
UNIVERSIDAD
EAFIT

Análisis post operativo de investigaciones de fraudes en Aguas EPM, utilizando procesamiento de lenguaje natural

Maestría en Ciencias de los Datos y Analítica
Proyecto Integrador
2025-1

Grupo # 3:

- Jose Luis Bedoya Martinez, jlbedoyam@eafit.edu.co
- Álvaro Javier Mutis Guerrero, ajmutisg@eafit.edu.co
- Juan Luis Amaya Arbeláez, jlamayaa@eafit.edu.co
- Kevin Daniel Genez Valencia, kgenezv@eafit.edu.co

Índice

01. Introducción

Esta presentación detalla el análisis post operativo de investigaciones de fraudes en EPM Aguas, utilizando técnicas avanzadas de procesamiento de lenguaje natural. Exploraremos cómo la minería de texto y conceptos de álgebra lineal y estadística nos permiten clasificar y validar la efectividad de las investigaciones en campo, optimizando la detección de anomalías y fraudes.

02. Descripción del Problema

El Reto Actual

La gestión de pérdidas comerciales en EPM Aguas requiere investigaciones en campo para detectar fraudes. La información clave se encuentra en el campo "Observación", un texto libre que describe la situación. Validar la efectividad de las investigaciones y retroalimentar al contratista es un proceso manual y laborioso.

03. Nuestra Solución

Implementamos una solución que, usando el campo "Observación" y otros predictores, clasifica si una investigación corresponde a fraude o anomalía. Esto implica técnicas de procesamiento de lenguaje natural para extraer información no estructurada y mejorar la precisión del seguimiento post operativo.

04. Ciclo de Vida de los Datos y Procesamiento Analítico

Data Source

Datos estructurados de Hana e Hidro, con el campo "Observación" como texto no estructurado.

Se usan archivos .csv de apoyo para diccionario y stopwords.

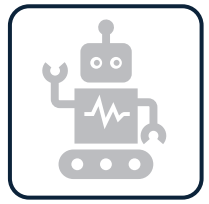
Ingesta de Datos

Ingesta semanal de 500 registros (1 GB) en S3 de AWS.

Entorno virtual en EMR con Pandas, NumPy, SciPy, scikit-learn, python-Levenshtein y spaCy.

Storage

Almacenamiento en Amazon S3 para analítica descentralizada, con bajo requerimiento de espacio.



Procesamiento (ETL)

Conexión a S3, carga de datos y asignación de tipos.

Importación de diccionario inicial y stopwords para preprocesamiento.

Procesamiento con Ciencia de Datos

Uso de Jupyter Hub con Pandas, Numpy, matplotlib, Scikit-learn y Spacy para procesamiento y entrenamiento del modelo de ML.

Aplicaciones

Modelo de clasificación multiclase con nltk, imbalanced-learn y xgboost.

Despliegue en Streamlit para automatizar preprocesamiento e inferencia.

04. Ciclo de Vida de los Datos y Procesamiento Analítico

Arquitectura de la Solución:

Excel / Macro
- Batch

[S3 - Raw]
- Datos crudos,Diccionarios y Stopwords

EMR + JupyterHub
- ETL
- NLP
- Training

[S3 - Refined]
Modelo (.joblib)

Streamlit APP

Predicción + Exportación

05. Procesamiento de Lenguaje Natural

Limpieza y Depuración

- Funciones para depurar observaciones (extraer cuerpo), remover etiquetas de causa evento y quitar información de sellos usando expresiones regulares.

Normalización de Texto

- Eliminación de números y caracteres especiales.
- Tokenización, lematización y categorización gramatical (POS) para seleccionar palabras de valor.

Diccionario y Stopwords

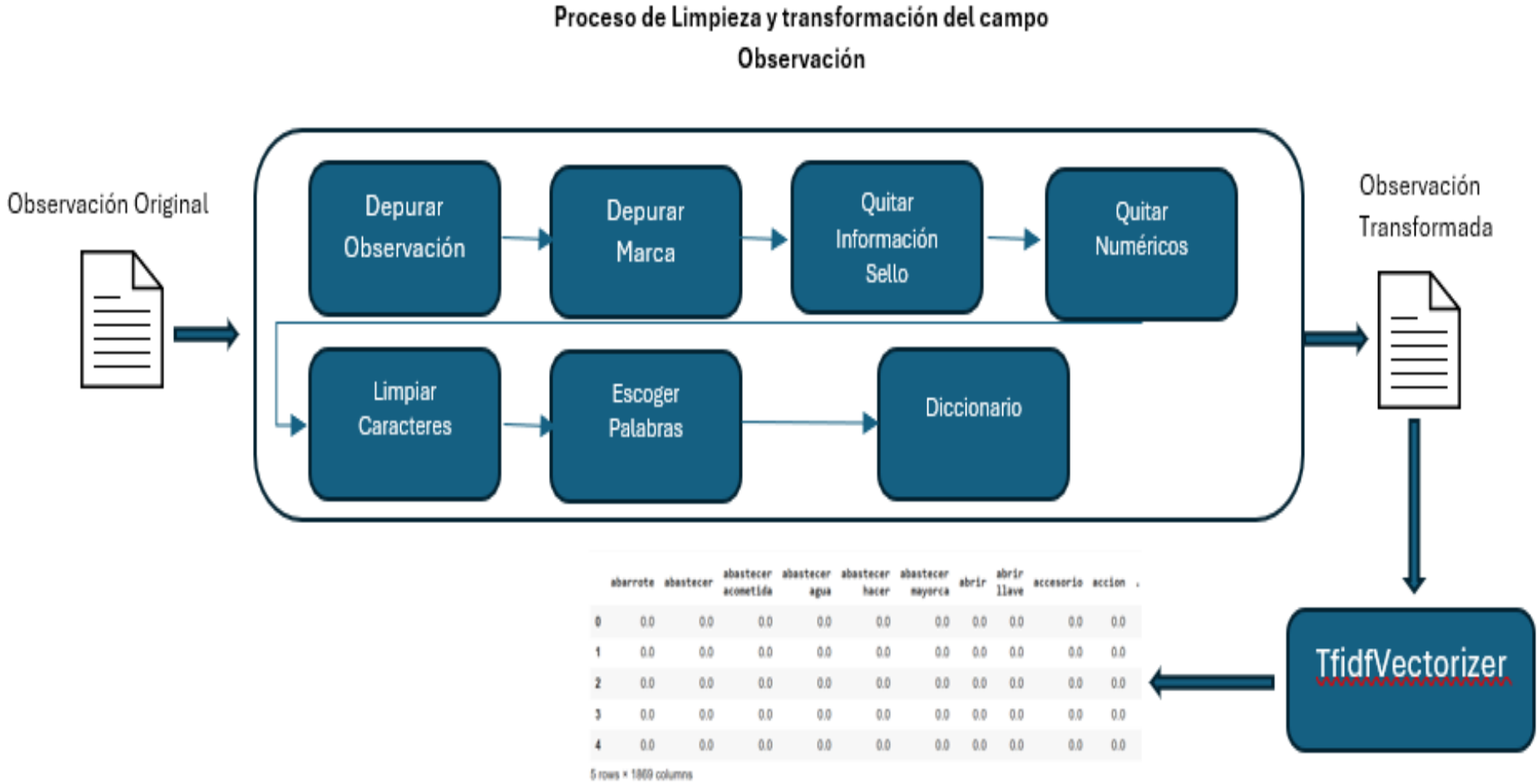
- Uso de un diccionario para homologar palabras y corregir errores.
- Eliminación de stopwords para reducir ruido y columnas en el set de entrenamiento.

Manejo de Palabras Nuevas

- Identificación y almacenamiento de palabras no presentes en el entrenamiento
- Marcado como "imputada" y actualización del diccionario para reentrenamiento.

Minería de Texto

05. Procesamiento de Lenguaje Natural



	abarrote	abastecer	abastecer acometida	abastecer agua	abastecer hacer	abastecer mayorca	abrir	abrir llave	accesorio	accion	.
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

5 rows × 1869 columns

05. Procesamiento de Lenguaje Natural – Función Diccionario

```python
def diccionario(frase):
    f = []
    doc = nlp(frase)
    for token in doc:
        palabra = token.text.lower()
        if palabra in dict_stopword:
            continue
        elif palabra in palabras_nuevas:
            f.append('imputada')
        elif palabra in remplazos:
            f.append(remplazos[palabra])
        elif palabra in dict_sinonimos:
            f.append(dict_sinonimos[palabra][0])
        elif any(palabra in syn_list for syn_list in dict_sinonimos.values()):
            f.append(palabra)
        else:
            mejor = ''
            porcentaje = 0
            for key in dict_sinonimos:
                dist = levenshtein(palabra, key)
                max_len = max(len(palabra), len(key))
                sim = 1 - (dist / max_len)
                if sim > porcentaje:
                    porcentaje = sim
                    mejor = dict_sinonimos[key][0]
            if porcentaje >= 0.7:
                remplazos[palabra] = mejor
                f.append(mejor)
            else:
                palabras_nuevas.append(palabra)
                f.append('imputada')
    return " ".join(f)
```

index	Observacion	Observaciones_Limpias
1	Carlos Jaramillo Medidor Water tech de ½ serie#20113047667 con lectura 03164 en posición correcta sin signos de manipulación externa registra en prueba de llaves predio no residencial actividad económica es un bar razón social bar el Oscar el cual tiene 2 puntos de consumo el cual usuario manifiesta que el consumo varía de acuerdo a la cantidad de personas que transcurren en el lugar Usuario: EPM\CARLOS.JARAMILLO Fecha: 2021-08-05 15:13:27	serie lectura posicion correcta sin signo manipulacion externa registra prueba llave predio no residencial actividad economica bar imputada social bar tener punto consumo usuario manifiesta consumo varian imputada cantidad persona transcurso lugar
0	Se encuentra vivienda sin medidor, con servicio directo. Usuario dice que nunca ha tenido medidor de acueducto. Usuario: EPM\JACOSAC Fecha: 2021-09-13 10:43:15	encontrar vivienda sin medidor servicio directo decir tener medidor acueducto
2	Yonatan Giraldo predio solo no hay quien suministre información medidor bien instalado con cúpula impactada medidor marca wáter tech de 1/2 serie 2015450174 lectura ilegible se envía medidor para el laboratorio en bolsa de seguridad BM -17 00236 presinto CB 17-00366 se normaliza instalación mediante cambio de medidor provisional se dejan llaves y racores sin fugas fecha laboratorio 01-09-2021 predio residencial Usuario: EPM\YONATAN.GIRALDO Fecha: 2021-08-19 13:22:16	predio no suministrar informacion medidor instalado cupula impactada serie lectura ilegible enviar medidor parar laboratorio bolsa seguridad precinto normalizar instalacion mediante cambio provisional dejar llave racor sin fuga fecha laboratorio predio residencial

06.

Implementación de Conceptos de Álgebra Lineal

Limpieza y Filtrado con Distancia Levenshtein.

- La distancia Levenshtein se usa para identificar y corregir palabras similares debido a errores de digitación u ortografía.
- Se calcula un porcentaje de similitud ponderado por la longitud de la palabra para mayor precisión.

A

Reducción de Dimensionalidad.

- La matriz dispersa (2.877x1.869) resultante de la observación se reduce.
- Se analizan la determinante (cercana a 0), el índice de condicionalidad (4.21×10^{18}) y la traza (1.858) para identificar colinealidad.

B

Eliminación de Variables Colineales.

- Se eliminan variables con producto interno absoluto superior a 0.9, priorizando la de menor varianza.
- También se descartan variables con eigenvalores menores a 0.01, logrando una reducción de 239 variables.

C

06.

Implementación de Conceptos de Álgebra Lineal – Función análisis_varianza_producto_interno

Metrica	Valor	Analisis
Determinate	0	Una matriz de covarianza con un determinante cercano a cero indica alta colinealidad entre las variables
Indice de condicion	4,21E+18	Es un alto indice de condicion lo que afirma lo encontrado con el calculo de determinantes: existen variables con alta colinealidad
Traza	1,858	Es la suma de todas las varianzas individuales. En si misma no es una métrica que nos permita establecer colinealidad

```python
# Escalar datos
X_scaled = StandardScaler().fit_transform(df)
cov_matrix = np.cov(X_scaled.T)
eigenvalues, eigenvectors = np.linalg.eig(cov_matrix)
# Redundancia por producto interno
redundantes_pi = set()
n = len(columnas)
for i in range(n):
    if columnas[i] in redundantes_pi:
        continue
    v1 = X_scaled[:, i]
    for j in range(i + 1, n):
        if columnas[j] in redundantes_pi:
            continue
        v2 = X_scaled[:, j]
        prod = np.dot(v1, v2) / len(v1)  # Normalizado al tamaño
        if abs(prod) >= umbral_producto:
            # Eliminar la que tenga menor varianza
            var1 = np.var(v1)
            var2 = np.var(v2)
            redundantes_pi.add(columnas[j] if var1 >= var2 else columnas[i])
# Redundancia por eigenvalores bajos
redundantes_eigen = [columnas[i] for i, eig in enumerate(eigenvalues) if eig < umbral_eigen]

# Unificar variables redundantes
redundantes_totales = set(redundantes_pi).union(set(redundantes_eigen))
df_reducido = df.drop(columns=redundantes_totales) if reducir else df.copy()

return {
    'Traza (Varianza Total)': np.trace(cov_matrix),
    'Determinante': np.linalg.det(cov_matrix),
    'Norma Frobenius': np.linalg.norm(cov_matrix, ord='fro'),
    'Eigenvalores': eigenvalues,
    'Variables redundantes (producto interno)': list(redundantes_pi),
    'Variables redundantes (eigenvalor)': redundantes_eigen,
    'Variables eliminadas': list(redundantes_totales),
    'DataFrame reducido': df_reducido
}
```

07. Implementación de Conceptos de Estadística para Analítica: EDA y Limpieza

1

Identificación de Faltantes

Se identificaron valores faltantes en "Causa Efectividad" (35%), variables numéricas (9%) y "Observación" (8.7%).

2

Eliminación por Fechas

Se eliminaron filas sin fechas clave, reduciendo el dataset de 3,150 a 2,895 observaciones para asegurar la integridad temporal.

07. Implementación de Conceptos de Estadística para Analítica: EDA y Limpieza

3

Imputación Categórica

Los valores faltantes en "Causa Efectividad" se imputaron con "No Aplica".

Campos de texto vacíos se etiquetaron como "Desconocido".

4

Análisis Descriptivo Numérico

Se calcularon estadísticos básicos.

Variables como "Consumo promedio" mostraron alta asimetría y curtosis, indicando outliers.

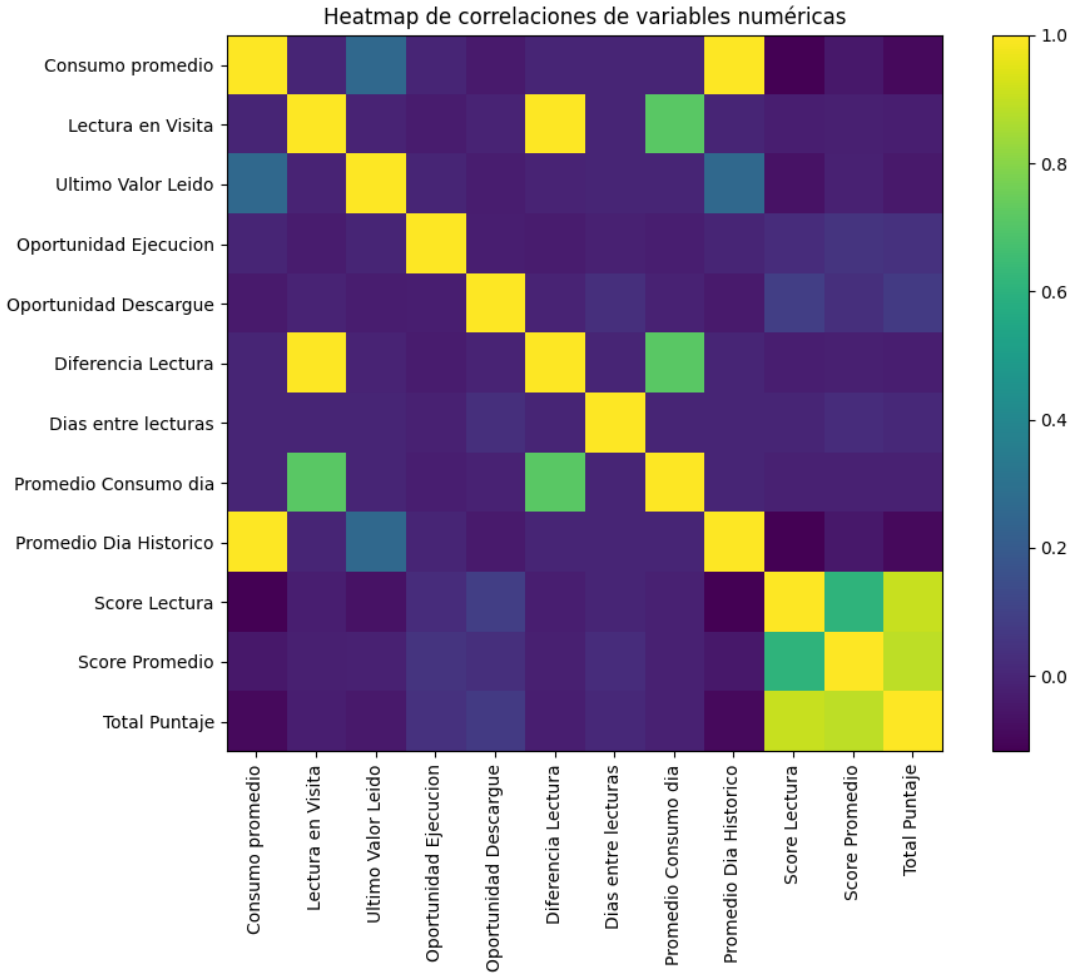
07. Implementación de Conceptos de Estadística para Analítica: EDA y Limpieza

5

Agrupación Categórica

Categorías minoritarias en "Estado Instalación", "Uso" y "Actividad Económica" se agruparon para simplificar y evitar ruido.

Análisis de Correlaciones y Reducción de Variables



- Se construyó un mapa de calor para analizar correlaciones entre variables numéricas.
- Se observó alta correlación entre "Consumo promedio" y "Promedio Día Histórico", así como entre "Lectura en Visita" y "Diferencia Lectura", indicando redundancia.
- Las métricas de puntaje interno también mostraron fuerte correlación.
- Para reducir variables, se eliminaron aquellas con correlación absoluta mayor a 0.7 con otras variables, priorizando la de menor correlación con la variable objetivo. Se descartaron "Promedio Día Histórico", "Lectura en Visita", "Promedio Consumo día" y "Total Puntaje".
- Para variables categóricas, se usó la prueba chi cuadrado o Fisher, eliminando "Estado de instalación" por su menor aporte a la variable objetivo.

Análisis de Correlaciones y Reducción de Variables

Análisis de Correlaciones

- Se construyó un mapa de calor para analizar correlaciones entre variables numéricas.
- Se observó alta correlación entre "Consumo promedio" y "Promedio Día Histórico", así como entre "Lectura en Visita" y "Diferencia Lectura", indicando redundancia.
- Las métricas de puntaje interno también mostraron fuerte correlación.

A

Reducción de variables Numéricas

- Para eliminar variables con alta correlación se implementa la función correlacion_numericas que examina cada par de variables numéricas presentes en el data set y al encontrar una alta correlación, que en nuestro caso es un valor absoluto mayor a 0.7 elimina aquella cuya correlación con la variable objetivo sea menor.
- El resultado de este proceso es la eliminación de las siguientes columnas:
  - Promedio Dia Histórico
  - Lectura en Visita
  - Promedio Consumo día
  - Total Puntaje

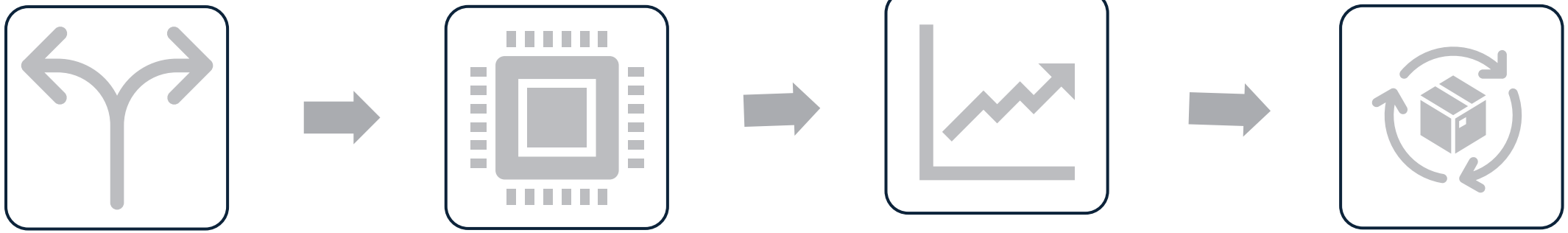
B

Reducción de variables Categóricas

- Se implementa la función chi_sq_fisher que establece tablas de contingencia para cada par de variables y aplica la prueba chi cuadrado cuando la tabla de contingencia es mayor de 2X2 o la prueba exacta de fisher para los casos de tablas de contingencia de 2X2.
- Al aplicar la prueba y encontrar que rechazar la hipótesis nula de independencia se pasa a examinar cuál de las dos tiene mayor aporte a la variable objetivo.
- Luego de realizar la prueba se descarta la variable “Estado de instalación”

C

08. Clasificación Supervisada y Despliegue



División de Datos

Conjunto dividido en 70% entrenamiento y 30% prueba, estratificado para mantener proporciones de clases.

Preprocesamiento

Imputación de medianas y escalado para numéricas.

Imputación de "missing" y OneHotEncoder para categóricas.

Modelos Evaluados

Regresión Logística
XGBoost
Árbol de Decisión
SVM
Random Forest
LDA/QDA.
XGBoost fue el de mejor rendimiento.

Despliegue

Pipeline final serializado y subido a S3.
Aplicación web con Streamlit para inferencia automatizada y descarga de predicciones.

- El modelo final, XGBoost, se entrenó con los hiperparámetros óptimos. La aplicación Streamlit permite a los usuarios subir un CSV y recibir predicciones clasificadas como "Anomalía" (56.85%), "Normal" (39.89%) o "No Revisado" (3.26%), facilitando la validación y retroalimentación.

08. Clasificación Supervisada y Despliegue – Modelos e Hiper parametros

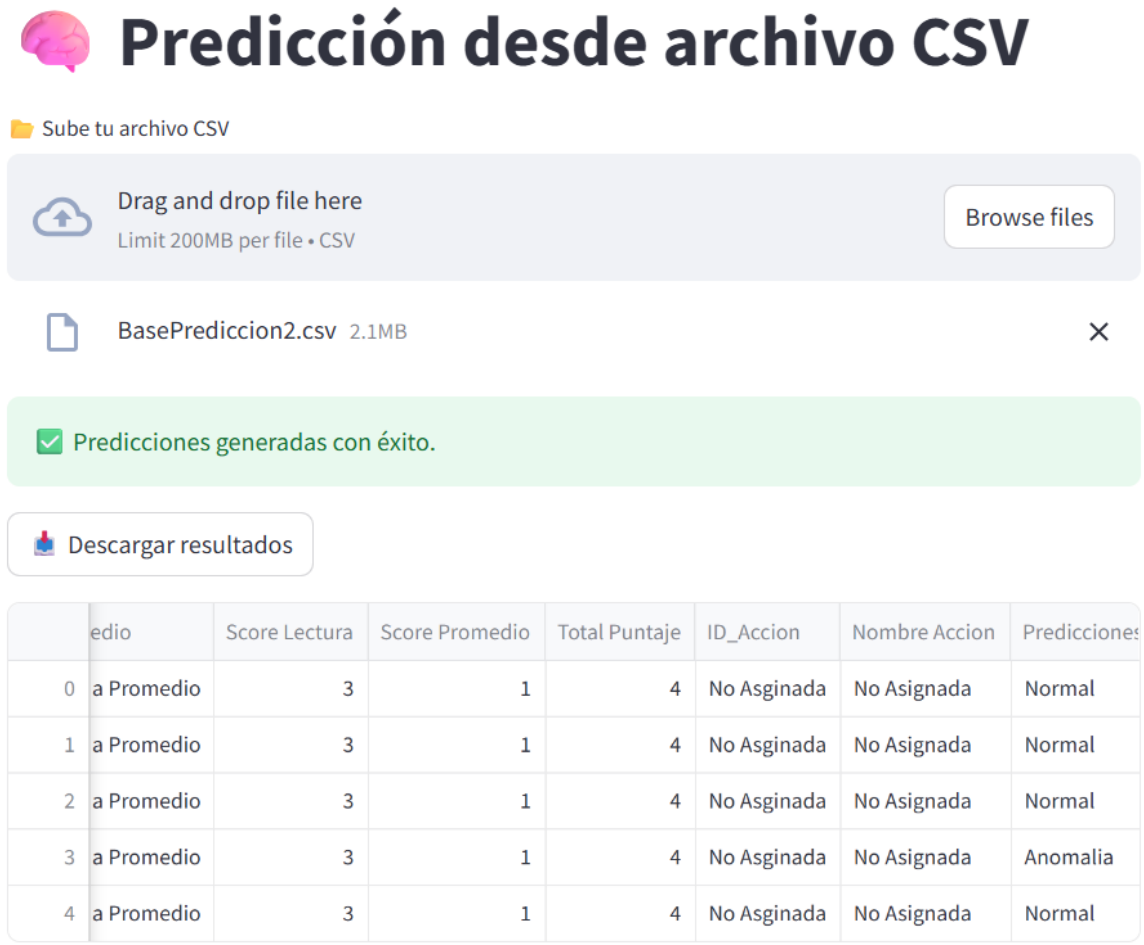
Modelo	Hiperparámetros
Regresión Logística	C: [0.01, 0.1, 1]; solver: ['lbfgs', 'liblinear']; penalty: ['l2']
XGBoost	n_estimators: [100, 200, 300]; max_depth: [3, 5, 7, 10]; learning_rate: [0.01, 0.1, 0.2]
Árbol de Decisión	criterion: ['gini', 'entropy']; max_depth: [5, 10, 20]
SVC	C: [1, 10]; gamma: ['scale', 'auto']
Random Forest	n_estimators: [10, 50]; criterion: ['gini', 'entropy']; max_depth: [5, 10]
LDA	solver: ['svd', 'lsqr']
QDA	reg_param: [0.1, 0.5, 1.0]

Modelo	F1-score	Hiperparámetros
XGBoost	0.549	learning_rate=0.2, max_depth=5, n_estimators=200
Árbol de Decisión	0.5154	criterion='entropy', max_depth=10
Random Forest	0.5069	criterion='entropy', max_depth=10, n_estimators=50
QDA	0.4517	reg_param=1.0
Regresión Logística	0.4506	C=1, penalty='l2', solver='lbfgs'
LDA	0.4391	solver='lsqr'
SVC	0.4306	C=10, gamma='auto'

09. Despliegue y Resultados de la Aplicación

Predicción desde archivo CSV

- Hemos implementado una aplicación web interactiva con Streamlit para la inferencia de modelos.
- Esta herramienta permite a los usuarios cargar datos y obtener predicciones instantáneas.
- Facilita la validación y el análisis post-operativo de los resultados.



	edio	Score Lectura	Score Promedio	Total Puntaje	ID_Accion	Nombre Accion	Predicciones
0	a Promedio	3	1	4	No Asginada	No Asignada	Normal
1	a Promedio	3	1	4	No Asginada	No Asignada	Normal
2	a Promedio	3	1	4	No Asginada	No Asignada	Normal
3	a Promedio	3	1	4	No Asginada	No Asignada	Anomalia
4	a Promedio	3	1	4	No Asginada	No Asignada	Normal

¡¡¡Gracias!!!