Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Puebla

Analítica de datos y herramientas de inteligencia artificial II (Gpo 101)

Actividad AG_4.1

Estudiantes:

María Matanzo Hermoso | A01737554

Marco Cornejo Cornejo | A01276411

Jorge Alberto Cortes Sánchez | A01736236

Eduardo Torres Naredo | A01734935

Laisha Fernanda Puentes Angulo | A01736397

19/10/2025

Reporte de Hallazgos: Actividad 4.1 Regresión Logística

Este reporte detalla el proceso de **limpieza de datos**, **transformación de variables** a un formato dicotómico y la aplicación de **cinco modelos de Regresión Logística** con sus respectivas métricas de desempeño.

1. Limpieza y Preparación de Datos

El análisis se centró en las columnas de tipo numérico del conjunto de datos original, las cuales son: Abundance_nbcell, TotalAbundance_SamplingOperation y Abundance_pm.

1.1. Detección y Tratamiento de Valores Atípicos (Outliers)

Se visualizó la presencia de valores atípicos mediante un **diagrama de caja horizontal** . Para definir los límites de detección.

- **Límites de Detección:**

```
Limite superior permitido Abundance_nbcell                    94.948382
TotalAbundance_SamplingOperation    437.096025
Abundance_pm                        233.878730
dtype: float64
Limite inferior permitido Abundance_nbcell                   -71.124054
TotalAbundance_SamplingOperation    374.808697
Abundance_pm                       -175.176372
dtype: float64
```

Los valores atípicos (aquellos fuera de estos límites) fueron tratados convirtiéndolos en **valores nulos (NaN)**.

1.2. Imputación de Valores Nulos

Tras la identificación y tratamiento de los valores atípicos, se encontraron las siguientes cantidades de valores nulos en las variables cuantitativas (luego de la limpieza):

```
1  valores_nulos=data3.isnull().sum()
2  valores_nulos
```

```
Abundance_nbcell                    37619
TotalAbundance_SamplingOperation    34628
Abundance_pm                        37352
dtype: int64
```

Los valores nulos se imputaron con la **mediana** de cada columna, redondeada a un decimal.

2. Conversión de Variables Categóricas a Numéricas (Dicotómicas)

Para aplicar la Regresión Logística (un modelo de clasificación binaria), las variables categóricas (`TaxonName`, `TaxonCode`, `SamplingOperations_code`, `CodeSite_SamplingOperations`, `Date_SamplingOperation`) fueron primero mapeadas a valores numéricos enteros basados en su orden de aparición y luego transformadas a variables **dicotómicas (binarias: 0 o 1)**.

2.1. Umbralización para Variables Dicotómicas

El umbral para la binarización de las variables numéricas y categóricas codificadas se determinó utilizando **percentiles** o la **fecha central**, según el tipo de variable.

Variable: 'TaxonName_num':

```
Q1 = Tabla_final_num['TaxonName_num'].quantile(0.25)
Q2 = Tabla_final_num['TaxonName_num'].quantile(0.50)
Q3 = Tabla_final_num['TaxonName_num'].quantile(0.75)

print("Q1:", Q1)
print("Q2:", Q2)
print("Q3:", Q3)
```

```
Q1: 352.0
Q2: 1196.0
Q3: 1656.0
```

Variable: 'TaxonCode_num':

```
P25 = np.percentile(Tabla_final_num['TaxonCode_num'], 25)
P50 = np.percentile(Tabla_final_num['TaxonCode_num'], 50)
P75 = np.percentile(Tabla_final_num['TaxonCode_num'], 75)
P90 = np.percentile(Tabla_final_num['TaxonCode_num'], 90)

print("P25:", P25, " P50:", P50, " P75:", P75, " P90:", P90)
```

```
P25: 352.0  P50: 1196.0  P75: 1656.0  P90: 2034.0
```

Variable: 'SamplingOperations_code_num':

```python
P25 = np.percentile(Tabla_final_num['SamplingOperations_code_num'], 25)
P50 = np.percentile(Tabla_final_num['SamplingOperations_code_num'], 50)
P75 = np.percentile(Tabla_final_num['SamplingOperations_code_num'], 75)
P90 = np.percentile(Tabla_final_num['SamplingOperations_code_num'], 90)

print("P25:", P25, " P50:", P50, " P75:", P75, " P90:", P90)
```

```
P25: 10699.0  P50: 21806.0  P75: 33679.0  P90: 42006.0
```

Variable: 'CodeSite_SamplingOperations_num':

```python
P25 = np.percentile(Tabla_final_num['CodeSite_SamplingOperations_num'], 25)
P50 = np.percentile(Tabla_final_num['CodeSite_SamplingOperations_num'], 50)
P75 = np.percentile(Tabla_final_num['CodeSite_SamplingOperations_num'], 75)
P90 = np.percentile(Tabla_final_num['CodeSite_SamplingOperations_num'], 90)

print("P25:", P25, " P50:", P50, " P75:", P75, " P90:", P90)
```

```
P25: 1351.0  P50: 2896.0  P75: 4697.0  P90: 6112.0
```

Variable: 'Date_SamplingOperation':

```python
Tabla_final2 = Tabla_final2.sort_values(by='Date_SamplingOperation', ascending=False)
Tabla_final2 = Tabla_final2.reset_index(drop=True)
indice_medio = len(Tabla_final2) // 2
fecha_central = Tabla_final2.loc[indice_medio, 'Date_SamplingOperation']
print("Fecha de en medio:", fecha_central)
```

```
Fecha de en medio: 2016-08-31
```

Esta transformación generó las variables dependientes binarias para los modelos de Regresión Logística.

3. Análisis de Regresión Logística

Se entrenaron **cinco modelos de Regresión Logística**, cada uno utilizando una de las variables dicotómicas como variable dependiente (y), y otras variables numéricas como independientes (x). Se aplicó **escalado estándar** a las variables independientes y una división de datos de **70% para entrenamiento y 30% para prueba**

Caso 1: Predicción de TaxonName_num

Clase 1 (Positivo, x ≥ 352) Clase 0 (Negativo, x ≤ 352)

Precisión del modelo (precision):

```
Precisión del modelo label 1:
0.7577301523116323
```

```
Precisión del modelo label 0:
0.539306305481834
```

Sensibilidad del modelo (recall):

```
Sensibilidad del modelo label 1:
0.9864106814095747
```

```
Sensibilidad del modelo label 0:
0.04801831752550438
```

Exactitud (Accuracy):

```
Exactitud del modelo:
0.7528905309005965
```

Puntaje F1 (F1-score):

```
Puntaje F1 del modelo label 1:
0.8570788567246764
```

```
Puntaje F1 del modelo label 0:
0.08818489947699605
```

Matriz de confusión:

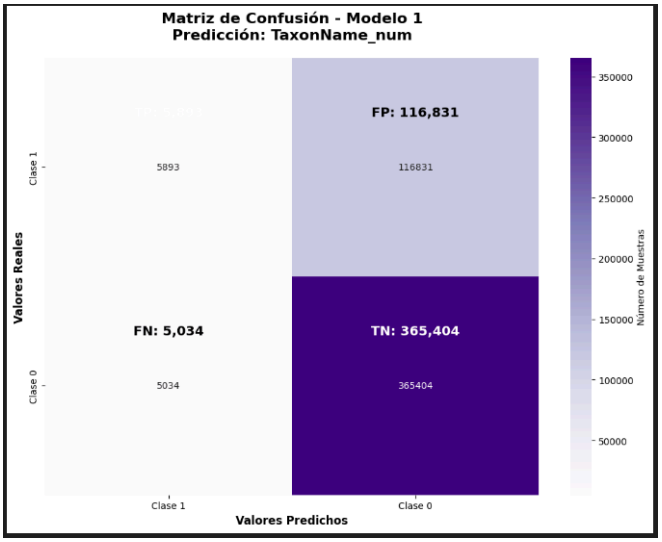
```
Matriz de Confusión:
[[  5893 116831]
 [  5034 365404]]
```

TP (Clase 0): 5,893

FP (Clase 0): 116,831

FN (Clase 1): 5,034

TN (Clase 1): 365,404



Hallazgos: El Accuracy general es del **75.29%**, lo que indica que el modelo clasifica correctamente una parte significativa de los datos. Sin embargo, el recall para la Clase 0 es extremadamente baja **4.80%**, lo que sugiere que el modelo tiene serias dificultades para identificar correctamente los casos de **TaxonName_num** por debajo del umbral de **352**. La Precisión para la Clase 0 es de 53.93% que también es baja, reflejando muchos falsos positivos.

Caso 2: Predicción de TaxonCode_num

Clase 1 (Positivo, x ≥ 352) Clase 0 (Negativo, x ≤ 352)

Precisión del modelo (precision):

```
Precisión del modelo label 1:
0.7564450123102898
```

```
Precisión del modelo label 0:
0.536441828881847
```

Sensibilidad del modelo (recall):

```
Sensibilidad del modelo label 1:
0.9861552397820532
```

```
Sensibilidad del modelo label 0:
0.04803521771911761
```

Exactitud (Accuracy):

```
Exactitud del modelo:
0.7515177568425792
```

Puntaje F1 (F1-score):

```
Puntaje F1 del modelo:
0.8561598813050807
```

```
Puntaje F1 del modelo label 0:
0.08817489136258111
```

Matriz de confusión:

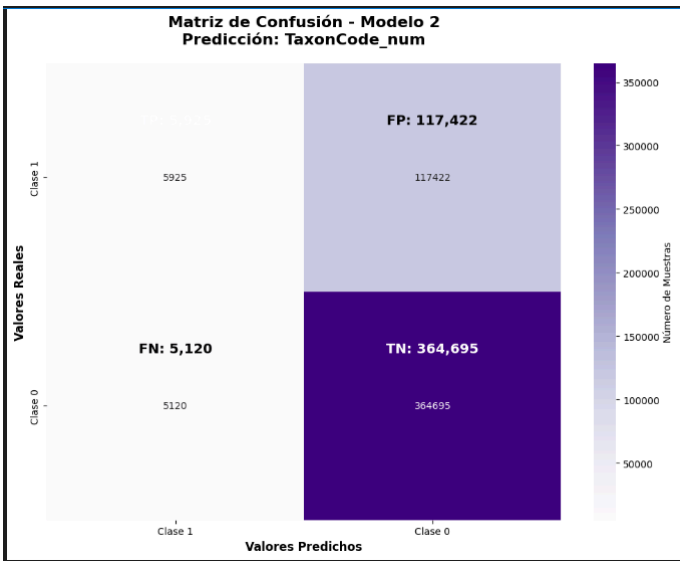
```
Matriz de Confusión:
[[  5925 117422]
 [  5120 364695]]
```

TP (Clase 0): 5,925

FP (Clase 0): 117,422

FN (Clase 1): 5,120

TN (Clase 1): 364,695



Hallazgos: El Modelo 2 presenta un desempeño muy similar al Modelo 1. Ya que el Accuracy es de **75.15%**. El Recall para la Clase 0 sigue siendo críticamente baja **4.80**%, lo que indica un problema persistente en la identificación de la clase minoritaria (posiblemente debido a un desbalance de clases).

Caso 3: Predicción de SamplingOperations_code_num:

Clase 1 (Positivo, x ≥ 21,806) Clase 0 (Negativo, x ≤ 21,806)

Precisión del modelo (precision):

```
Precisión del modelo label 1:
0.6123978146335127
```

```
Precisión del modelo label 0:
0.610479910533396
```

Sensibilidad del modelo (recall):

```
Sensibilidad del modelo label 1:
0.6108110734152207
```

```
Sensibilidad del modelo label 0:
0.6120671444124864
```

Exactitud (Accuracy):

```
Exactitud del modelo:
0.6114380264497264
```

Puntaje F1 (F1-score):

```
Puntaje F1 del modelo label 1:
0.6116034148674834
```

```
Puntaje F1 del  label 0:
0.611272497119395
```

Matriz de confusión:

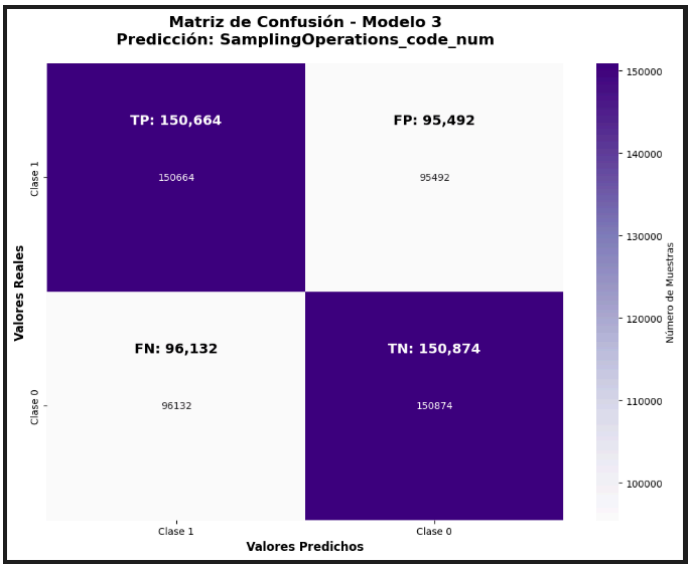
```
Matriz de Confusión:
[[150664  95492]
 [ 96132 150874]]
```

TP (Clase 1): 150,664

FP (Clase 1): 95,492

FN (Clase 0): 96,132

TN (Clase 0): 150,874



Hallazgos: Este modelo muestra una **distribución de métricas mucho más equilibrada** entre las clases. El Accuracy de **61.14** es menor que en los modelos anteriores, pero el Recall es consistentemente alrededor del **61%** para ambas clases. Esto sugiere que las variables independientes seleccionadas (CodeSite_SamplingOperations_num, Date_SamplingOperation) están igualmente correlacionadas con ambas categorías de la variable dependiente, indicando un desempeño justo y balanceado.

Caso 4: Predicción de CodeSite_SamplingOperations_num:

Clase 1 (Positivo, x ≥ 2896) Clase 0 (Negativo, x ≤ 2896)

Precisión del modelo (precision):

```
Precisión del modelo label 1:
0.610411551644505
```

```
Precisión del modelo label 0:
0.6096208492327677
```

Sensibilidad del modelo (recall):

```
Sensibilidad del modelo label 0:
0.6096381727174192
```

```
Sensibilidad del modelo label 1:
0.6103942405470995
```

Exactitud (Accuracy):

```
Exactitud del modelo:
0.6100165868416464
```

Puntaje F1 (F1-score):

```
Puntaje F1 del modelo label 1:
0.610402895973066
```

```
Puntaje F1 del modelo label 0:
0.6096295108520255
```

Matriz de confusión:

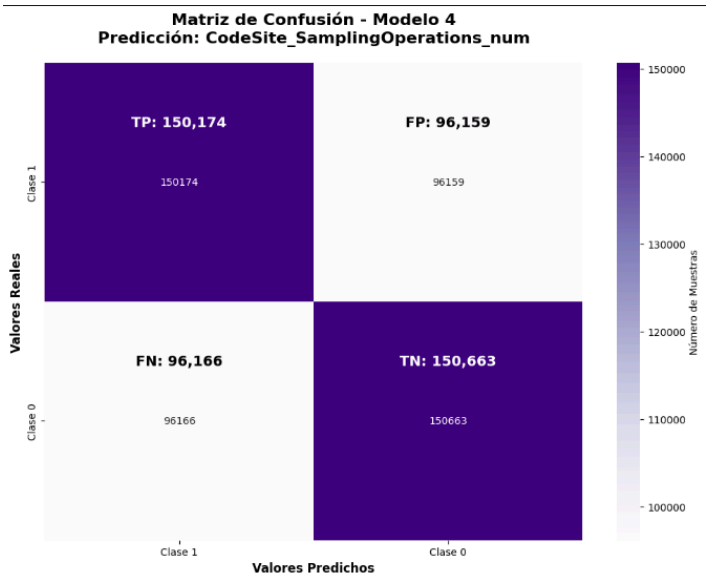
```
Matriz de Confusión:
[[150174  96159]
 [ 96166 150663]]
```

TP (Clase 0): 150,174

FP (Clase 0): 96,159

FN (Clase 1): 96,166

TN (Clase 1): 150,663



Hallazgos: El Modelo 4 también presenta un rendimiento equilibrado entre clases, con un Accuracy del **61.00%**. La consistencia en las métricas (todas alrededor del **61%** para ambas clases indica que la relación es débil pero sin sesgo significativo hacia una u otra clase.

Caso 5: Predicción de Date_SamplingOperation:

Clase 1 (Positivo, ≥ 2016-08-31) Clase 0 (Negativo, ≤ 2016-08-31)

Precisión del modelo (precision):

```
Precisión del modelo label 1:
0.5359145456240594
```

```
Precisión del modelo label 0:
0.5282006875212284
```

Sensibilidad del modelo (recall):

```
Sensibilidad del modelo label 1:
0.4884810715658701
```

```
Sensibilidad del modelo label 0:
0.5751543404308933
```

Exactitud (Accuracy):

```
Exactitud del modelo:
0.5317238554470944
```

Puntaje F1 (F1-score):

```
Puntaje F1 del modelo label 1:
0.5110996320587351
```

```
Puntaje F1 del modelo label 0:
0.5110996320587351
```

Matriz de confusión:

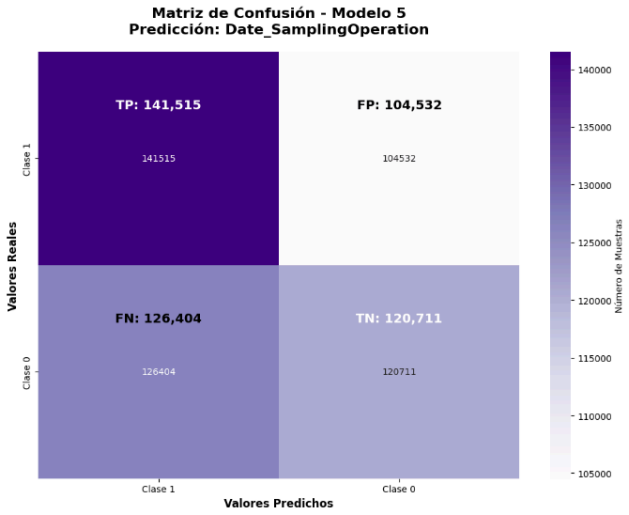
```
Matriz de Confusión:
[[141515 104532]
 [126404 120711]]
```

TP (Clase 0): 141,515

FP (Clase 0): 104,532

FN (Clase 1): 126,404

TN (Clase 1): 120,711



Hallazgos: Este modelo es el que presenta el **desempeño más bajo** en términos de Accuracy del 53.17%. Las métricas son bajas, aunque el Recall **de la Clase 0** 57.52% es ligeramente superior a la de la Clase 1 48.85% Un valor de exactitud tan cercano al 50% que sugiere que el modelo no tiene mucha más capacidad predictiva que una simple conjetura.

Conclusiones del Análisis de Correlación

1. **Desbalance y Desempeño Sesgado (Modelos 1 y 2):** Los Modelos 1 y 2, que predicen las variables binarias de los taxones (TaxonName_num y TaxonCode_num), muestran la **Exactitud más alta** (75%). Sin embargo, la **Sensibilidad es extremadamente baja para la Clase 0** (alrededor del 4.8%) y el alto número de falsos positivos (116,000) en la matriz de confusión, sugieren un problema de **desbalance de clases severo**. Es probable que la clase mayoritaria (Clase 1) sea la que esté impulsando la alta exactitud, mientras que la minoritaria no se predice correctamente.
2. **Desempeño Balanceado (Modelos 3 y 4):** Los Modelos 3 y 4, que predicen códigos de operación y sitio, muestran un **desempeño moderado pero equilibrado** (alrededor del **61%** en todas las métricas). Esto indica que la binarización de estas variables generó clases con una proporción más equitativa, y que la correlación con sus variables independientes es débil a moderada, pero sin un sesgo marcado.
3. **Bajo Poder Predictivo (Modelo 5):** El Modelo 5, que intenta predecir si una muestra es "reciente" o "antigua" (Date_SamplingOperation), tiene la **Exactitud más baja** (53.17%), lo que sugiere que las variables de abundancia utilizadas (TotalAbundance_SamplingOperation y Abundance_pm) tienen una **correlación muy débil** con el factor temporal (antes o después de 2016-08-31).