# Memoria Técnica del Proyecto de Machine Learning: Clasificación de Ingresos

## 1. Introducción

Este proyecto tiene como objetivo predecir si una persona tiene un ingreso anual superior a 50.000 dólares a partir de variables demográficas y socioeconómicas, utilizando el Adult Income Dataset del censo estadounidense. Se abordan enfoques de aprendizaje supervisado y no supervisado, se implementa una red neuronal, y se desarrolla una aplicación de prueba utilizando Streamlit.

## 2. Análisis Exploratorio y Preprocesamiento

2.1 Carga y Limpieza de Datos  
- Lectura de datos sin encabezados y asignación de nombres de columna adecuados.  
- Limpieza de espacios en blanco, tratamiento de valores desconocidos y detección de inconsistencias.  
- Conversión de la variable objetivo income a formato binario (0: ≤50K, 1: >50K).

2.2 Ingeniería de Características  
- Creación de variables nuevas como extras (diferencia entre capital gain y capital loss).  
- Agrupación y ordenación semántica de variables categóricas (workclass, marital-status).  
- Aplicación de codificación ordinal y one-hot encoding según corresponda.  
- Estandarización de variables numéricas.

2.3 Balanceo de Clases  
Se identificó un desbalance en la variable objetivo. Para mitigar sus efectos se aplicó la técnica SMOTE (Synthetic Minority Oversampling Technique), que genera muestras sintéticas de la clase minoritaria para equilibrar la distribución de clases durante el entrenamiento.

## 3. Modelado

3.1 Aprendizaje Supervisado  
Se probaron y compararon múltiples modelos de clasificación:  
- Regresión Logística  
- Random Forest  
- K-Nearest Neighbors (KNN)  
- Support Vector Machines (SVM)  
- XGBoost  
- CatBoost

Los modelos XGBoost y CatBoost ofrecieron resultados particularmente sólidos debido a su capacidad para manejar relaciones no lineales y variables categóricas de forma eficiente. Se utilizó validación cruzada y búsqueda de hiperparámetros con GridSearchCV y RandomizedSearchCV.

3.2 Red Neuronal  
Se implementó una red neuronal mediante Keras con las siguientes características:  
- Varias capas densas con activación ReLU.  
- Regularización con Dropout.  
- Función de pérdida binary crossentropy y optimización con Adam.  
- Evaluación sobre conjunto de validación con métricas estándar.

## 4. Aprendizaje No Supervisado

- K-Means Clustering: identificación de patrones ocultos y grupos en los datos.  
- PCA (Análisis de Componentes Principales): reducción de dimensionalidad para visualización y análisis multivariado.

## 5. Evaluación de Modelos

Se utilizaron las siguientes métricas de evaluación:  
- Accuracy  
- Precision  
- Recall  
- F1-score  
- Matriz de confusión  
- Curva ROC y AUC

Los modelos con mejores resultados generales fueron XGBoost, CatBoost y la red neuronal, destacando especialmente en recall y AUC.

## 6. Despliegue: Aplicación en Streamlit

Se construyó una aplicación de prueba utilizando Streamlit, con el objetivo de:  
- Permitir ingreso de datos por parte del usuario.  
- Visualizar predicciones generadas por el modelo.  
- Explorar métricas y estadísticas del modelo en una interfaz sencilla e interactiva.

La aplicación puede ser ejecutada localmente o desplegada en la nube para demostraciones.

## 7. Conclusiones

- El uso combinado de técnicas supervisadas, no supervisadas y redes neuronales permitió una solución integral al problema.  
- SMOTE mejoró significativamente la capacidad predictiva en clases minoritarias.  
- XGBoost y CatBoost fueron altamente efectivos en comparación con modelos más clásicos.  
- La app en Streamlit facilita la comprensión y aplicación del modelo.

## 8. Recomendaciones

- Realizar seguimiento del modelo en producción para detectar posibles desbalances o deriva de datos.  
- Explorar ensambles de modelos para incrementar robustez.  
- Ampliar la demo para incluir carga de archivos o conexión con bases de datos en tiempo real.