avance#4:

En **Reporte\_Modelos.pdf**, el estudiante debe consolidar y comparar los resultados obtenidos en los avances anteriores, agregar las visualizaciones necesarias, insights de negocio y recomendaciones estratégicas



### Conocimientos necesarios:

- Síntesis y comparación de múltiples enfoques de ML
- Análisis de trade-offs en modelos de ML
- Integración de aprendizaje supervisado y no supervisado



### Tech Stack necesario:

- Todos los utilizados en avances anteriores
- Pandas para análisis comparativo



# Proyecto Integrador

## Detalle de Avance#4:

### EXTRA CREDIT

Como parte de las clases de aprendizaje supervisado, implementar los siguientes análisis en **4\_Extra\_credit.ipynb**, para el mejor modelo supervisado, es decir, el de mejor resultado según métrica de evaluación:

#### Optimización de threshold personalizada:

- Optimización del punto de corte según métricas de negocio
- Matriz de confusión con costos personalizados