PYTHON AVANZADO (PYTHON CON LIBRERÍAS DE ANÁLISIS DE DATOS)

Fernanfrain Montoya Arbelaez

CC:1060652348

TALENTO TICS 2023

OBEJTIVOS

- Hacer uso de funcionalidades avanzadas de Python aplicado a tareas básicas de análisis de datos.

- Usar librerías como pandas, matplolib y demás en tareas básicas de carga, limpieza, análisis y visualización de datos.

- Aplicar de forma correcta las capacidades de análisis necesarias para hacer un uso adecuado de las librerías en tareas de análisis de datos.

Se realiza el análisis de datos de los siguientes Dataset los cuales se encuentran en el repositorio de GitHub al igual que el workflow del proyecto en Python.

- Repositorio GitHub: https://github.com/Efrain06/TalentoTICS_2023/tree/799a72d9b0f31475507591996 1ecb41c8544b49f/Proyect_Python2

- Carga del Dataframe

Se utiliza el comando read_csv de la librería de Pandas, el cual carga con éxito el Dataframe mostrando como resultado 7 columnas; las cuales podemos visualizar que en la columnas "Segmento, terminal, tecnología y No. Suscriptores" poseen valores nulos.

```
Proyect_Python2 > proyecto_final_pt1.py > ...
1   # WORKLOW DA:
2
3   import pandas as pd
4   import matplotlib.pyplot as plt
5   import seaborn as sb
6
7   # 1. CARGA DE DATOS ->
8
9   # Se carga el dataframe desde el archivo plano
10  df_Datos1 = pd.read_csv('Proyect_Python2\Dataset\Internet_proveedor.csv')
11  df_Datos1.info()
12
```

```
Data columns (total 4 columns):
PS C:\Users\Hp\Documents\Proyects\TalentoTICS_2023> & C:/Users/Hp/AppData/Loca
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 635 entries, 0 to 634
Data columns (total 7 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   AÑO              635 non-null    int64
 1   TRIMESTRE        635 non-null    int64
 2   PROVEEDOR        635 non-null    object
 3   SEGMENTO         633 non-null    object
 4   TERMINAL         633 non-null    object
 5   TECNOLOGÍA       633 non-null    object
 6   No. SUSCRIPTORES  633 non-null    float64
dtypes: float64(1), int64(2), object(4)
memory usage: 34.9+ KB
```

- Limpieza del Dataframe

Se utiliza el comando dropna de la librería de Pandas, El cual elimina con éxito los valores nulos del Dataframe para continuar con un analisis de datos mas limpio y organziado; verificamos tambien si posee filas duplicadas con el comando “duplicated” y “drop.duplicates”.

```
11  df_Datos1.info()
12
13  # Se eliminan los valores nulos Dataframe:
14  df_Datos1.dropna(inplace=True)
15  df_Datos1.info()
16
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 630 entries, 0 to 634
Data columns (total 7 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   AÑO              630 non-null    int64
 1   TRIMESTRE        630 non-null    int64
 2   PROVEEDOR        630 non-null    object
 3   SEGMENTO         630 non-null    object
 4   TERMINAL         630 non-null    object
 5   TECNOLOGÍA       630 non-null    object
 6   No. SUSCRIPTORES  630 non-null    float64
dtypes: float64(1), int64(2), object(4)
memory usage: 39.4+ KB
```

```
16
17  #mostrar qué filas están duplicadas Dataframe:
18  print(df_Datos1.duplicated())
19
20
21  #eliminar filas duplicadas del dataframe Datasframe:
22  df_Datos1.drop_duplicates(inplace=True)
23
24
```

```
dtypes: float64(1), int64(2), object(4)
memory usage: 39.4+ KB
0      False
1       True
2       True
3      False
4      False
       ...
630     True
631     True
632    False
633    False
634    False
Length: 630, dtype: bool
```

- Tareas de análisis básicas del dataset:

Se seleccionan las columnas que tomaremos como prioridad para iniciar el analisis de datos del Dataframe para lo cual iniciamos con datos estadisticos como el comando “describe” de la librería Pandas el cual nos arroja valores estadisticos del Dataframe ya depurado “df_Dt1_Tb; tambien utilizamos la linea de codigo “factorize” y “corr” para ver la correlacion del Dataframe.

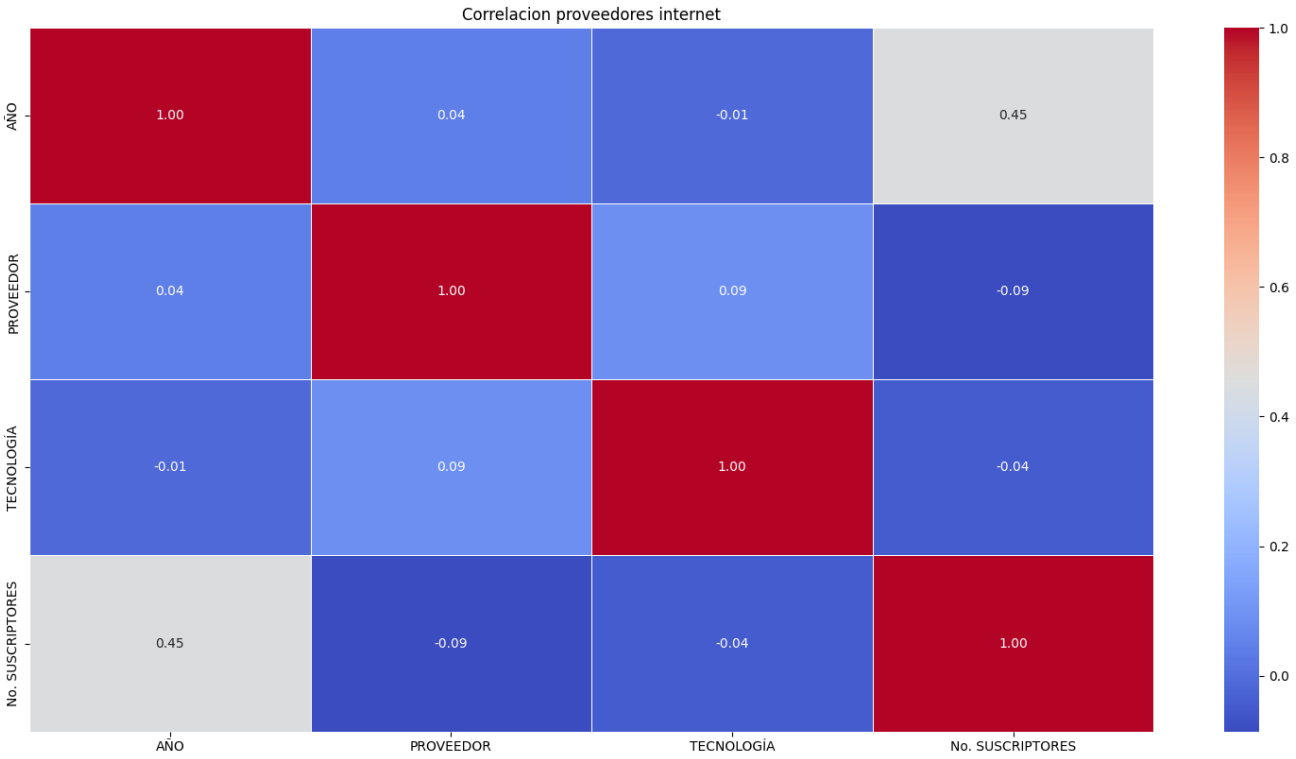
```
24
25  #Se seleccionan las siguientes columnas del dataframe:
26  df_Dt1_Tb = df_Datos1 [["AÑO","PROVEEDOR","TECNOLOGÍA","No. SUSCRIPTORES"]]
27  df_Dt1_Tb.info()
28  print("................................")
29
30  # (a) obtener parámetros estadísticos de columnas
31  print(df_Dt1_Tb.describe())
32  print("................................")
33  print(f"Numero minimo de suscriptores entre el 2020 y el 2023:{df_Dt1_Tb['No. SUSCRIPTORES'].min()}")
34  print(f"Numero maximo de suscriptores entre el 2020 y el 2023:{df_Dt1_Tb['No. SUSCRIPTORES'].max()}")
35  print("................................")
36
37  # 3. Factorizacion (para analisis de correlacion)
38  df_Dt1_Cr = df_Dt1_Tb.apply(lambda x: pd.factorize(x)[0])
39  print(df_Dt1_Cr.corr())
40  print("................................")
41
```

```
Length: 630, dtype: bool
<class 'pandas.core.frame.DataFrame'>
Index: 613 entries, 0 to 634
Data columns (total 4 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   AÑO              613 non-null    int64
 1   PROVEEDOR        613 non-null    object
 2   TECNOLOGÍA       613 non-null    object
 3   No. SUSCRIPTORES  613 non-null    float64
dtypes: float64(1), int64(1), object(2)
memory usage: 23.9+ KB
.................................
                AÑO  No. SUSCRIPTORES
count   613.000000      6.130000e+02
mean   2021.500816      3.243265e+05
std       0.933447      1.029138e+06
min    2020.000000      0.000000e+00
25%    2021.000000      7.620000e+02
50%    2022.000000      1.790300e+04
75%    2022.000000      1.206790e+05
max    2023.000000      7.652954e+06
.................................
Numero minimo de suscriptores entre el 2020 y el 2023:0.0
Numero maximo de suscriptores entre el 2020 y el 2023:7652954.0
.................................
                    AÑO  PROVEEDOR  TECNOLOGÍA  No. SUSCRIPTORES
AÑO            1.000000   0.043890   -0.010992          0.451781
PROVEEDOR      0.043890   1.000000    0.087646         -0.085850
TECNOLOGÍA    -0.010992   0.087646    1.000000         -0.044984
No. SUSCRIPTORES  0.451781  -0.085850   -0.044984          1.000000
.................................
```

Del analisis estadistico podemos ver que los datos suministrados en el Dataframe no cuentan con alta correlacion.

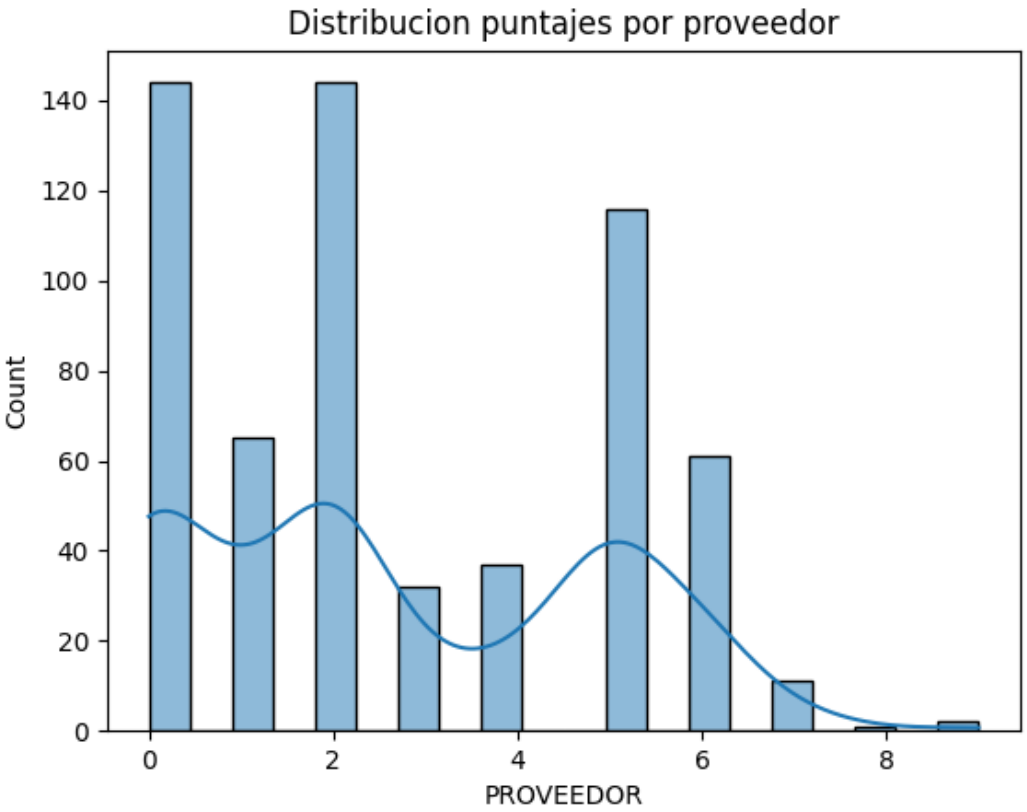
A continuacion se realiza el analisis de datos con diferentes tipos de graficos para tener la informacion mas concisa posible.

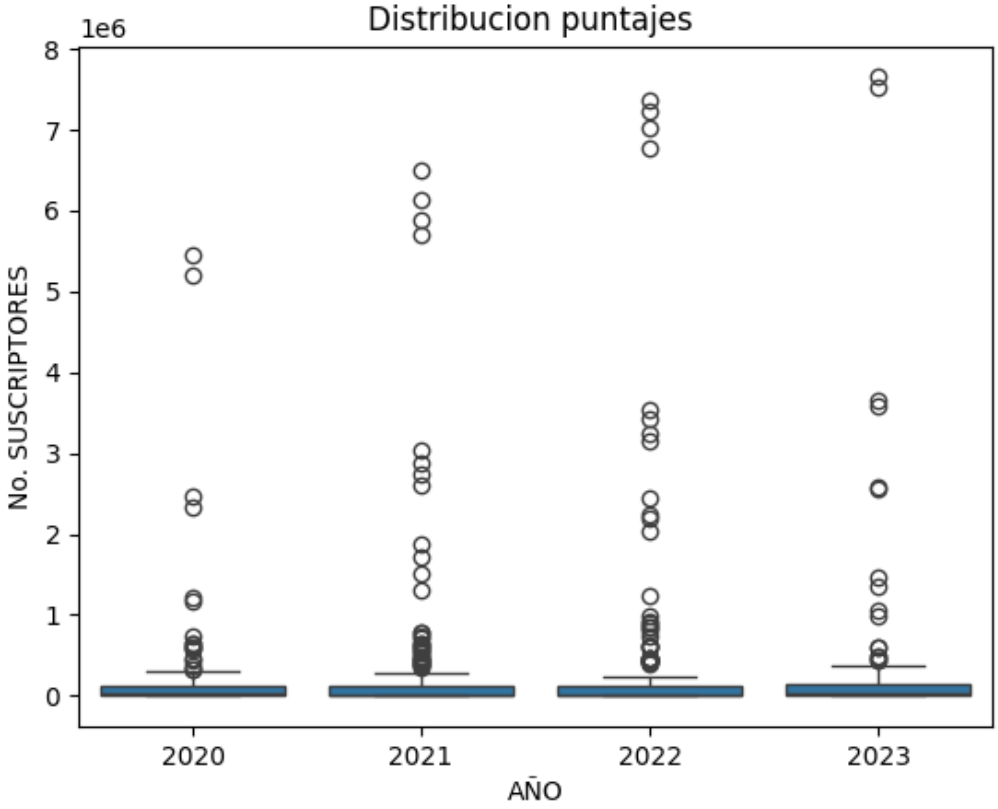
```
42
43  # Se grafican los datos por tabla de calor
44  plt.figure(figsize=(10,10))
45  corr = df_Dt1_Cr.corr()
46  sb.heatmap(corr, annot=True, cmap="coolwarm",fmt=".2f",linewidths=.5) #
47  plt.title("Correlacion proveedores internet" )
48  plt.show()
49
50  print("                                    ")
```

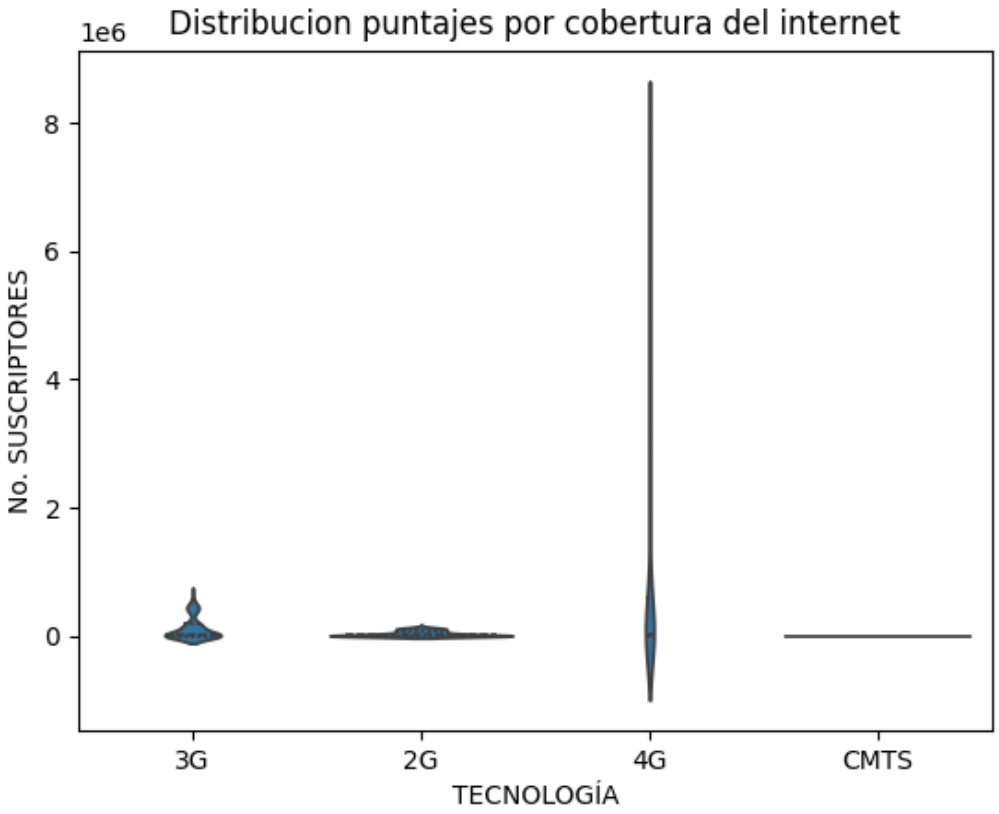


Nota: No posee correlación entre los datos del Dataframe.

```
51
52  #grafico de agrupacion:
53  sb.histplot(df_Dt1_Cr["PROVEEDOR"], bins=20, kde=True)
54  plt.title("Distribucion puntajes por proveedor")
55  plt.show()
56
57  print("................................")
58
59  sb.boxplot(x="AÑO", y="No. SUSCRIPTORES", data=df_Dt1_Tb)
60  plt.title("Distribucion puntajes ")
61  plt.show()
62
63  print("................................")
64
65  sb.violinplot(x="TECNOLOGÍA", y="No. SUSCRIPTORES", data=df_Dt1_Tb, inner="quartile")
66  plt.title("Distribucion puntajes por cobertura del internet")
67  plt.show()
```


Distribucion puntajes por proveedor
Count
PROVEEDOR
0
20
40
60
80
100
120
140
0
2
4
6
8


Distribucion puntajes
1e6
No. SUSCRIPTORES
AÑO
0
1
2
3
4
5
6
7
8
2020
2021
2022
2023



Nota: El internet con mayor cobertura en tecnología es el 4G.

Para el siguiente punto del proyecto “Series de tiempo” seleccionamos los siguientes Dataset los cuales se encuentran en el repositorio y también el workflow.

Se realiza la limpieza del Dataframe eliminando los valores nulos y las filas duplicadas de los dos Dataframe; iniciando el análisis de datos utilizando “merge”, “to_numeric” de la librería de Pandas.

```python
import pandas as pd
import matplotlib.pyplot as plt


# 1. CARGA DE DATOS ->

# Se carga el dataframe desde el archivo plano "Perfil de morbilidad Julio"
df_dat_PM1 = pd.read_csv('Proyect_data\Dataset\Perfil_de_morbilidad_Julio.csv')
df_dat_PM1.info()

# Se carga el dataframe desde el archivo plano "Perfil de morbilidad Agosto"
df_dat_PM2 = pd.read_csv('Proyect_data\Dataset\Perfil_de_morbilidad_Agosto.csv')
df_dat_PM2.info()

# Se eliminan los valores nulos Dataset 1 y 2:
df_dat_PM1.dropna(inplace=True)
df_dat_PM2.dropna(inplace=True)

#eliminar filas duplicadas del dataframe Dataset 1 y 2:
df_dat_PM1.drop_duplicates(inplace=True)
df_dat_PM2.drop_duplicates(inplace=True)

# Serie de tiempo
df_all_data = pd.merge(df_dat_PM1, df_dat_PM2, on="AÑO REPORTADO", how="inner", copy=False )
print(df_all_data.head())
print("..........................................................................")
print(df_all_data.info())
print("..........................................................................")
print(df_all_data.tail())
print("..........................................................................")

#adecuación del dataset para manejo de datos temporales:
df_Dat_Tmp = df_dat_PM2.rename(columns={"EDAD DE ATENCION (AÑOS)":"EDAD DE ATENCION"})
#valores inválidos con opción errors = "coerce", asigna NaN:
df_Dat_Tmp["EDAD DE ATENCION"] = pd.to_numeric(df_Dat_Tmp["EDAD DE ATENCION"],errors="coerce")

print(f"dataset limpio:{df_Dat_Tmp.info()}")
print("..........................................................................")
print(df_dat_PM2.head(50))
print("..........................................................................")
print(f"La edad maxima de atencion es: ", df_Dat_Tmp["EDAD DE ATENCION"].max())
print("..........................................................................")
print(f"La edad minima de atencion es: ", df_Dat_Tmp["EDAD DE ATENCION"].min())

```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 239 entries, 0 to 238
Data columns (total 5 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   CÓDIGO CIE-10                   220 non-null    object
 1   NOMBRE DEL DIAGNOSTICO          239 non-null    object
 2   UNIDAD FUNCIONAL + GRUPO ETAREO  239 non-null    object
 3   TOTAL                           239 non-null    int64
 4   AÑO REPORTADO                   239 non-null    int64
dtypes: int64(2), object(3)
memory usage: 9.5+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 308548 entries, 0 to 308547
Data columns (total 6 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   CIDIGO CIE-10           308548 non-null  object
 1   NOMBRE DEL DIAGNOSTICO  308548 non-null  object
 2   UNIDAD FUNCIONAL        308548 non-null  object
 3   DESTINO AL EGRESO       308548 non-null  object
 4   EDAD DE ATENCION (AÑOS)  308548 non-null  int64
 5   AÑO REPORTADO           308548 non-null  int64
dtypes: int64(2), object(4)
memory usage: 14.1+ MB
```

Nota: Se visualiza la informacion de cada uno de los Dataframe

```
memory usage: 14.1+ MB
  CÓDIGO CIE-10         NOMBRE DEL DIAGNOSTICO_x UNIDAD FUNCIONAL + GRUPO ETAREO  TOTAL  AÑO REPORTADO CIDIGO CIE-10                    NOMBRE DEL DIAGNOSTICO_y                 UNIDAD FUNCIONAL DESTINO AL EGRESO  EDAD DE ATENCION (AÑOS)
0         I10X  HIPERTENSION ESENCIAL (PRIMARIA)             CONSULTA EXTERNA  15627         2018        N459  ORQUITIS, EPIDIDIMITIS Y ORQUIEPIDIDIMITIS SIN...  ATENCION INICIAL DE URGENCIAS ADULTOS          SALIDA                    16
1         I10X  HIPERTENSION ESENCIAL (PRIMARIA)             CONSULTA EXTERNA  15627         2018        A90X               FIEBRE DEL DENGUE [DENGUE CLASICO]          OBSERVACION ADULTO URGENCIAS          SALIDA                    17
2         I10X  HIPERTENSION ESENCIAL (PRIMARIA)             CONSULTA EXTERNA  15627         2018        A419                    SEPTICEMIA, NO ESPECIFICADA               HOSPITALIZACION CIRUGIA          SALIDA                    15
3         I10X  HIPERTENSION ESENCIAL (PRIMARIA)             CONSULTA EXTERNA  15627         2018        R11X                                NAUSEA Y VOMITO          OBSERVACION ADULTO URGENCIAS          SALIDA                    15
4         I10X  HIPERTENSION ESENCIAL (PRIMARIA)             CONSULTA EXTERNA  15627         2018        R103  DOLOR LOCALIZADO EN OTRAS PARTES INFERIORES DE...          OBSERVACION ADULTO URGENCIAS       REFERENCIA                    18
.....
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9413360 entries, 0 to 9413359
Data columns (total 10 columns):
 #   Column                          Dtype
---  ------                          -----
 0   CÓDIGO CIE-10                   object
 1   NOMBRE DEL DIAGNOSTICO_x        object
 2   UNIDAD FUNCIONAL + GRUPO ETAREO  object
 3   TOTAL                           int64
 4   AÑO REPORTADO                   int64
 5   CIDIGO CIE-10                   object
 6   NOMBRE DEL DIAGNOSTICO_y        object
 7   UNIDAD FUNCIONAL                object
 8   DESTINO AL EGRESO               object
 9   EDAD DE ATENCION (AÑOS)         int64
dtypes: int64(3), object(7)
memory usage: 718.2+ MB
None
.....
```

Nota: Podemos visualizar con el método head las primeras 5 filas del Dataframe (“df_all_data”), resultado de la unión de los dos Dataframe con el método “merger”; también visualizamos la información de Dataframe unido.

```
None
.....................................................................
       CÓDIGO CIE-10                NOMBRE DEL DIAGNOSTICO_x    UNIDAD FUNCIONAL + GRUPO ETAREO  TOTAL  AÑO REPORTADO CIDIGO CIE-10                  NOMBRE DEL DIAGNOSTICO_y  UNIDAD FUNCIONAL DESTINO AL EGRESO  EDAD DE ATENCION (AÑOS)
9413355         N189  INSUFICIENCIA RENAL CRONICA NO ESPECIFICADA  URGENCIAS MAYORES DE 65 AÑOS Y MAS     23          2018         A499          INFECCION BACTERIANA, NO ESPECIFICADA  CONSULTA EXTERNA           SALIDA                     19
9413356         N189  INSUFICIENCIA RENAL CRONICA NO ESPECIFICADA  URGENCIAS MAYORES DE 65 AÑOS Y MAS     23          2018         T783                    EDEMA ANGIONEUROTICO  CONSULTA EXTERNA           SALIDA                     19
9413357         N189  INSUFICIENCIA RENAL CRONICA NO ESPECIFICADA  URGENCIAS MAYORES DE 65 AÑOS Y MAS     23          2018         I370          ESTENOSIS DE LA VALVULA PULMONAR  CONSULTA EXTERNA           SALIDA                     19
9413358         N189  INSUFICIENCIA RENAL CRONICA NO ESPECIFICADA  URGENCIAS MAYORES DE 65 AÑOS Y MAS     23          2018         I341            PROLAPSO (DE LA VALVULA) MITRAL  CONSULTA EXTERNA           SALIDA                     19
9413359         N189  INSUFICIENCIA RENAL CRONICA NO ESPECIFICADA  URGENCIAS MAYORES DE 65 AÑOS Y MAS     23          2018         S526  FRACTURA DE LA EPIFISIS INFERIOR DEL CUBITO Y ...  CONSULTA EXTERNA           SALIDA                     36
.....................................................................
<class 'pandas.core.frame.DataFrame'>
Index: 161244 entries, 0 to 308547
Data columns (total 6 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   CIDIGO CIE-10        161244 non-null  object
 1   NOMBRE DEL DIAGNOSTICO  161244 non-null  object
 2   UNIDAD FUNCIONAL     161244 non-null  object
 3   DESTINO AL EGRESO    161244 non-null  object
 4   EDAD DE ATENCION     161244 non-null  int64
 5   AÑO REPORTADO        161244 non-null  int64
dtypes: int64(2), object(4)
memory usage: 8.6+ MB
dataset limpio:None
```

Nota: Con el metodo “tail”, visualizamos las 5 ultimas filas del Dataframe; tambien visualizamos la informacion del Dataframe con la modificacion del nombre de la columna “Año reportado”.

```
.....................................................................
  CIDIGO CIE-10                          NOMBRE DEL DIAGNOSTICO                     UNIDAD FUNCIONAL       DESTINO AL EGRESO  EDAD DE ATENCION (AÑOS)  AÑO REPORTADO
0          A000  COLERA DEBIDO A VIBRIO CHOLERAE 01, BIOTIPO CH...                  CONSULTA EXTERNA                  SALIDA                     39.0           2017
1          A000  COLERA DEBIDO A VIBRIO CHOLERAE 01, BIOTIPO CH...                  CONSULTA EXTERNA                  SALIDA                     44.0           2017
2          A010                                  FIEBRE TIFOIDEA                  CONSULTA EXTERNA                  SALIDA                     28.0           2017
3          A010                                  FIEBRE TIFOIDEA                  CONSULTA EXTERNA                  SALIDA                     76.0           2017
4          A010                                  FIEBRE TIFOIDEA                  CONSULTA EXTERNA                  SALIDA                     50.0           2017
5          A021                    SEPTICEMIA DEBIDA A SALMONELLA                  CONSULTA EXTERNA                  SALIDA                     30.0           2017
6          A021                    SEPTICEMIA DEBIDA A SALMONELLA          HOSPITALIZACION SEPTIMO PISO              SALIDA                     27.0           2017
7          A021                    SEPTICEMIA DEBIDA A SALMONELLA  HOSPITALIZACION INFECTOLOGIA PEDIATRICA           SALIDA                      0.0           2017
8          A022         INFECCIONES LOCALIZADAS DEBIDA A SALMONELLA                  CONSULTA EXTERNA                  SALIDA                     40.0           2017
9          A028  OTRAS INFECCIONES ESPECIFICADAS COMO DEBIDAS A...       HOSPITALIZACION MEDICINA INTERNA  HOSPITALIZACION EN CASA     42.0           2017
.....................................................................
La edad maxima de atencion es:  106.0
.....................................................................
La edad minima de atencion es:  0.0
```

Nota: Con el metodo “to_numeric” convierte los valores de “edad de atenion” en numeros enteros.

CONCLUSIONES

Python es uno de los lenguajes mas utilizados hoy en día en la analítica de datos, también es utilizado en compañía de lenguajes funcionales que permiten una mejor comprensión y visualización de los datos por la variedad de librerías que este maneja.

Este curso de Python 2 enfocado a la analítica de datos es un gran paso al campo laboral que este abarca, hoy en día el análisis los podemos analizar mediante gráficos, tablas, esquemas, con tal de conocer las necesidades por la que estamos realizando el análisis a un Dataset