

# Informe de Análisis por Niveles

## Tipo de Proyecto: Machine Learning

### 1. Estadísticas del Repositorio

Total Commits: 301

Lenguajes: Jupyter Notebook (57.85%), HTML (41.9%), Python (0.25%), Dockerfile (0.0%)

### 2. Análisis por Niveles

#### Nivel Esencial

Compleitud: 75%

Requisitos Cumplidos:

- Un modelo de ML funcional que prediga el grado de satisfacción de los clientes
- Un análisis exploratorio de los datos.
- Una solución que productivice el modelo (Una aplicación de Streamlit)

Requisitos Faltantes:

- Overfitting inferior al 5%
- Un informe del rendimiento del modelo con sus métricas y la explicación de su performance, (feature importance, curva ROC, matrices de confusión)

#### Nivel Medio

Compleitud: 0%

Requisitos Faltantes:

- Un modelo de ML con técnicas de ensemble
- Uso de técnicas de Validación Cruzada.
- Optimización del modelo escogido con técnicas de ajuste de hiperparámetros

- Un sistema de recogida de feedback que monitorice la performance del modelo en producción.
- Un sistema de recogida de datos nuevos para futuros entrenamientos.

## Nivel Avanzado

Compleitud: 0%

Requisitos Faltantes:

- Una versión dockerizada del programa.
- Guardado en bases de datos de los datos recogidos por la aplicación
- Despliegue en Cloud de las soluciones aportadas.
- Incluir test unitarios.

## Nivel Experto

Compleitud: 0%

Requisitos Faltantes:

- Experimentos o despliegues con modelos de redes neuronales.
- Sistemas de entrenamiento y despliegue automático de nuevas versiones del modelo que se aseguren de que la nueva versión arroja mejores métricas y en función de eso sustituyan el viejo modelo por el nuevo o no. (A/B testing, Data Drifting, MLOps)

## 3. Análisis Técnico

Calidad Código: El código está bien escrito y documentado. Se utiliza la estructura de directorios y se separa el código en diferentes módulos.

Mejores Practicas: Se utiliza un enfoque modular para el código y se separa la interfaz de usuario de la lógica de predicción.

Experimentacion: Se implementan diferentes modelos de ML (XGBoost, logístico, stacking y neuronal) para predecir la satisfacción del cliente.

Manejo Datos: Los datos se preprocesan y normalizan antes de realizar las predicciones.

Optimizacion: No se observa ninguna optimización explícita del modelo, como ajuste de hiperparámetros o validación cruzada.

Etica Sesgos: No se identifican consideraciones éticas o prejuicios específicos, pero el conjunto de datos puede tener un impacto en el rendimiento del modelo.

#### **4. Recomendaciones**

- Añadir informe de rendimiento del modelo
- Optimizar el overfitting del modelo
- Implementar técnicas de validación cruzada
- Ajustar los hiperparámetros del modelo
- Añadir sistemas de recopilación de comentarios y datos
- Crear una versión dockerizada del programa
- Implementar pruebas unitarias
- Experimentar con modelos de redes neuronales

**Puntuación de Madurez: 65/100**