

Taller 7

Métodos Computacionales para Políticas Públicas - UROSARIO

Entrega: viernes 23-oct-2020 11:59 PM

[Ivonne Paola Ubaque Galán]

[ivonne.ubaque@urosario.edu.co (<mailto:ivonne.ubaque@urosario.edu.co>)]

Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador, idealmente en una carpeta destinada al material del curso.
- Modifique el nombre del archivo del *notebook*, agregando al final un guión inferior y su nombre y apellido, separados estos últimos por otro guión inferior. Por ejemplo, mi *notebook* se llamaría: mcpp_taller7_santiago_mataallana
- Marque el *notebook* con su nombre y e-mail en el bloque verde arriba. Reemplace el texto "[Su nombre acá]" con su nombre y apellido. Similar para su e-mail.
- Desarrolle la totalidad del taller sobre este *notebook*, insertando las celdas que sea necesario debajo de cada pregunta. Haga buen uso de las celdas para código y de las celdas tipo *markdown* según el caso.
- Recuerde salvar periódicamente sus avances.
- Cuando termine el taller:
 1. Descárguelo en PDF. Si tiene algún problema con la conversión, descárguelo en HTML.
 2. Suba todos los archivos a su repositorio en GitHub, en una carpeta destinada exclusivamente para este taller, antes de la fecha y hora límites.

(Todos los ejercicios tienen el mismo valor.)

En este taller exploraremos los datos de crimen de Chicago.

Descargue los datos de crimen del Chicago Data Portal solo para el año 2015

(<https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2/data>
(<https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2/data>)).

Calcule el número de crímenes en cada Community Area en 2015. Haga un gráfico de barras que lo ilustre.

```
In [1]: 1 import pandas as pd
2 import seaborn as sb
3 import matplotlib.pyplot as plt
4 import matplotlib as mpl
5
6 ## se importan las bibliotecas necesarias para el desarrollo del ejercicio
```

```
In [2]: 1 file = "Crimes_Chicago-2015.csv"
2 crimes = pd.read_csv(file)
3 crimes.head(10)
```

	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest
0	11861892	JC474163	01/01/2015 12:00:00 AM	021XX E 84TH ST	0266	CRIM SEXUAL ASSAULT	PREDATORY	RESIDENCE	F
1	9993839	HY183887	01/01/2015 12:00:00 AM	068XX S PAULINA ST	1541	OBSCENITY	SALE/DIST OBSCENE MAT TO MINOR	RESIDENCE	F
2	11028448	JA360336	01/01/2015 12:00:00 AM	051XX W HURON ST	0281	CRIM SEXUAL ASSAULT	NON- AGGRAVATED	APARTMENT	
3	11116187	JA463360	01/01/2015 12:00:00 AM	033XX W CRYSTAL ST	1752	OFFENSE INVOLVING CHILDREN	AGG CRIM SEX ABUSE FAM MEMBER	RESIDENCE	F
4	11800937	JC400566	01/01/2015 12:00:00 AM	110XX S INDIANA AVE	1130	DECEPTIVE PRACTICE	FRAUD OR CONFIDENCE GAME	RESIDENCE	F

```
In [3]: 1 len(crimes)
2 ## establecer la cantidad de observaciones de delictividad en el 2015. Se en
```

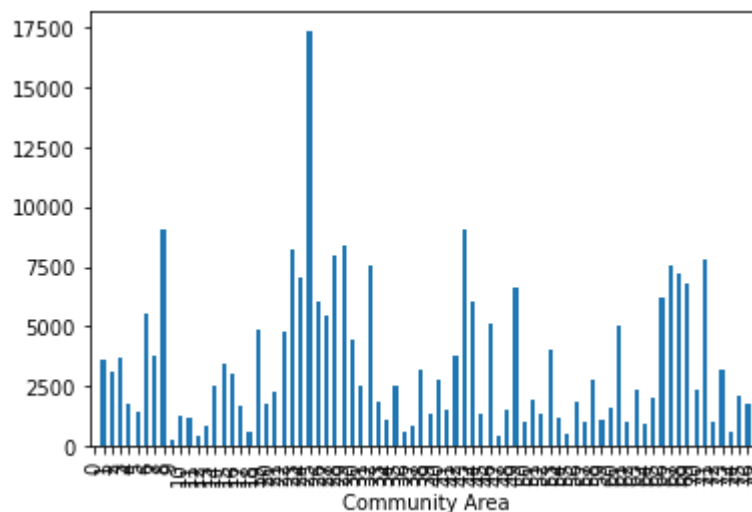
Out[3]: 263652

```
In [4]: 1 ## Se determina con count el número de veces de ocurrencia de los crímenes p
2 ## Por ejemplo, en la número 0 se han registrado durante 2015 2 delitos, en
3 crimes_by_community = crimes.groupby('Community Area')
4
5 crimes_by_community.groups
6
7 community_crime_count = crimes_by_community['ID'].agg('count')
8 community_crime_count.to_frame()
```

Out[4]:

Community Area	ID
0	2
1	3584
2	3109
3	3654
4	1766
...	...
73	3159
74	618
75	2084
76	1746

```
In [5]: 1 ## Se efectua la gráfica de barras con ocurrencia de delitos.
2 community_crime_count.plot(kind='bar');
```



2.

Ordene las Community Areas de acuerdo con el número de crímenes. ¿Qué Community Area (por nombre, idealmente) presenta el mayor número de crímenes? ¿El menor?

```
In [6]: 1 ## Se hace uso del método sort (booleano), que devuelve la serie de delictiv
2 ## con el parámetro ascending de True el orden es creciente.
3
4 table = community_crime_count.to_frame()
5
6 by_intensidad = table.sort_values('ID' , ascending =True)
7 by_intensidad.head(10)
```

Out[6]:

	ID
Community Area	
0	2
9	258
47	389
12	449
55	515
18	584
74	618
36	629
37	840
13	842

```
In [7]: 1 ## Y con el parámetro False, el orden es decreciente.
2 by_intensidad = table.sort_values('ID' , ascending =False)
3 by_intensidad.head(10)
```

Out[7]:

	ID
Community Area	
25	17367
43	9077
8	9051
29	8407
23	8194
28	7930
71	7834
32	7574
67	7530
68	7234

De esta forma se tiene que las community áreas con mayor y menor intensidad de delictividad son Austin (área número25) con 17367

delitos y Edison Park (número 9) con 258 delitos, respectivamente.

3.

Cree una tabla cuyas filas sean días del año (yyyy-mm-dd) y las columnas las 77 Community Areas. En cada campo de la tabla deberá haber el correspondiente número de crímenes. Seleccione algunas Community Areas que le llamen la atención y haga un gráfico de serie de tiempo.

Pista: El siguiente código puede serle útil.

```
In [8]: 1 import pandas as pd
        2 import seaborn as sb
        3 import matplotlib.pyplot as plt
        4 import matplotlib as mpl
        5
        6 file = "Crimes_Chicago-2015.csv"
        7 crimes = pd.read_csv(file, error_bad_lines=False, sep=',',)
        8 crimes.head()
        9 crimes.shape
```

Out[8]: (263652, 22)

```
In [9]: 1 import datetime
        2 def to_day(timestamp):
        3     timestamp = timestamp.split(" ")[0]
        4     timestamp = datetime.datetime.strptime(timestamp, "%m/%d/%Y")
        5     return timestamp.replace(minute=0, hour=0, second=0)
```

```
In [10]: 1 crimes["Day"] = crimes["Date"].apply(to_day)
2 crimes_by_community = crimes.groupby(['Community Area', 'Day'])
3 crimes_by_community = crimes_by_community
4 crimes_by_community.groups
5 community_crime_count = crimes_by_community['ID'].agg('count')
6 table = community_crime_count.to_frame()
7
8 by_year = table.sort_values('ID' , ascending =False)
9 by_year.head(10)
```

Out[10]:

		ID
Community Area	Day	
25	2015-01-01	94
	2015-11-01	84
	2015-06-01	83
	2015-07-05	74
	2015-07-27	73
	2015-08-16	72
32	2015-08-01	71
6	2015-06-28	71
25	2015-04-12	69
	2015-03-20	66

4.

Descargue la base de datos de información socioeconómica (<https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2>) (<https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2>)).

Cree una tabla que agregue el número de crímenes por Community Area. Una esa tabla con la de datos socioeconómicos y cree un "scatter plot" de número de crímenes vs ingreso per cápita. Explique la relación en palabras.

```
In [11]: 1 file = "Crimes_Chicago-2015.csv"
2 crimes = pd.read_csv(file)
3 crimes .head()
4
5 file = "Socioeconomic_Chicago.csv"
6 SocioEco = pd.read_csv(file)
7 SocioEco.head(10)
8
9 crimes_by_community = crimes.groupby('Community Area')
10 crimes_by_community.groups
11
12 community_crime_count = crimes_by_community['ID'].agg('count')
13 community_crime_count.to_frame()
```

Out[11]:

	ID
Community Area	
0	2
1	3584
2	3109
3	3654
4	1766
...	...
73	3159
74	618
75	2084
76	1746
77	2257

78 rows × 1 columns