Segunda entrega de proyecto

POR:

Juan José Gomez Mejia

MATERIA:

Introducción a la inteligencia artificial

PROFESOR:

Raul Ramos Pollan

UNIVERSIDAD DE ANTIOQUIA

FACULTAD DE INGENIERÍA

MEDELLÍN 2023

1. Avances

1.1 Introducción

Para la clasificación de exoplanetas usando el dataset "**Kepler Exoplanet Search Results**.", primero es necesario entender la naturaleza del dataset y hacer un preprocesamiento antes de hacer cualquier entrenamiento; el análisis de los datos recopilados por los telescopios espaciales puede ser un desafío debido a la complejidad y el ruido que presentan. Es por eso que el preprocesamiento de los datos es esencial para intentar obtener resultados más precisos y confiables en la clasificación de exoplanetas.

En este avance, ya se realizaron técnicas de preprocesamiento de datos y las primeras pruebas usando modelos de aprendizaje automático (ML). Los modelos de ML son las herramientas que identifican patrones en los datos que no son evidentes a simple vista.

Recordemos que este proyecto tiene como objetivo aplicar técnicas de ML para mejorar la clasificación y por ende, la detección de exoplanetas.

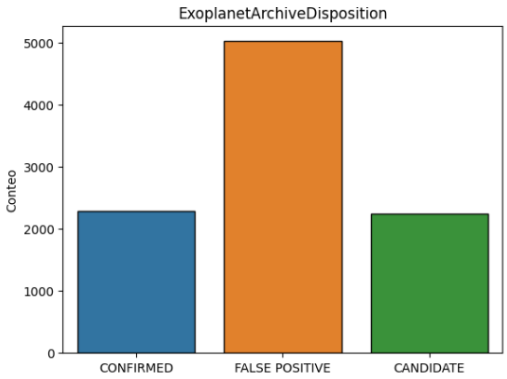
1.2 Preprocesamiento de datos:

a. Análisis exploratorio de datos y naturaleza del dataset.

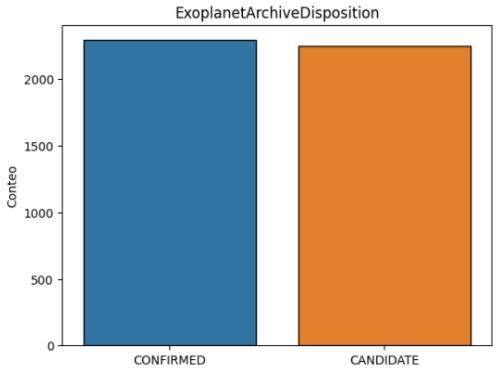
Inicialmente, se realizó un renombrado de columnas, para tener un mejor contexto de lo que significa cada característica.

Se analizó la variable objetivo **ExoplanetArchiveDisposition** y se tomó como decisión trabajar sólo con las categorías **CONFIRMED** y **CANDIDATE**; **FALSE POSITIVE** se excluirá.

Antes



Después



Se eliminaron columnas irrelevantes como el id de la fila, la **KepID** o las columnas que están completamente vacías como **EquilibriumTemperatureUpperUncK.**

Se utilizó One_Hot_Encoding para ordenar algunas columnas como valores categóricos, la **TCEDeliver.**

Se aplicaron Técnicas de Limpieza y transformación de los datos para reparar los valores nulos, utilizando la *moda* para datos categóricos y la *media* para datos continuos.

Finalmente, la estructura del dataset será la siguiente:

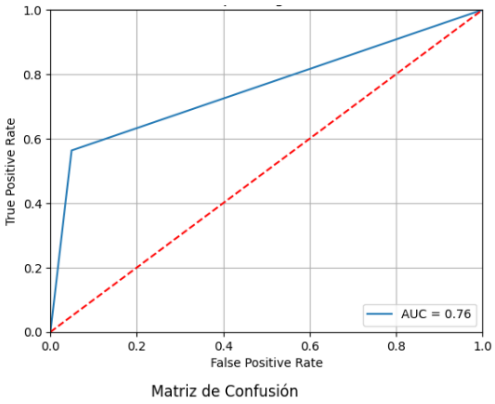
```
RangeIndex: 4541 entries, 0 to 4540
Data columns (total 45 columns):
 #   Column                                             Non-Null Count  Dtype
---  ------                                             --------------  -----
 0   ExoplanetArchiveDisposition                        4541 non-null   int64
 1   DispositionScore                                   4541 non-null   float64
 2   NotTransit-LikeFalsePositiveFlag                   4541 non-null   int64
 3   koi_fpflag_ss                                      4541 non-null   int64
 4   CentroidOffsetFalsePositiveFlag                    4541 non-null   int64
 5   EphemerisMatchIndicatesContaminationFalsePositiveFlag  4541 non-null   int64
 6   OrbitalPeriod_days                                 4541 non-null   float64
 7   OrbitalPeriodUpperUnc_days                         4541 non-null   float64
 8   OrbitalPeriodLowerUnc_days                         4541 non-null   float64
 9   TransitEpoch_BKJD                                  4541 non-null   float64
 10  TransitEpochUpperUnc_BKJD                          4541 non-null   float64
 11  TransitEpochLowerUnc_BKJD                          4541 non-null   float64
 12  ImpactParamete                                     4541 non-null   float64
 13  ImpactParameterUpperUnc                            4541 non-null   float64
 14  ImpactParameterLowerUnc                            4541 non-null   float64
 15  TransitDuration_hrs                                4541 non-null   float64
 16  TransitDurationUpperUnc_hrs                        4541 non-null   float64
 17  TransitDurationLowerUnc_hrs                        4541 non-null   float64
 18  TransitDepth_ppm                                   4541 non-null   float64
 19  TransitDepthUpperUnc_ppm                           4541 non-null   float64
 20  TransitDepthLowerUnc_ppm                           4541 non-null   float64
 21  PlanetaryRadius_Earthradii                         4541 non-null   float64
 22  PlanetaryRadiusUpperUnc_Earthradii                 4541 non-null   float64
 23  PlanetaryRadiusLowerUnc_Earthradii                 4541 non-null   float64
 24  EquilibriumTemperatureK                            4541 non-null   float64
 25  InsolationFlux_Earthflux                           4541 non-null   float64
 26  InsolationFluxUpperUnc_Earthflux                   4541 non-null   float64
 27  InsolationFluxLowerUnc_Earthflux                   4541 non-null   float64
 28  TransitSignal-to-Nois                              4541 non-null   float64
 29  TCEPlanetNumbe                                     4541 non-null   float64
 30  StellarEffectiveTemperatureK                       4541 non-null   float64
 31  StellarEffectiveTemperatureUpperUncK               4541 non-null   float64
 32  StellarEffectiveTemperatureLowerUncK               4541 non-null   float64
 33  StellarSurfaceGravity_log10(cm/s**2)               4541 non-null   float64
 34  StellarSurfaceGravityUpperUnc_log10(cm/s**2)       4541 non-null   float64
 35  StellarSurfaceGravityLowerUnc_log10(cm/s**2)       4541 non-null   float64
 36  StellarRadius_Solarradii                           4541 non-null   float64
 37  StellarRadiusUpperUnc_Solarradii                   4541 non-null   float64
 38  StellarRadiusLowerUnc_Solarradii                   4541 non-null   float64
 39  RA_decimaldegrees                                  4541 non-null   float64
 40  Dec_decimaldegrees                                 4541 non-null   float64
 41  Kepler-band_mag                                    4541 non-null   float64
 42  TCEDeliver_q1_q16_tce                              4541 non-null   int64
 43  TCEDeliver_q1_q17_dr24_tce                         4541 non-null   int64
 44  TCEDeliver_q1_q17_dr25_tce                         4541 non-null   int64
```

También se analizó la distribución y frecuencia de las características, así como la correlación entre ellas.

1.3 Selección y entrenamiento de modelos:

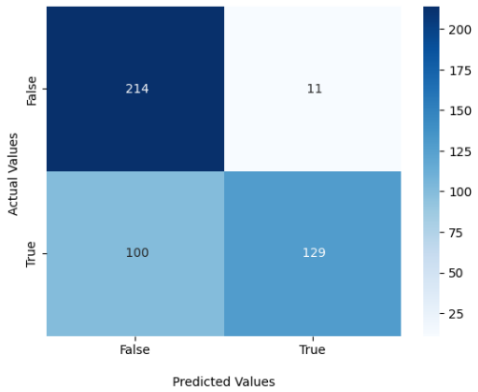
Inicialmente, se han seleccionado 3 modelos para el entrenamiento supervisado (Sólo es una decisión temporal la elección de ellos) y se ha utilizado una metodología de validación usando ***Validación-Cruzada*** con 10 folds para el entrenamiento; sin embargo, se espera también utilizar la técnica de ***Train-Test-Split***.
Y recordando que como métricas de evaluación utilizadas están el ***Accuracy***, ***Pecision***, ***Recall***, ***AUC*** y en general la ***Matriz de Confusión***.

Random forest: Funciona creando múltiples árboles de decisión en los que cada árbol es un clasificador que toma una decisión basada en un subconjunto aleatorio de características de los datos de entrenamiento.
En particular, para nuestro caso, algunos resultados son los siguientes:




```
--------------Métricas de todo el promedio -----------
Eficiencia - Exactitud: 0.790164593116135
Precision:  0.8052943785592065
Recall:  0.8411239794949686
Error de validación:  0.19470562144079354
Tiempo de ejecución:  402.4459102153778 segundos
```

Matriz de Confusión



En general, las métricas son buenas, dando resultados no perfectos pero apreciables, donde no es necesario un numero alto de estimadores.

Redes neuronales: Las redes neuronales se construyen a partir de capas de neuronas artificiales que se comunican entre sí a través de conexiones ponderadas. Cada neurona procesa la información de entrada y transmite la salida a las neuronas de la capa siguiente.

Para redes neuronales apenas se está ensamblando el código, aún no hay resultados disponibles, pero hay una gran expectativa sobre los resultados que se puedan obtener.

KNN: El algoritmo funciona encontrando los k puntos de datos más cercanos a una nueva instancia y asignando una etiqueta de clasificación o un valor de regresión basado en los valores de las instancias vecinas.
En general, para nuestro caso se intentó utilizar un k=15; sin embargo, con las pruebas que se han hecho, se puede decir que este algoritmo es muy ineficiente. Se han realizado ejecuciones de 10 minutos donde sólo calcula los primero 2 KNN. Quizá termine por desecharse.

Por otra parte, para el entrenamiento no supervisado se ha trabajado con:

K-Means: Utilizado para clustering o agrupamiento de datos. Funciona dividiendo los datos en k grupos o clusters, donde k se define previamente. El algoritmo asigna cada punto de datos al cluster más cercano y luego recalcula el centro de cada cluster.
En particular, para nuestro caso, utilizando una agrupación de 2 clusters, los resultados son los siguientes:

```
Silhouette score: 0.9991702025349769
Davies-Bouldin score: 0.00042299208828107615
Calinski-Harabasz score: 114935.60942978354
Tiempo de ejecución:  395.6469497680664 segundos
```

Silhouette score: Indice que mide cuán similares son los objetos dentro de un grupo y cuán diferentes son de los objetos de otros grupos. El puntaje varía de -1 a 1, donde valores más cercanos a 1 indican que los grupos están bien separados y los objetos dentro de cada grupo están muy juntos

Davies-Bouldin score: Este es otro índice que mide la calidad del clustering, basado en la distancia entre los centroides de los grupos y la distancia entre los objetos dentro de cada grupo. Un puntaje más bajo indica una mejor separación de los grupos.

Calinski-Harabasz score: Este es un índice que mide la relación entre la varianza dentro de cada grupo y la varianza entre los grupos. Cuanto mayor sea el puntaje, mejor será la calidad del clustering.

En general, las métricas indican que el agrupamiento es bueno, pero tenemos un tiempo de ejecución elevado, además resulta imposible interpretar el agrupamiento si tenemos más de 40 características. Como solución se ha intentado usar PCA.

PCA: Utilizado para reducir la dimensionalidad de un conjunto de datos. Funciona proyectando los datos en un espacio de menor dimensión que aún mantiene la mayor cantidad posible de variabilidad de los datos originales.

Para nuestro caso, debido al alto número de características presentes en el dataset, podríamos aplicar PCA para reducir la dimensionalidad e inventar visualizar en 2 dimensiones el agrupamiento de los datos al usar K-Means. Sin embargo, tener presente que el PCA transforma las características en valores no interpretables y que además reducir la dimensión puede generar pérdida de información relevante para el dataset.
Los resultados son los siguientes:

```
Silhouette score: 0.9994565913730302
Davies-Bouldin score: 0.00026746975462098587
Calinski-Harabasz score: 130571.7461380656
(4541,)
Tiempo de ejecución:  1.5586040019989014 segundos
```

Se mejoró el tiempo de ejecución y se están obteniendo resultados similares que al aplicar solamente K-Means.

2. Bibliografía

- Kepler Objects of Interest. https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=koi

- Nasa. “Kepler Exoplanet Search Results.” Kaggle, 10 Oct. 2017. www.kaggle.com/nasa/kepler-exoplanet-search-results.

- “Overview.” NASA, NASA, 2 Apr. 2021. https://exoplanets.nasa.gov/what-is-an-exoplanet/planet-types/overview/

- Malik, A. P. Moster, B. Obermeier C “Exoplanet Detection using Machine Learning.” https://arxiv.org/pdf/2011.14135.pdf

- Clayton, G. Manry, B. Rafiqi, S “Machine Learning Pipeline for Exoplanet Classification” 2019 https://scholar.smu.edu/cgi/viewcontent.cgi?article=1070&context=datasciencereview