Amazon Employee Access Challenge - Proyecto de Clasificación

# 1. Descripción del Proyecto

El Amazon Employee Access Challenge tiene como objetivo predecir si un empleado debe tener acceso a un recurso específico dentro de la empresa, utilizando varios atributos relevantes del empleado y el recurso. Este proyecto aborda un problema clásico de clasificación binaria, donde las clases están desbalanceadas (la mayoría de los empleados tienen acceso permitido, mientras que una minoría no lo tiene). El enfoque principal es probar múltiples modelos de clasificación para determinar cuál proporciona el mejor rendimiento, especialmente en la predicción precisa de accesos denegados, que son menos frecuentes.

# 2. Objetivos del Proyecto

1. Implementar varios modelos de Machine Learning para predecir si un empleado debe tener acceso a un recurso.  
2. Evaluar el rendimiento de cada modelo, centrándose en la precisión, recall y F1-score.  
3. Optimizar el modelo seleccionado mediante técnicas como RandomizedSearchCV y Bayesian Optimization.  
4. Utilizar técnicas de interpretación de modelos, como SHAP, para identificar las características más influyentes en las predicciones.

# 3. Descripción de los Modelos Evaluados

## 3.1. Random Forest con Clases Desbalanceadas

El modelo Random Forest fue entrenado con un ajuste en los pesos de las clases para abordar el desbalance presente en los datos. A continuación se presentan los resultados de su evaluación:

- Precisión:  
 - Clase 0 (Acceso Permitido): 1.00  
 - Clase 1 (Acceso Denegado): 0.96  
 - Recall:  
 - Clase 0: 1.00  
 - Clase 1: 0.76  
 - F1-Score:  
 - Clase 0: 1.00  
 - Clase 1: 0.85  
 - Exactitud Total: 1.00  
 - Macro Avg (F1-Score): 0.92

### 3.1.1. Análisis SHAP del Random Forest

Se generó un gráfico SHAP que muestra la importancia de las características en las predicciones del modelo. Las variables más influyentes fueron:  
 - V14 y V4, las cuales muestran una alta variabilidad en su impacto sobre las predicciones.  
 - Time: Tiene una influencia equilibrada en ambas clases.

## 3.2. Gradient Boosting con Optimización

Se entrenó un modelo Gradient Boosting, optimizado utilizando RandomizedSearchCV, con los siguientes hiperparámetros:

- Hiperparámetros:  
 - Subsample: 1.0  
 - Número de Estimadores: 200  
 - Min Samples Split: 10  
 - Max Depth: 5  
 - Learning Rate: 0.01

### 3.2.1. Resultados del Modelo Gradient Boosting

- Precisión:  
 - Clase 0: 1.00  
 - Clase 1: 0.94  
 - Recall:  
 - Clase 0: 1.00  
 - Clase 1: 0.74  
 - F1-Score:  
 - Clase 0: 1.00  
 - Clase 1: 0.83  
 - Macro Avg (F1-Score): 0.91

### 3.2.2. Análisis SHAP del Gradient Boosting

Las características clave en este modelo fueron V17, V4, y V14, mostrando un impacto significativo en las predicciones.

## 3.3. XGBoost con Optimización

Para este modelo, se implementó RandomizedSearchCV con los siguientes hiperparámetros optimizados:

- Hiperparámetros:  
 - Subsample: 0.8  
 - n\_estimators: 300  
 - max\_depth: 5  
 - learning\_rate: 0.1  
 - gamma: 0  
 - colsample\_bytree: 1.0

### 3.3.1. Resultados del Modelo XGBoost

- Precisión (Clase 0): 1.00  
 - Recall (Clase 1): 0.76  
 - Macro Avg (F1-Score): 0.92

### 3.3.2. Análisis SHAP del XGBoost

El análisis de SHAP mostró que las variables más influyentes fueron V14 y Time.

## 3.4. CatBoost con Bayesian Optimization (Modelo Final)

Se utilizó el modelo CatBoost con una optimización bayesiana de los hiperparámetros. Este fue el modelo que mostró el mejor rendimiento general en el proyecto.

- Mejores Hiperparámetros:  
 - Subsample: 1.0  
 - Scale\_pos\_weight: 3  
 - Learning\_rate: 0.1  
 - Iterations: 200  
 - Depth: 5  
 - Colsample\_bylevel: 0.8

### 3.4.1. Resultados del Modelo CatBoost

- Precisión (Clase 0): 1.00  
 - Recall (Clase 1): 0.80  
 - F1-Score (Clase 1): 0.85  
 - Macro Avg (F1-score): 0.93

### 3.4.2. Análisis SHAP del CatBoost

Se generaron gráficos SHAP que mostraron que las variables V4, V14 y Time fueron las más influyentes en las predicciones del modelo, siendo cruciales para la detección de accesos no autorizados.

# 4. Conclusiones Finales

Después de evaluar varios modelos, el modelo CatBoost con optimización bayesiana fue el que ofreció los mejores resultados, especialmente en la predicción de la clase minoritaria (accesos denegados).

## 4.1. Recomendaciones

1. Adoptar CatBoost como modelo final, ya que proporciona una alta precisión y recall en un problema de clasificación desbalanceado.  
2. Monitoreo de Variables Críticas: Implementar un sistema de monitoreo en tiempo real que destaque las transacciones donde las variables V14, V4 y Time se encuentren en rangos críticos. Este módulo de detección de anomalías podría ser crucial para mejorar la seguridad en los accesos a los recursos empresariales.