Fraude de Proveedores Médicos

Elias Yañez, Instituto Tecnológico de Estudios Superiores Monterrey, A01028482@tec.mx

Introducción:
Se estima que alrededor del 5 al 10% de las transacciones en reclamaciones de seguros son fraudulentas/abusos. Dentro de ellas, una parte importante se debe no a los clientes/beneficiarios de los seguros sino de los proveedores que prestan los servicios.
En este reto se toma un dataset con transacciones de seguros con información de los padecimientos, beneficiarios y proveedores. También una serie de etiquetas sobre qué proveedores se han considerado como fraudulentos. La meta normal del dataset es utilizar los datos disponibles para determinar los casos fraudulentos, esto nos deja con varios problemas.

- Poder determinar un caso como fraudulento, es un proceso complicado y los trámites representan un reto muy grande y costoso para la aseguradora.
- Las aseguradoras tienen recursos limitados, no pueden lanzar una investigación sobre 1000 proveedores instantáneamente y es probable que muchos casos no procedan. Ya sea porque eran falsos positivos o no había suficiente evidencia.
- Ya que se necesita presentar evidencia. Un modelo tipo caja negra no es suficiente para la aseguradora. Por los puntos anteriores, no necesariamente queremos no alertar o marcar como sospechosos a un número menor de proveedores solo porque es un reto para la aseguradora. Sino que necesitamos hacer dos cosas para ellos. Dar un Score de alerta para que sepan con qué proveedores empezar y hacer explicables nuestras decisiones.

1. **Estructura de Datos y Archivos**

Los datos incluyen detalles sobre beneficiarios, episodios hospitalarios, consultas ambulatorias, y detalles de reembolsos. Los principales archivos procesados son:

- **Archivos de beneficiarios** (Train_Beneficiarydata, Test_Beneficiarydata), que incluyen detalles personales y condiciones crónicas de los beneficiarios.
- **Archivos de datos hospitalarios** (Train_Inpatientdata, Test_Inpatientdata), que contienen información sobre admisiones, reembolsos y códigos de diagnóstico.
- **Archivos de consultas externas** (Train_Outpatientdata, Test_Outpatientdata), con información similar para consultas fuera del hospital.

2. **Preparación de Datos**

- **Carga y Exploración Inicial de los Datos:**

- Todos los archivos CSV y ZIP se cargan desde Google Drive en un entorno de Colab.
- Se revisa cada archivo para verificar los nombres de columnas y tipos de datos, y para observar valores faltantes en columnas relevantes (e.g., códigos de

diagnóstico y códigos de procedimientos).

- **Transformación de los Datos:**
- **Conversión de Tipos de Datos:** Las fechas se convierten en formato de fecha (datetime) para calcular duraciones de reclamaciones.
- **Limpieza de Datos:** Las columnas con muchos valores faltantes, como OtherPhysician o algunos códigos de diagnóstico/procedimiento, se manejan mediante imputación o eliminación si no son informativas.
- **Creación de Características Derivadas:** Duración de las reclamaciones: Se calcula el número de días entre ClaimStartDt y ClaimEndDt. Frecuencia de Reclamaciones: Para cada beneficiario, se calcula el número de reclamaciones realizadas en un período de tiempo.
- **Agrupación de Datos:** En los datos hospitalarios y ambulatorios, se agrupan reclamaciones por beneficiario (BeneID) y se crean características agregadas, como el promedio y la suma de los montos reembolsados.

- **Reducción de Cardinalidad**: Los valores de baja frecuencia (aquellos que aparecen en pocas observaciones) se agrupan en una categoría única denominada "Other".

- **Imputación de Valores Faltantes:** Para columnas numéricas (e.g., InscClaimAmtReimbursed), se reemplazan los valores faltantes con la media de la columna. Para columnas categóricas (e.g., AttendingPhysician), los valores faltantes se reemplazan por "Unknown" para indicar que no se tiene información sobre el valor real.

- **Codificación One-Hot:** Se aplica One-Hot Encoding, que crea nuevas columnas binarias para cada categoría dentro de una variable. Por ejemplo, si Gender tiene dos valores posibles (Male y Female), se crean dos columnas binarias: Gender_Male y Gender_Female.

- **Ajuste de Balance de Clases** El dataset inicial presentaba un problema de desbalance de clases, donde los casos etiquetados como "fraude" (PotentialFraud=1) eran significativamente menos frecuentes que los no fraudulentos (PotentialFraud=0). Este desbalance puede causar que los modelos de machine learning sean menos efectivos al priorizar la clase mayoritaria.

  Para abordar este problema, se implementó un método de undersampling utilizando la biblioteca imblearn. Este enfoque reduce la cantidad de muestras de la clase mayoritaria para igualar la cantidad de la clase minoritaria, garantizando un conjunto de datos balanceado.

3. Modelado de Machine Learning

Se dividen los datos en conjuntos de entrenamiento y prueba. También se crea una columna de label que indica si una reclamación es fraudulenta o no.

- **Random Forest:**
  Se utiliza como primer modelo de referencia debido a su capacidad para manejar grandes volúmenes de datos con muchas características, además de su robustez frente a outliers.
- **XGBoost:**
  Se implementa para comparar el rendimiento y optimizar la detección de fraudes, dado que XGBoost es efectivo en tareas de clasificación con datos desbalanceados.

4. Predicción

Para evaluar la precisión y efectividad de los modelos en la detección de fraudes, se midió el accuracy de ambos modelos utilizando un conjunto de pruebas independiente. A continuación, se presentan los resultados:

- **Random Forest:**
  Accuracy: 0.8449 (84.49%)
  Este modelo logró una precisión aceptable, capturando la mayoría de los patrones de fraude en los datos. Sin embargo, dado el valor del accuracy, puede existir margen para mejorar la sensibilidad o especificidad en la detección de fraudes.

- **XGBoost**:
  Accuracy: 0.8731 (87.31%)
  Este modelo presentó un desempeño superior, posiblemente debido a su capacidad para optimizar los pesos de las observaciones desbalanceadas y ajustar errores en cada iteración.
  La mayor precisión indica una capacidad más efectiva para identificar fraudes y reducir los falsos positivos.

5. Reglas Manuales para Detección de Fraudes

Con base en el análisis exploratorio y la comprensión del dominio del problema, se desarrollaron tres reglas manuales diseñadas para identificar comportamientos anómalos entre los proveedores. Estas reglas buscan capturar patrones estadísticos que podrían indicar fraude.

- **Frecuencia de Tratamientos por Paciente**
  Evalúa si un proveedor administra un número inusualmente alto de tratamientos por paciente. Esto puede indicar un posible sobretratamiento para inflar las reclamaciones.
  Proveedores con valores extremos tienen mayor probabilidad de estar involucrados en actividades fraudulentas.
- **Reembolsos Totales por Paciente**
  Analiza el monto total de reembolsos obtenidos por un proveedor en relación con su número de pacientes. Valores anormalmente altos en esta métrica pueden indicar reclamaciones infladas.
  Proveedores con reembolsos altos por paciente, combinados con un bajo número de tratamientos, podrían estar inflando los costos.
- **Distribución de Condiciones Crónicas**
  Evalúa si un proveedor trata a una proporción inusualmente alta de pacientes con condiciones crónicas específicas que suelen estar asociadas con tratamientos costosos.
  Proveedores que se desvían significativamente del patrón promedio podrían estar seleccionando pacientes con

condiciones más costosas para justificar mayores reclamaciones.

6. **Reentrenamiento del Modelo de Machine Learning con Reglas Manuales**

Integra las reglas manuales como nuevas características en el dataset y reentrenar los modelos de Machine Learning para mejorar la precisión en la predicción de fraudes.
Cada regla se calculó utilizando transformaciones sobre las columnas relevantes, creando valores numéricos adicionales que representan el score de cada regla para cada proveedor.

- **Random Forest**
  El modelo se reentrenó utilizando el dataset enriquecido.
  Parámetros clave como la profundidad máxima (max_depth) y el número de estimadores (n_estimators) se ajustaron para optimizar el desempeño.
  Resultado: Accuracy = 0.9279 (92.79%).
- **XGBoost**
  El modelo se reentrenó utilizando un enfoque similar.
  Se ajustaron los hiperparámetros como el learning rate, el número de árboles (n_estimators) y la regularización.
  Resultado: Accuracy = 0.9427 (94.27%).

La incorporación de reglas manuales proporcionó información adicional valiosa al modelo, mejorando su capacidad para detectar patrones de fraude que no estaban representados adecuadamente en las características originales. Este enfoque híbrido (manual + automático) resultó en un modelo más robusto y explicable.