Tercera entrega

Integrantes

Diego Alejandro Poveda Alzate
Pedro Pablo Saldarriaga Jaramillo
Jhonier Andrés Córdoba Asprilla

Tutor

Raul Ramos Pollán

Modelos y Simulación de Sistemas 1 / Introducción a la Inteligencia Artificial

Universidad de Antioquia
Medellín, Antioquia
2023

1. Introducción

Las tarjetas de crédito son uno de los medios de pago más utilizados en todo el mundo. Con el aumento del comercio electrónico, las transacciones con estas tarjetas en línea también han aumentado significativamente. Esto, lamentablemente, también ha llevado a un aumento en las actividades fraudulentas relacionadas con las tarjetas. Los bancos y las empresas emisoras de tarjetas están constantemente buscando formas de detectar y prevenir transacciones fraudulentas para minimizar las pérdidas financieras y mantener la confianza de sus clientes.

2. Exploración descriptiva del dataset

El dataset a utilizar proviene de una competencia de Kaggle, compuesto por un conjunto de archivos **.csv** (comma separated values), los cuales nos dan la información requerida.

Este dataset fue generado mediante un modelo de deep learning que previamente fue entrenada en el problema de Detección de Fraudes con Tarjetas de Crédito (https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud)

El enlace a la competencia es https://www.kaggle.com/competitions/playground-series-s3e4

Columnas del dataset

- **Id**: un identificador único para cada transacción, utilizado para indexar los datos y no tiene poder predictivo.
- **Tiempo**: el tiempo transcurrido en segundos entre esta transacción y la primera transacción en el dataset. No es la marca de tiempo real de la transacción sino más bien una medida de tiempo relativa.
- **V1 a V28:** son el resultado de una transformación (específicamente Análisis de Componentes Principales, o PCA) aplicada a los datos originales sin procesar. PCA es una técnica de reducción de dimensionalidad que puede transformar un conjunto de variables correlacionadas en un conjunto de variables no correlacionadas llamadas componentes principales. La idea principal detrás del PCA es capturar la mayor cantidad de variabilidad en los datos con la menor cantidad posible de componentes principales. La naturaleza exacta y el significado de estas variables no se proporcionan por razones de confidencialidad. Sin embargo, son características numéricas derivadas de los datos originales sin procesar sobre la transacción.
- **Amount (Monto):** El monto de compra de la transacción. Esta es una característica crucial, ya que ciertas cantidades pueden ser indicativas de actividades fraudulentas, según el contexto.
- **Class (Clase):** Variable respuesta que tiene el valor de 1 en caso de que la transacción sea fraudulenta, y 0 en otro caso.

La competencia recomienda el cálculo de la métrica AUC-ROC, aunque también decidimos hacer el análisis de otras métricas como, por ejemplo:

- **Accuracy**: Calcula la proporción de predicciones de fraude correctas. Si solo el 0.1% de las transacciones son fraudulentas, un modelo que predice que todas las transacciones son legítimas aún tendría una precisión del 99.9%.

- **Precision:** Indica qué proporción de las transacciones que el modelo predice como fraudulentas lo son realmente. Dada la gravedad del fraude, es importante tener una precisión alta para evitar falsas alarmas.

- **Recall:** Mide qué proporción de las transacciones fraudulentas reales fueron correctamente detectadas por el modelo. En el contexto de detección de fraude, un recall alto es crucial porque no queremos pasar por alto eventos fraudulentos

- **F1-Score:** Dado que tanto la precisión como el recall son importantes en la detección de fraudes, el F1-Score ayuda a encontrar un equilibrio entre ambos.

3. Iteraciones de desarrollo

3.1 Preprocesado de datos

Al hacer la revisión del dataset que nos brinda Kaggle para esta competencia, y de acuerdos a las indicaciones dadas por el profesor a la hora de proponer el uso del mismo para el proyecto, se evidencia que no se tiene el número de columnas categóricas necesarias, y tampoco se tiene el porcentaje mínimo de datos faltantes requeridos para el proyecto.

Por esto mismo, se decidió simular los requisitos faltantes, es decir, agregar al menos 3 columnas categóricas y simular la falta de información.

Las columnas categóricas que se agregaron mediante la simulación son:

- Rango de monto pagado
- Medio en el cual se realizó la transacción
- Tipo de tarjeta usada para la transacción

Además de esto, en estas columnas simuladas se eliminaron un porcentaje de datos, para así cumplir con el requisito de valores faltantes. Se decidió hacerlo en estas columnas ya que, en un caso de la vida real, es posible que alguno de estos valores no se conozca.

3.2 Modelos predictivos

Decidimos usar los algoritmos de Regresión Logística y Árboles de Decisión.

- Regresión Logística:
  Es un algoritmo clásico y ampliamente utilizado para problemas de clasificación binaria, cómo es el caso de la detección de fraude (fraudulento vs no fraudulento). Como un modelo lineal, es eficiente y suficientemente robusto en muchos escenarios, especialmente cuando las relaciones entre las variables son aproximadamente lineales.

- Árboles de Decisión:
  A diferencia de la regresión logística, los árboles de decisión pueden capturar relaciones no lineales entre las características y la etiqueta. Además, proporcionan información útil sobre la importancia relativa de las características utilizadas en la predicción.

3.3 Modelos no supervisados

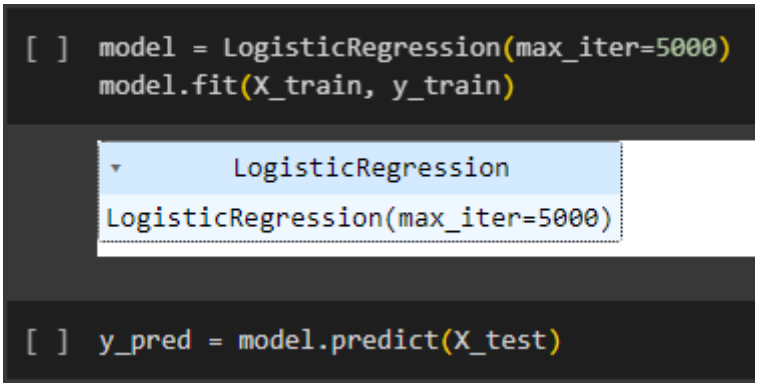
En cuanto a modelos no supervisados + modelos supervisados, escogimos las combinaciones de PCA + Regresión Logística y K-Means + Árboles de Decisión

- PCA (Análisis de Componentes Principales) + Regresión Logística:
  PCA es útil para reducir la dimensionalidad del dataset, manteniendo la mayor cantidad de información posible. Esto es especialmente relevante en datasets con muchas variables, como es el caso de las características V1 a V28 en el dataset. Al reducir la dimensionalidad, PCA puede mejorar la eficiencia del modelo de regresión logística y, en algunos casos, incluso su rendimiento al eliminar el ruido o la colinealidad entre las características.

- K-Means + Árboles de Decisión:
  K-Means es un algoritmo de clustering que puede ser usado para crear nuevas características que podrían ayudar en la tarea predictiva. Por ejemplo, asignar a cada observación la etiqueta de su clúster más cercano como una nueva característica. La adición de características basadas en clústeres puede mejorar la separabilidad de las clases, lo cual puede ser útil en un problema de clasificación como la detección de fraudes.

3.4 Resultados, métricas y curvas de aprendizaje

Para el cálculo de Accuracy, Precision, Recall y F1-Score entrenamos un modelo de Regresión Logística, con un máximo de iteraciones de 5000 (debido a que las 1000 por defecto se quedaban cortas para el tamaño del dataset).



A partir de este modelo, empezamos a calcular las métricas mencionadas.

- Accuracy

```
[ ]  accuracy = accuracy_score(y_test, y_pred)

[ ]  accuracy

     0.9975128918906585
```

- Precision

```
[ ]  precision = precision_score(y_test, y_pred)
     precision

     0.15384615384615385
```

- Recall

```
[ ]  recall = recall_score(y_test, y_pred)
     recall

     0.04395604395604396
```

- F1-Score

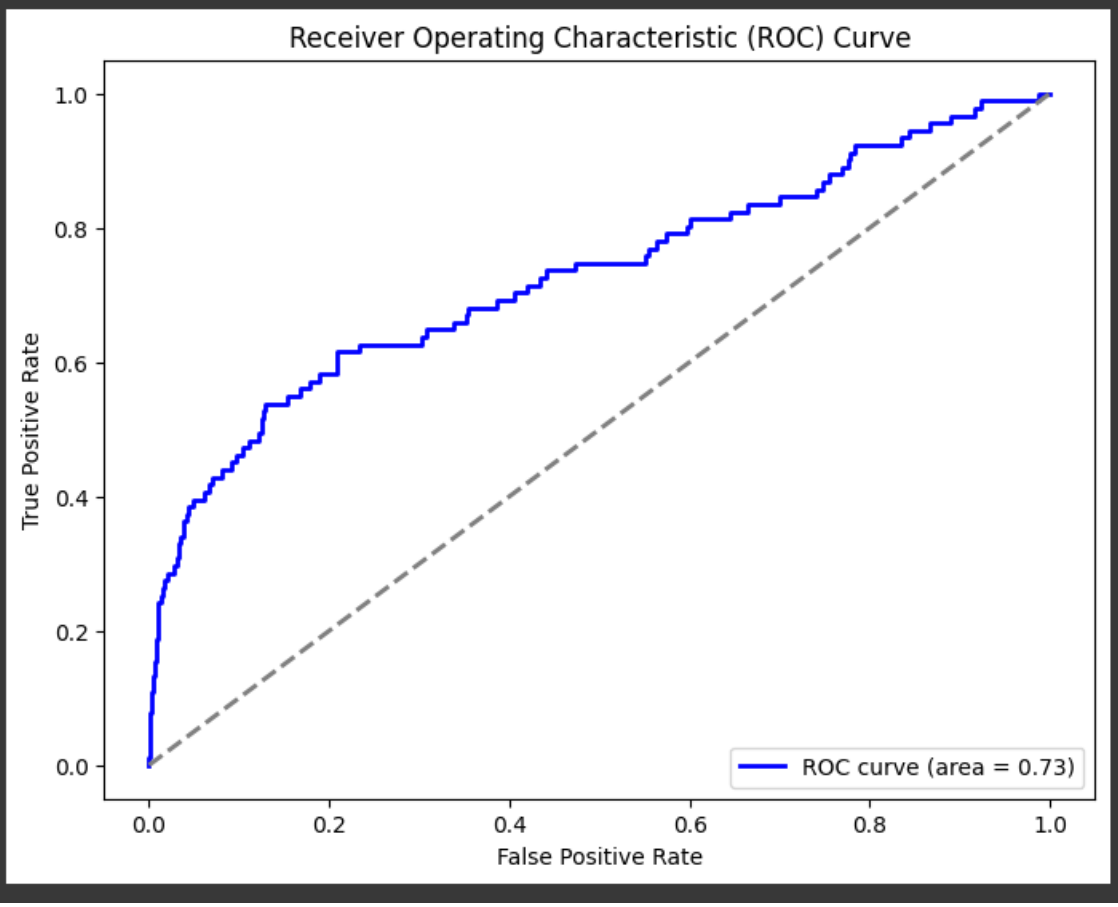
```
[ ]  f1 = f1_score(y_test, y_pred)
     f1

     0.06837606837606838
```

También calculamos una versión del ROC-AUC

```
[ ]  y_probs = model.predict_proba(X_test)[:, 1]  # Probabilidades de la clase positiva

     # Calcular la curva ROC
     fpr, tpr, thresholds = roc_curve(y_test, y_probs)

     auc = roc_auc_score(y_test, y_probs)
     auc

     0.7287775903072576
```

E hicimos su respectiva gráfica

```
[ ]  plt.figure(figsize=(8, 6))
     plt.plot(fpr, tpr, color='blue', lw=2, label=f'ROC curve (area = {auc:.2f})')
     plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
     plt.xlabel('False Positive Rate')
     plt.ylabel('True Positive Rate')
     plt.title('Receiver Operating Characteristic (ROC) Curve')
     plt.legend(loc='lower right')
     plt.show()
```



4. Retos

Se encontraron pocos retos en el desarrollo de la competencia, siendo el principal reto el entendimiento de la métrica propuesta por la competencia (ROC-AUC), la cual mediante distintas consultas por parte de los miembros del equipo, logramos resolver con ejemplos encontrados en tutoriales en internet y foros de desarrollo.

5. Conclusiones

- La implementación de modelos avanzados de machine learning, como los utilizados en este proyecto, puede ayudar a reducir significativamente las pérdidas financieras debido a fraudes y aumentar la seguridad de las transacciones para los usuarios.
- Los modelos utilizados muestran un potencial significativo para el aprendizaje continuo y la adaptación. En un campo que cambia rápidamente como la detección

de fraude, la capacidad de un modelo para adaptarse a nuevas tendencias y tácticas fraudulentas es crucial. El uso de técnicas como el aprendizaje no supervisado y la optimización de hiperparámetros sugiere que los modelos pueden ser ajustados continuamente para mantener su relevancia y efectividad.

- El manejo de datos sensibles, como transacciones con tarjetas de crédito, lleva implícitas importantes consideraciones éticas y de privacidad. El proyecto demuestra una conciencia de estas cuestiones al utilizar datos transformados mediante PCA, lo que ayuda a anonimizar la información.
- El proyecto contiene un enfoque metodológico riguroso, utilizando técnicas de preprocesamiento de datos, como la imputación de valores faltantes y la codificación de variables categóricas.