

Instituto Tecnológico de Estudios Superiores de Monterrey

Campus Estado de México

Inteligencia artificial avanzada para la ciencia de datos I (Gpo 101)

**Momento de Retroalimentación: Módulo 2 Análisis y Reporte sobre el desempeño del modelo.**

A01753729 | Marco Antonio Caudillo Morales

Septiembre 11, 2024

**Introducción**

En el contexto financiero moderno, la aprobación de préstamos es un proceso crucial que requiere la evaluación cuidadosa de múltiples factores para minimizar riesgos. Para optimizar este proceso, este proyecto propone un sistema de predicción de aprobación de préstamos utilizando técnicas de aprendizaje automático, con un enfoque en el uso del modelo de Gradient Boosting, conocido por su capacidad para combinar múltiples modelos débiles en un clasificador robusto y preciso.

En este documento se encontrará el análisis completo de porque se escogio este modelo junto con todos los análisis de las gráficas.

Como se comento en la introducción, loq ue se busca aumentar la aprobación de préstamos a las personas que cumplan con el perfil y para esto se debe hacer una evaluación cuidadosa.

El dataset que se ocupo es:

<https://www.kaggle.com/datasets/architsharma01/loan-approval-prediction-dataset>

El conjunto de datos de aprobación de préstamos es una colección de registros financieros e información asociada que se utiliza para determinar la elegibilidad de personas u organizaciones para obtener préstamos de una institución crediticia. Incluye varios factores como la puntuación CIBIL, los ingresos, la situación laboral, el plazo del préstamo, el monto del préstamo, el valor de los activos y el estado del préstamo.

A screenshot of a computer

Description automatically generated

A screenshot of a graph

Description automatically generated

Como primer paso se hizo la revisión de la base de datos y se observaron las columnas que contenian y se vio si es que no existián valores nulos.

A screenshot of a computer code

Description automatically generatedA screen shot of a computer code

Description automatically generated

Una vez observado y al no tener valores nulos lo que se busco fue que “loan\_status” que es nuestra variable objetivo fuera binario por lo que se realizó el cambio.

{*' Approved'*: 1, *' Rejected'*: 0}

A screenshot of a black and white screen

Description automatically generated

Una vez con los cambios previos lo que se busco realizar matriz de correlación para analizar nuestras variables que eran altamente dependientes con nuestra variable objetivo.

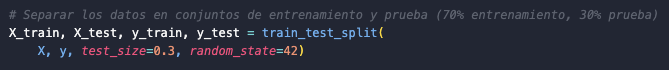
A screenshot of a graph

Description automatically generated

Las variables dependientes que se escogieron fue las de:

*'cibil\_score'*, *'loan\_term'*, *'no\_of\_dependents'*

Para este punto se ocupo la librería de “sklearn” para poder dividir el dataset en train y test (70% entrenamiento y 30% de prueba)



A screenshot of a computer

Description automatically generatedCon esto ya pudimos ocupar diferentes modelos para poder ver cuál es el que mejor resultados nos daban.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

El modelo que se ha escogido tras el análisis es **Gradient Boosting** por las siguientes razones:

**1. Exactitud:**

* **Gradient Boosting** tiene la mayor exactitud con un 0.94, lo que indica que predice correctamente el 94% de las veces.
* Otros modelos como el **Decision Tree** (0.92), **Random Forest** (0.93), y **Logistic Regression** (0.90) tienen una exactitud ligeramente inferior.

**2. Matriz de Confusión:**

* La matriz de confusión del modelo de **Gradient Boosting** muestra:
  + Falsos negativos y falsos positivos más bajos en comparación con otros modelos.
  + En la clase "0" (Rejected), tiene 2003 verdaderos negativos (correctamente rechazados) y solo 228 falsos positivos (mal clasificados como aprobados).
  + En la clase "1" (Approved), tiene 3611 verdaderos positivos (correctamente aprobados) y solo 158 falsos negativos (mal clasificados como rechazados).
* Comparando con otros modelos, **Gradient Boosting** tiene menos errores (falsos positivos y falsos negativos).

**3. Reporte de Clasificación:**

* **Gradient Boosting** tiene los mejores valores de:
  + **Recall** (Sensibilidad): 0.96 para la clase "1" (Approved), lo que indica que el modelo es capaz de identificar correctamente el 96% de las solicitudes aprobadas, lo cual es importante cuando se quiere minimizar el riesgo de no aprobar una buena solicitud.
  + **F1-Score**: Con 0.95 para la clase "1", muestra que tiene un buen balance entre precisión y recall.
  + Otros modelos como **Decision Tree** y **Random Forest** también tienen buenos F1-scores, pero son inferiores en comparación a Gradient Boosting (0.94 frente a 0.95 en la clase "1").

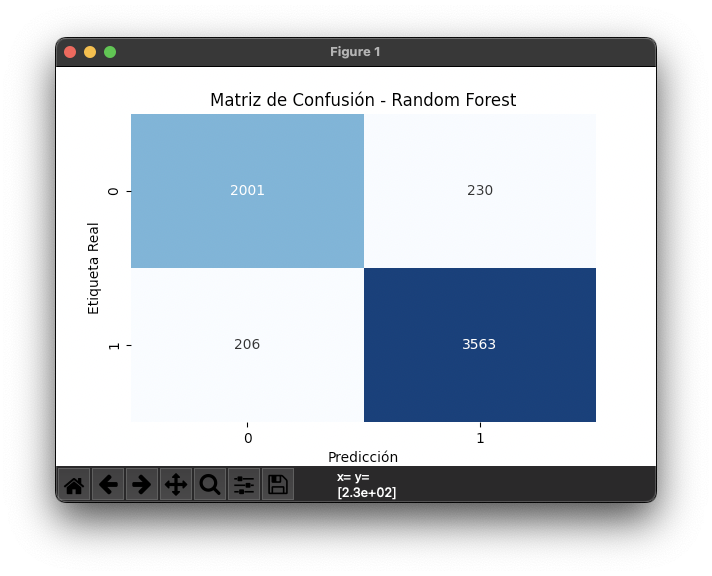
**4. Precisión y Recall en Clase Rechazada (0):**

* **Gradient Boosting** tiene una precisión de 0.93 para la clase "0" (Rejected), lo que significa que es preciso en predecir cuando una solicitud debe ser rechazada.
* Otros modelos como el **Decision Tree** y **Random Forest** también tienen buenos resultados en esta métrica, pero su recall (la capacidad de identificar correctamente todos los rechazos) es ligeramente inferior a Gradient Boosting.

**Conclusión:**

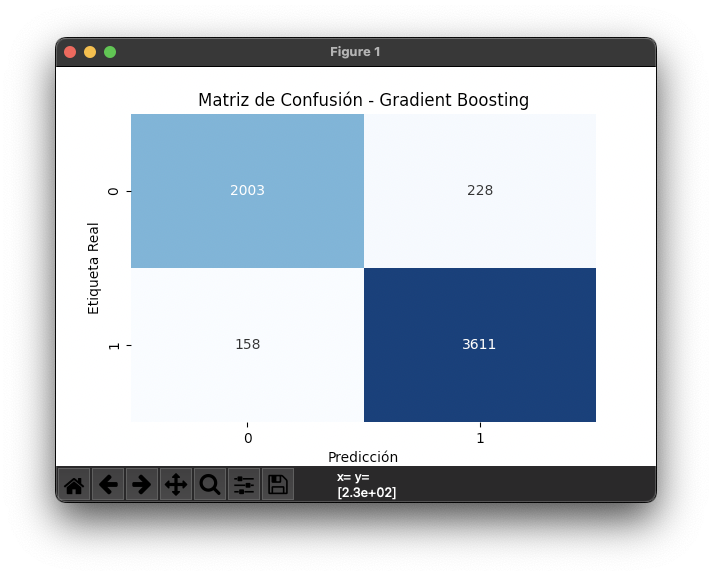
**Gradient Boosting** se destaca porque no solo tiene la mejor exactitud general (0.94), sino que también muestra una excelente capacidad de balancear tanto la precisión como el recall en ambas clases (aprobada y rechazada), reduciendo significativamente tanto los falsos positivos como los falsos negativos. Esto es particularmente importante en contextos donde los errores en la clasificación de aprobaciones y rechazos de préstamos pueden tener consecuencias costosas.

Por lo tanto, **Gradient Boosting** es el mejor modelo en este caso debido a su capacidad de generar predicciones más equilibradas y su menor tasa de errores en comparación con los otros modelos.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated



A screen shot of a computer

Description automatically generatedA screen shot of a graph

Description automatically generatedPara hacer el análisi de Overfitting se ocuparon las gráficas de Curvas de Aprendizaje

A screen shot of a graph

Description automatically generatedA screen shot of a graph

Description automatically generated

**Árbol de Decisión**

**Interpretación:**

**Curva de Entrenamiento:**

La línea azul muestra que el modelo de árbol de decisión tiene una precisión muy alta (~100%) en el conjunto de entrenamiento, lo que indica que está ajustando perfectamente los datos de entrenamiento.

**Curva de Prueba:**

La línea naranja muestra que la precisión en el conjunto de prueba es considerablemente menor (~92%).

**Conclusión:**

Esto indica un claro sobreajuste (overfitting). El modelo se adapta demasiado a los datos de entrenamiento y no generaliza bien en nuevos datos. Para mejorar esto, sería recomendable podar el árbol o ajustar algunos hiperparámetros como la profundidad máxima del árbol.

**Gradient Boosting**

**Interpretación:**

**Curva de Entrenamiento:**

La precisión en el conjunto de entrenamiento es alta (~98%), pero no alcanza el 100%, lo que indica que el modelo no está completamente sobreajustado.

**Curva de Prueba:**

La precisión en el conjunto de prueba es similar a la del entrenamiento (~92-93%), lo que indica una buena capacidad de generalización.

**Conclusión:**

El modelo de Gradient Boosting parece estar funcionando bien, logrando un buen equilibrio entre ajuste y generalización. Es menos probable que esté sobreajustado en comparación con el modelo de árbol de decisión.

**Regresión Logística**

**Interpretación:**

**Curva de Entrenamiento:**

La precisión en el conjunto de entrenamiento mejora gradualmente, pero nunca alcanza valores muy altos, quedándose alrededor del 90.25%.

**Curva de Prueba:**

La precisión en el conjunto de prueba es también alrededor del 90.75%.

**Conclusión:**

El modelo de Regresión Logística muestra un rendimiento bastante consistente entre los conjuntos de entrenamiento y prueba, lo que indica que el modelo está generalizando bien y no está sobreajustado. Es un buen modelo de base, aunque podría no captar las relaciones más complejas en los datos.

**Random Forest**

**Interpretación:**

**Curva de Entrenamiento:**

La precisión del conjunto de entrenamiento es muy alta (~99-100%).

**Curva de Prueba:**

La precisión en el conjunto de prueba es significativamente más baja (~92%), pero se mantiene estable.

**Conclusión:**

Al igual que el árbol de decisión, Random Forest también muestra un cierto grado de sobreajuste, aunque no tan extremo. Random Forest tiende a mejorar la capacidad de generalización en comparación con los árboles de decisión individuales, pero todavía hay margen de mejora en la diferencia entre las curvas de entrenamiento y prueba.

Conclusión General

**Regresión Logística:** Buen rendimiento con poca o ninguna señal de sobreajuste. Funciona bien en problemas lineales.

**Árbol de Decisión:** El modelo está claramente sobreajustado y necesita una mayor regularización o poda.

**Random Forest:** Mejora en la capacidad de generalización respecto a los árboles de decisión, pero aún muestra signos de sobreajuste.

**Gradient Boosting:** Presenta el mejor equilibrio entre ajuste y generalización, siendo el modelo más efectivo en términos de rendimiento.

**Recomendaciones**

**Regresión Logística** es una opción segura y simple cuando las relaciones entre las variables son lineales.

**Gradient Boosting** es probablemente la mejor opción cuando se desea un modelo que generalice bien y capture relaciones no lineales entre las variables.

Si se usa **Árbol de Decisión** o **Random Forest**, es recomendable ajustar sus hiperparámetros para reducir el sobreajuste, como la profundidad máxima o el número mínimo de muestras en las hojas.

A screen shot of a computer code

Description automatically generated

Dentro del código vienen datos de prueba para validar los modelos, estos se encuentran en las imagenes de evaluación.

A screenshot of a graph

Description automatically generated

**Interpretación:**

1. **Baja Varianza**: Una varianza de solo **0.0076** indica que el modelo tiene un buen equilibrio entre aprender de los datos de entrenamiento y generalizar bien en los datos de prueba. El modelo no ha memorizado los datos de entrenamiento ni se ha sobreajustado, lo que es un resultado excelente para un problema de clasificación como este.
2. **Exactitud Alta en Ambos Conjuntos**: Tanto la exactitud en entrenamiento (94.33%) como la exactitud en prueba (93.57%) son muy altas, lo que sugiere que el modelo está aprendiendo patrones relevantes de los datos sin caer en un sobreajuste. Esto es ideal en situaciones donde la exactitud es crítica, como en la predicción de aprobaciones de préstamos.

El modelo de **Gradient Boosting** ha demostrado ser robusto, con un bajo nivel de varianza, lo que indica que no está sobreajustado a los datos de entrenamiento y generaliza bien en los datos de prueba. Este resultado sugiere que los ajustes realizados en los hiperparámetros del modelo han sido efectivos, logrando un equilibrio óptimo entre el ajuste del modelo y su capacidad de generalización.

Para este proyecto se opto por una interfaz gráfica para poder interactuar con el modelo.

A screenshot of a video game

Description automatically generatedA screenshot of a computer

Description automatically generated

Las instrucciones para instalar y correr el proyecto se encuentra en el archivo README del repositorio.

Se intentarón realizar estos cambios, pero el tiempo de ejecución es demasiado alto.

**Ajuste de Hiperparámetros**:

* **Número de estimadores (n\_estimators)**: Controlar cuántos árboles de decisión se entrenan en el modelo de boosting. Incrementar este valor puede ayudar a mejorar el rendimiento hasta cierto punto, aunque si es demasiado alto, puede causar sobreajuste.
* **Profundidad máxima de los árboles (max\_depth)**: Controla la profundidad máxima que puede alcanzar cada árbol en el modelo de boosting. Si es muy profundo, el modelo puede sobreajustarse a los datos de entrenamiento. Reducir este valor ayuda a que los árboles sean más simples y generalicen mejor.
* **Tasa de aprendizaje (learning\_rate)**: Esta es la contribución de cada árbol al modelo final. Si la tasa de aprendizaje es demasiado alta, el modelo puede aprender demasiado rápido y sobreajustarse. Una tasa de aprendizaje más baja permite un ajuste más gradual.
* **Fracción mínima de datos en las hojas (min\_samples\_leaf)**: Esta métrica controla cuántos datos debe haber en una hoja de decisión. Si es demasiado bajo, el modelo puede ser muy sensible al ruido. Ajustarlo hacia arriba puede mejorar la robustez.

**Control de la tasa de aprendizaje (learning\_rate)**:

* Reducir la tasa de aprendizaje obliga al modelo a aprender de forma más lenta y gradual, lo que ayuda a evitar el sobreajuste. Una tasa de aprendizaje más baja, junto con un mayor número de árboles, permite que el modelo construya de manera incremental una predicción más precisa y robusta.

**Subsampleo (subsample)**:

* Usar solo una fracción de los datos en cada iteración introduce una especie de aleatoriedad que reduce la varianza y previene el sobreajuste. Este enfoque es similar al que utiliza el **Random Forest**, pero en el contexto del **Gradient Boosting**, ayuda a que el modelo sea más robusto al ruido de los datos.

**Regularización**:

* La regularización L1 y L2 ayuda a reducir la complejidad del modelo, evitando que los coeficientes se vuelvan demasiado grandes, lo que a su vez evita el sobreajuste y mejora la generalización del modelo.

Después de aplicar estas técnicas de ajuste de hiperparámetros y regularización, deberías observar que el modelo de **Gradient Boosting** mejora en términos de:

* **Exactitud** en los datos de prueba.
* **Reducción del sobreajuste**, es decir, una menor discrepancia entre el rendimiento en los datos de entrenamiento y los datos de prueba.
* **Mejora de las métricas de precisión, recall y F1-score**, particularmente en las clases minoritarias (como la clase "rechazada").

**Conclusión**

En este proyecto, implementamos y evaluamos múltiples modelos de clasificación para predecir la aprobación o rechazo de préstamos, utilizando un conjunto de datos relacionados con el historial crediticio de los solicitantes. Los modelos que se implementaron incluyen **Regresión Logística**, **Árboles de Decisión**, **Bosques Aleatorios** y **Gradient Boosting**. Después de analizar el rendimiento de cada modelo a través de métricas como la exactitud, la matriz de confusión, y el reporte de clasificación (precisión, recall, F1-score), identificamos que el modelo de **Gradient Boosting** tuvo el mejor desempeño general.