Segunda Entrega Proyecto IA “Forest Cover Type Prediction”

Bryan Zuleta Vélez
Daniel Alejandro Yepes Mesa

Facultad de Ingeniería
Universidad de Antioquia

Introducción a la inteligencia artificial para ciencias e ingeniería

Profesor: Raul Ramos Pollan

22 de Agosto de 2022

1. **Preprocesamiento del Dataset.**

Para poder continuar con el dataset, primero que todo vimos que cumpliera las pautas que el profesor había pedido, para sorpresa de nosotros una de las pautas no cumplía ya que nuestro Dataset no tenía datos faltantes, así que ahí se generó nuestro primer problema ya que para ello usamos un método que genere un porcentaje fijo de datos nulos para poder empezar a entrenar nuestro Dataset.

```
#Preparamos el dataset para cumplir con los requisitos del problema

df_forest = df.copy()

from random import randrange

porcentaje = 6

n_filas, n_columnas = df.shape
numero_nans = (n_filas*n_columnas*porcentaje)//100  # // es la división
entera

for i in range(numero_nans):
 fila = randrange(0, n_filas)
 columna = randrange(1, n_columnas-1)
 df_forest.iloc[fila, columna] = float("nan")
```

Uno de los problemas principales aparte de la generación de datos, fue poder volver a llenar dichos datos para el entrenamiento, ya que nuestro Dataset cuenta con 56 columnas, de las cuales 1 es fija porque es el ID, 10 son enteros y 45 son booleanos los cuales solamente entregan un 1 o un 0, así que para ello lo primero que hicimos fue hacer dos FOR uno para rellenar los valores nulos con el promedio de los valores enteros y otro para llenar las columnas que tenían los valores booleanos

```
aux = 0

for column in columns_means:
 df_forest[column].fillna(value=means_vector[aux], inplace=True)
 aux += 1


for column in columns_bool:
```

```
df_forest[column].fillna(value=random.randint(0,1), inplace=True)
```

Aparte nuestro Dataset tenía tantos datos, que las características que posee cada columna eran difíciles de entender y ciertas cosas no generaban sentido para nosotros

2. **Procesamiento de Datos**

En el procesamiento de datos, no tuvimos problemas, aunque hemos hecho pocos modelos y apenas haremos un entrenamiento para mirar como está funcionando ya que la base de datos es un poco grande y nos tomó algo de tiempo adaptarla a lo que nos estaban pidiendo

3. **Análisis de Datos**

```
#En este paso vamos a poder observar toda la información relacionada al dataframe que hemos cargado

df_forest.info()
```

Luego de rellenar todos los valores nulos del dataframe, nos fijamos en la información que nos muestra el dataframe

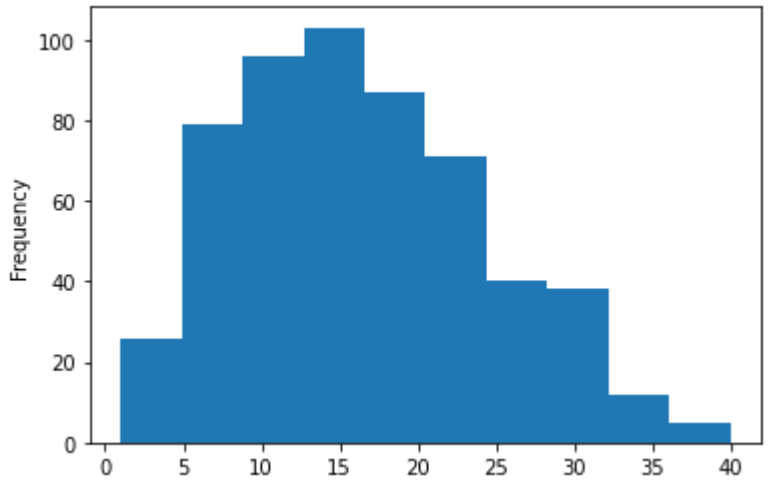
Como se puede observar en la lista ya no hay datos nulos aparte que nos muestran que 54 columnas son tipo flotante y 2 son tipo entero que básicamente son el ID y el Cover_Type, pero haciendo una búsqueda más exhaustiva como mencionamos arriba, de la columna 11 a 55 son tipo booleano ya que solo hay resultados de 0 y 1.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15120 entries, 0 to 15119
Data columns (total 56 columns):
 #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----
 0   Id                                  15120 non-null  int64
 1   Elevation                           15120 non-null  float64
 2   Aspect                              15120 non-null  float64
 3   Slope                               15120 non-null  float64
 4   Horizontal_Distance_To_Hydrology    15120 non-null  float64
 5   Vertical_Distance_To_Hydrology      15120 non-null  float64
 6   Horizontal_Distance_To_Roadways     15120 non-null  float64
 7   Hillshade_9am                       15120 non-null  float64
 8   Hillshade_Noon                      15120 non-null  float64
 9   Hillshade_3pm                       15120 non-null  float64
 10  Horizontal_Distance_To_Fire_Points  15120 non-null  float64
 11  Wilderness_Area1                    15120 non-null  float64
 12  Wilderness_Area2                    15120 non-null  float64
 13  Wilderness_Area3                    15120 non-null  float64
 14  Wilderness_Area4                    15120 non-null  float64
 15  Soil_Type1                          15120 non-null  float64
 16  Soil_Type2                          15120 non-null  float64
 17  Soil_Type3                          15120 non-null  float64
 18  Soil_Type4                          15120 non-null  float64
 19  Soil_Type5                          15120 non-null  float64
 20  Soil_Type6                          15120 non-null  float64
 21  Soil_Type7                          15120 non-null  float64
 22  Soil_Type8                          15120 non-null  float64
 23  Soil_Type9                          15120 non-null  float64
 24  Soil_Type10                         15120 non-null  float64
 25  Soil_Type11                         15120 non-null  float64
 26  Soil_Type12                         15120 non-null  float64
 27  Soil_Type13                         15120 non-null  float64
 28  Soil_Type14                         15120 non-null  float64
```

Aunque no se ve por completo, si se puede notar que todas las columnas ya no poseen datos nulos y aparte muestra la cantidad total de columnas.

También realizamos un pequeño histograma que nos mostraba la frecuencia en la que se mostraban los datos en pequeños grupos de 5.

```
data_forest['Slope'].plot(kind = "hist")
```



y por último hicimos una función que mostraba la relación de todas las columnas con respecto al Cover_Type la cual es esta:

```
abs(df_forest.corr()['Cover_Type'].sort_values(ascending=False))
```

la cual nos mostró lo siguiente:

Cover_Type	1.000000
Soil_Type38	0.252060
Soil_Type39	0.231307
Soil_Type40	0.200967
Soil_Type10	0.126305
Soil_Type35	0.112345
Wilderness_Area3	0.111624
Id	0.108363
Slope	0.083026
Vertical_Distance_To_Hydrology	0.072836
Wilderness_Area4	0.071385
Soil_Type37	0.071210
Soil_Type17	0.041495
Soil_Type13	0.038083
Soil_Type2	0.023756
Soil_Type14	0.021709
Elevation	0.021484
Wilderness_Area2	0.017603
Soil_Type1	0.014254
Soil_Type5	0.012960
Soil_Type15	0.010508
Soil_Type18	0.009682
Soil_Type11	0.007165
Soil_Type6	0.006544
Aspect	0.006392
Soil_Type28	0.002094

en donde se puede notar que el tipo de suelo 38, 39, 40, 10 y 35 son los que más están relacionados con la cobertura que se está buscando, aunque tiene

relación con todos, los mencionados son los que màs tienen fuerza con referencia al cover type.