Análisis y visualización de datos

Actividad 2 Marce Martinez.

Situación

Una vez que se entregó la primera parte del informe, **continua con la segunda parte, enfocada en los aspectos demográficos y sociales.** Los datos para esta sección no están completamente listos para analizar, necesitan preprocesamiento antes de poder construir los gráficos o tablas. Como sabe, **algunas de estas tareas son más prácticas y eficientes haciéndolas en Python** que manualmente (en Excel, por ejemplo).

Requerimientos

Para realizar esta actividad, **descargue los archivos consignados al inicio de la actividad.**

Importante: Al momento de importar los archivos .csv, usando pd.read_csv(), usar la opción encoding = "latin-1 " para que i**mporte correctamente los nombres de las provincias con tildes.**

Consignas

Como próximo paso, decide integrar distintas bases de datos que tiene disponibles. Estas contienen variables demográficas: población, hogares y viviendas, esperanza de vida y fecundidad.

1. En Jupyter, crear un nuevo notebook e importar las librerías necesarias, y luego la base de proyecciones de población por año **(en formato .csv)** y las o**tras (ej. expectativa de vida, fecundidad).** Tener en cuenta que **algunas bases contienen datos de varios años y otras tienen únicamente el año del censo 2010.** Hacer los chequeos básicos (head, describe, etc.).

2. Calcular un campo nuevo, **densidad (población/superficie)** y **usar la función descrita sobre ese campo nuevo.**

- **Informe se encuentra comentado dentro del codigo de notebook, en forma breve y clara.**

Firefox File Edit View History Bookmarks Tools Window Help — Wed 22 May 22:01

localhost:8889/notebooks/Analisis_Visualizacion_Onmex/API2 Analisis_Visualizacion_Marce.ipynb

UPDATE Read the migration plan to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions. Don't show anymore

jupyter API2 Analisis_Visualizacion_Marce Last Checkpoint: 2 hours ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

In [74]:
```
#Este notebook ha sido generado mediante la instalaciøn de la distribuciøn de anaconda y Jupyter notebook
```

In [75]:
```
#CONSIGNAS
```

In [76]:
```
#Se cargan las librerias de pandas y numpy

import pandas as pd
import numpy as np
```

In [77]:
```
#Importaciøn de os para acceder a los archivos cargados
import os
```

In [78]:
```
#se carga un dataframe con la base de poblacion por a;o
df=pd.read_csv("poblacion(2).csv",encoding = "latin-1 ")
```

In [79]:
```
#visualizar las primeras filas del contenido del dataframe de Poblaciøn
#df.head(20)
print(df.head(10)) #Aqui se muestran los datos un poco mas crudos
```

```
    provincia  anio  poblacion_total  poblacion_varones  poblacion_mujeres
0  Total País  2010         40788453           19940704           20847749
1  Total País  2011         41261490           20180791           21080699
2  Total País  2012         41733271           20420391           21312880
3  Total País  2013         42202935           20659037           21543898
4  Total País  2014         42669500           20896203           21773297
5  Total País  2015         43131966           21131346           22000620
6  Total País  2016         43590368           21364470           22225898
7  Total País  2017         44044811           21595623           22449188
8  Total País  2018         44494502           21824372           22670130
9  Total País  2019         44938712           22050332           22888380
```

In [80]:
```
#chequeo basico de descripciøn de metricas
df.describe()
```

Out[80]:

	anio	poblacion_total	poblacion_varones	poblacion_mujeres
count	775.000000	7.750000e+02	7.750000e+02	7.750000e+02
mean	2025.000000	3.777746e+06	1.856888e+06	1.920858e+06
std	8.950048	9.560571e+06	4.699604e+06	4.861043e+06
min	2010.000000	1.316610e+05	6.723500e+04	6.442600e+04
25%	2017.000000	5.845510e+05	2.906740e+05	2.934905e+05
50%	2025.000000	1.017731e+06	5.061010e+05	5.161370e+05
75%	2033.000000	1.855285e+06	9.138865e+05	9.404745e+05
max	2040.000000	5.277848e+07	2.603809e+07	2.674038e+07

In [81]:
```
#Calcular un campo nuevo, densidad (población/superficie) y usar la función descrita sobre ese campo nuevo.
#df= dataframe superficie
dfs = pd.read_csv("hogares_viviendas_superficie(2).csv",encoding="latin-1")
dfs.head()
```

Out[81]:

	provincia_id	provincia	hogares	viviendas_particulares	viviendas_particulares_habitadas	superficie_km2
0	2	Capital Federal	1150134	1423973	1082998	200
1	6	Buenos Aires	4789484	5377786	4425193	307571
2	10	Catamarca	96001	113634	89376	102602
3	14	Córdoba	1031843	1232211	978553	165321
4	18	Corrientes	267797	292644	248844	88199

In [82]:
```
print(dfs.head())
```

```
   provincia_id        provincia  hogares  viviendas_particulares  \
0             2  Capital Federal  1150134                 1423973
1             6     Buenos Aires  4789484                 5377786
2            10        Catamarca    96001                  113634
3            14          Córdoba  1031843                 1232211
4            18       Corrientes   267797                  292644

   viviendas_particulares_habitadas  superficie_km2
0                           1082998             200
1                           4425193          307571
2                             89376          102602
3                            978553          165321
4                            248844           88199
```

In [83]:
```
df1= (df[(df['provincia'] == 'Capital Federal') & (df['anio']==2010) & (df['poblacion_total'] > 0 ) ])
```

```
In [84]: df1
```

Out[84]:

	provincia	anio	poblacion_total	poblacion_varones	poblacion_mujeres
31	Capital Federal	2010	3028481	1405566	1622915

```
In [85]: pt=(df1['poblacion_total'])

In [86]: print(pt)

31    3028481
Name: poblacion_total, dtype: int64

In [87]: dfs2=dfs[['provincia','superficie_km2']]

In [88]: print(dfs2)
```

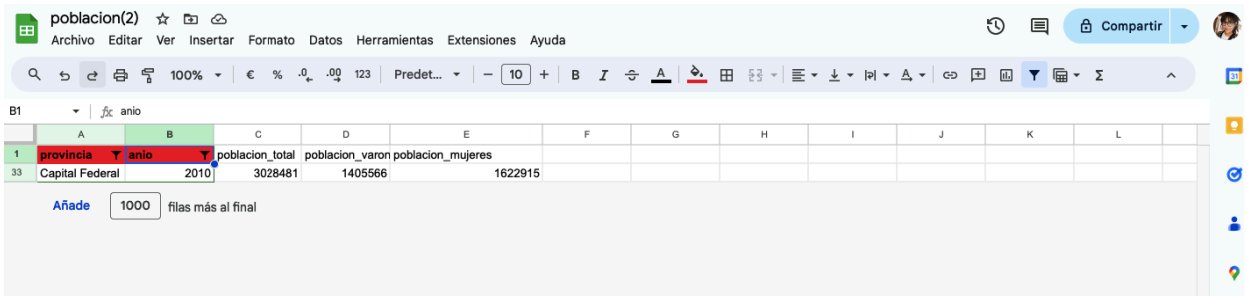
```
              provincia  superficie_km2
0       Capital Federal             200
1          Buenos Aires          307571
2             Catamarca          102602
3               Córdoba          165321
4             Corrientes           88199
5                  Chaco           99633
6                 Chubut          224686
7             Entre Ríos           78781
8                Formosa           72066
9                  Jujuy           53219
10             La Pampa          143440
11             La Rioja           89680
12              Mendoza          148827
13             Misiones           29801
14              Neuquén           94078
15            Río Negro          203013
16                Salta          155488
17            San Juan           89651
18            San Luis           76748
19           Santa Cruz          243943
20             Santa Fe          133007
21  Santiago del Estero          136351
22             Tucumán           22524
23     Tierra del Fuego         1002445
```

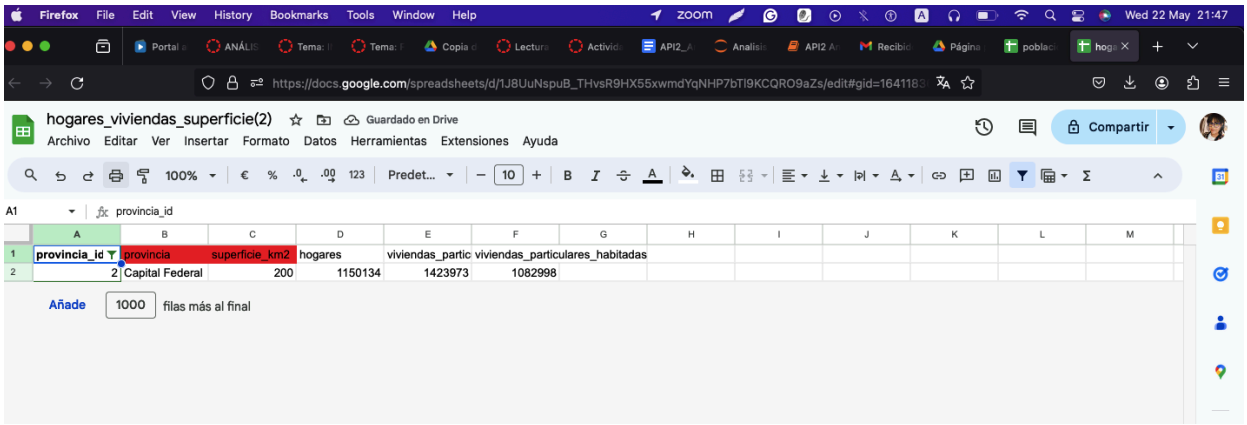
```
In [89]: dfs1=(dfs[(dfs['provincia'] == 'Capital Federal') & (dfs['superficie_km2'] > 0 ) ] )

In [90]: print(dfs1)
```

```
   provincia_id        provincia  hogares  viviendas_particulares  \
0             2  Capital Federal  1150134                 1423973

   viviendas_particulares_habitadas  superficie_km2
0                           1082998             200
```

```
In [91]: sup=(dfs1['superficie_km2'])

In [92]: print(sup)

0    200
Name: superficie_km2, dtype: int64

In [93]: print(pt)

31    3028481
Name: poblacion_total, dtype: int64

In [94]: #Generar un nuevo dataframe para contener el resultado de la densidad de poblacion de capital federal
densidad= pt/200

In [95]: #Densidad de capital federal es la siguiente:
print(densidad)

31    15142.405
Name: poblacion_total, dtype: float64
```

- La validacion de resultados obtenidos entre Jupyter notebook y excel confirma resultados proporcionados en el codigo.



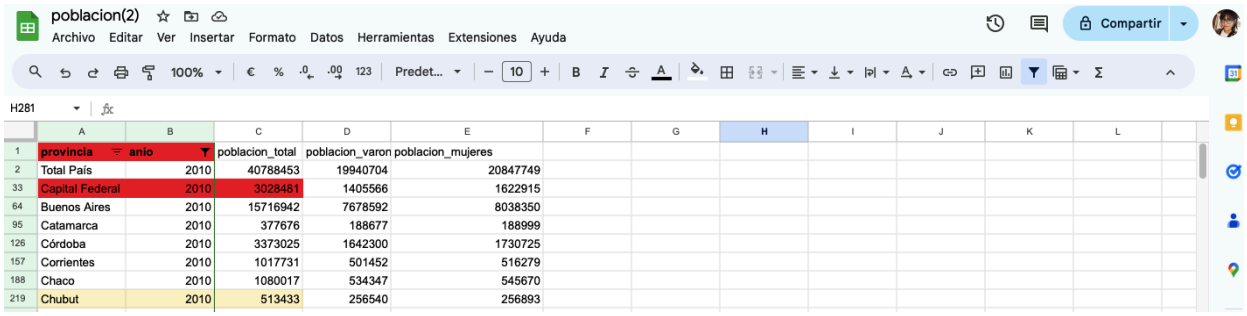


3. Identificar si existe algún valor extremo en la densidad de población y explicar a qué podría deberse esto.

- Se encontraron valores extremos en densidad de poblacion al momento de comparar diferentes provincias con su superficie correspondiente.(muestra tomada: Capital federal vs Chubut)

- Detalles:
  No existe linealidad en la densidad asociada a todas y cada una de las superficies o provincias proporcionadas entre el archivo de poblacion y archivo de hogares_viviendas_superficie, debido a que en algunas situaciones la superficie no es directamente proporcional al incremento o decremento de Poblacion si lo comparamos cada una de las provincias con la base tomada para este ejemplo(Capital Federal) y la Superficie correspondiente(200km2).

- Evidencia del analisis y comparacion de poblaciones totales y superficies de otras provincias.

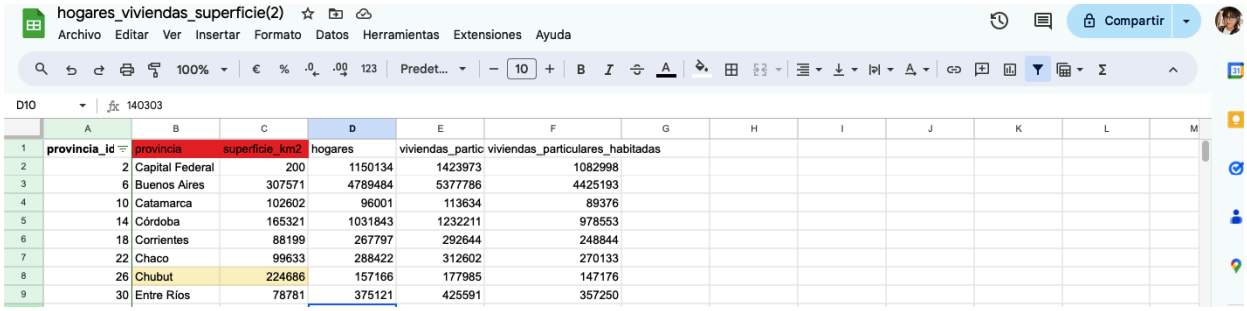

poblacion(2)

Archivo Editar Ver Insertar Formato Datos Herramientas Extensiones Ayuda

Compartir

H281

	A	B	C	D	E
1	provincia	anio	poblacion_total	poblacion_varon	poblacion_mujeres
2	Total País	2010	40788453	19940704	20847749
33	Capital Federal	2010	3028481	1405566	1622915
64	Buenos Aires	2010	15716942	7678592	8038350
95	Catamarca	2010	377676	188677	188999
126	Córdoba	2010	3373025	1642300	1730725
157	Corrientes	2010	1017731	501452	516279
188	Chaco	2010	1080017	534347	545670
219	Chubut	2010	513433	256540	256893


hogares_viviendas_superficie(2)

Archivo Editar Ver Insertar Formato Datos Herramientas Extensiones Ayuda

Compartir

D10 140303

	A	B	C	D	E	F
1	provincia_id	provincia	superficie_km2	hogares	viviendas_partic	viviendas_particulares_habitadas
2	2	Capital Federal	200	1150134	1423973	1082998
3	6	Buenos Aires	307571	4789484	5377786	4425193
4	10	Catamarca	102602	96001	113634	89376
5	14	Córdoba	165321	1031843	1232211	978553
6	18	Corrientes	88199	267797	292644	248844
7	22	Chaco	99633	288422	312602	270133
8	26	Chubut	224686	157166	177985	147176
9	30	Entre Ríos	78781	375121	425591	357250

Gracias!