CC - AI

Programación para la Computación Científica - IA

Visualización con Matplotlib, Pandas y Seaborn

Universidad Sergio Arboleda
Prof. John Corredor

Agenda

- Concatenación de DataFrames desde Hojas de Cálculo
- Visualizando Data con MatplotLib
- Graficas Básicas de Pandas
- Diferencias entre Biblioteca Seaborn y Pandas

Ejercicio: Comparación de Índice de Aprobación de Presidentes de USA.
El Proyecto de la Presidencia Americana de la Universidad de California, Santa Bárbara, proporciona un índice de aprobación agregado hasta un único punto de datos cada día.

https://www.presidency.ucsb.edu/statistics/data/presidential-job-approval

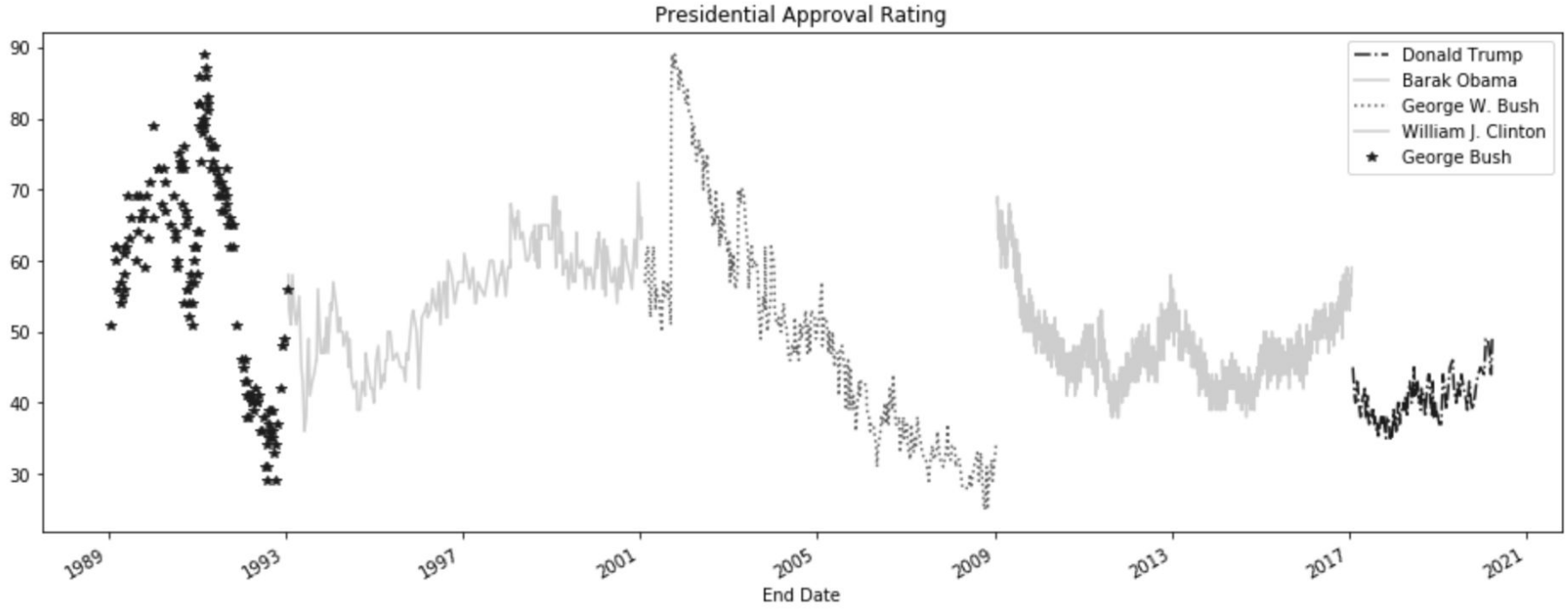
```
import pandas as pd
from pandas import ExcelWriter
from pandas import ExcelFile

trump = pd.read_excel('POTUS.xlsx', sheet_name='Donald Trump')
trump.head()
```

	President	Start Date	End Date	Approving	Disapproving	Unsure/NoData
0	Donald Trump	2020-03-13	2020-03-22	49	45	6
1	Donald Trump	2020-03-02	2020-03-13	44	52	4
2	Donald Trump	2020-02-17	2020-02-28	47	51	2
3	Donald Trump	2020-02-03	2020-02-16	49	48	3
4	Donald Trump	2020-01-16	2020-01-29	49	50	1

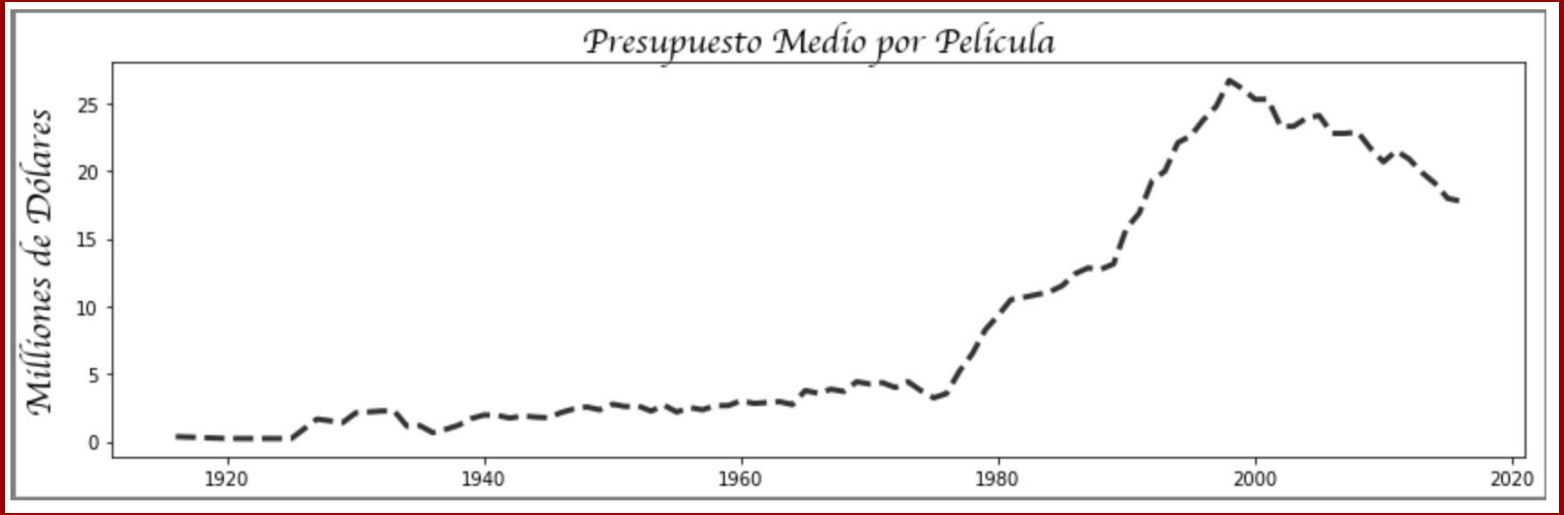
```
presidentes = [trump, obamab,georgew,william,georgeb]
fivePresidents = pd.concat(presidentes)
fivePresidents.groupby('President').head(3)
```

	President	Start Date	End Date	Approving	Disapproving	Unsure/NoData
0	Donald Trump	2020-03-13	2020-03-22	49	45	6
1	Donald Trump	2020-03-02	2020-03-13	44	52	4
2	Donald Trump	2020-02-17	2020-02-28	47	51	2
0	Barak Obama	2017-01-17	2017-01-19	59	37	4
1	Barak Obama	2017-01-15	2017-01-18	58	38	4
2	Barak Obama	2017-01-14	2017-01-17	57	39	4
0	George W. Bush	2009-01-09	2009-01-11	34	61	5
1	George W. Bush	2008-12-12	2008-12-14	29	67	4
2	George W. Bush	2008-12-04	2008-12-07	32	61	7
0	William J. Clinton	2001-01-10	2001-01-14	66	29	5
1	William J. Clinton	2001-01-05	2001-01-07	63	31	4
2	William J. Clinton	2000-12-15	2000-12-17	71	26	1
0	George Bush	1993-01-08	1993-01-11	56	37	7
1	George Bush	1992-12-18	1992-12-20	49	40	10
2	George Bush	1992-12-04	1992-12-06	48	47	3

Agruparemos por cada presidente, iteraremos a través del grupo, e individualmente trazaremos el índice de aprobación para cada fecha

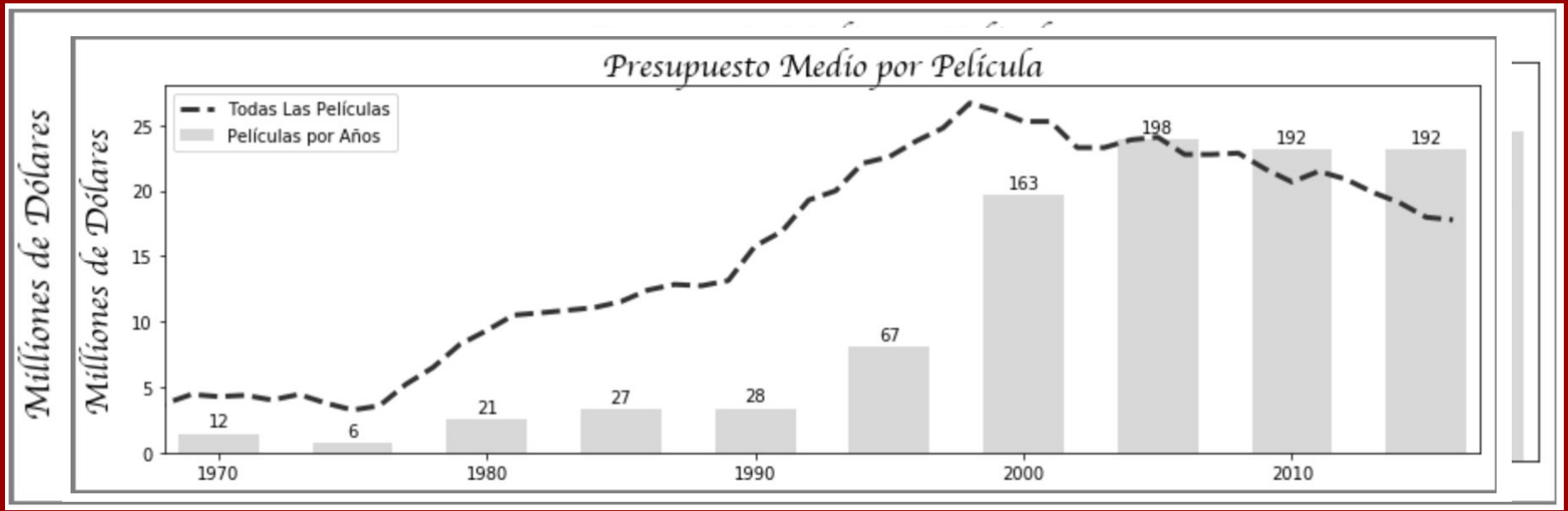


1. En movie.csv, se calcula la media del presupuesto para cada año, y luego el promedio móvil de cinco años para suavizar los datos.
2. Se llevan a Numpy arrays.
3. Se grafica la mediana "rolling" de los presupuestos a lo largo del tiempo en una nueva figura.

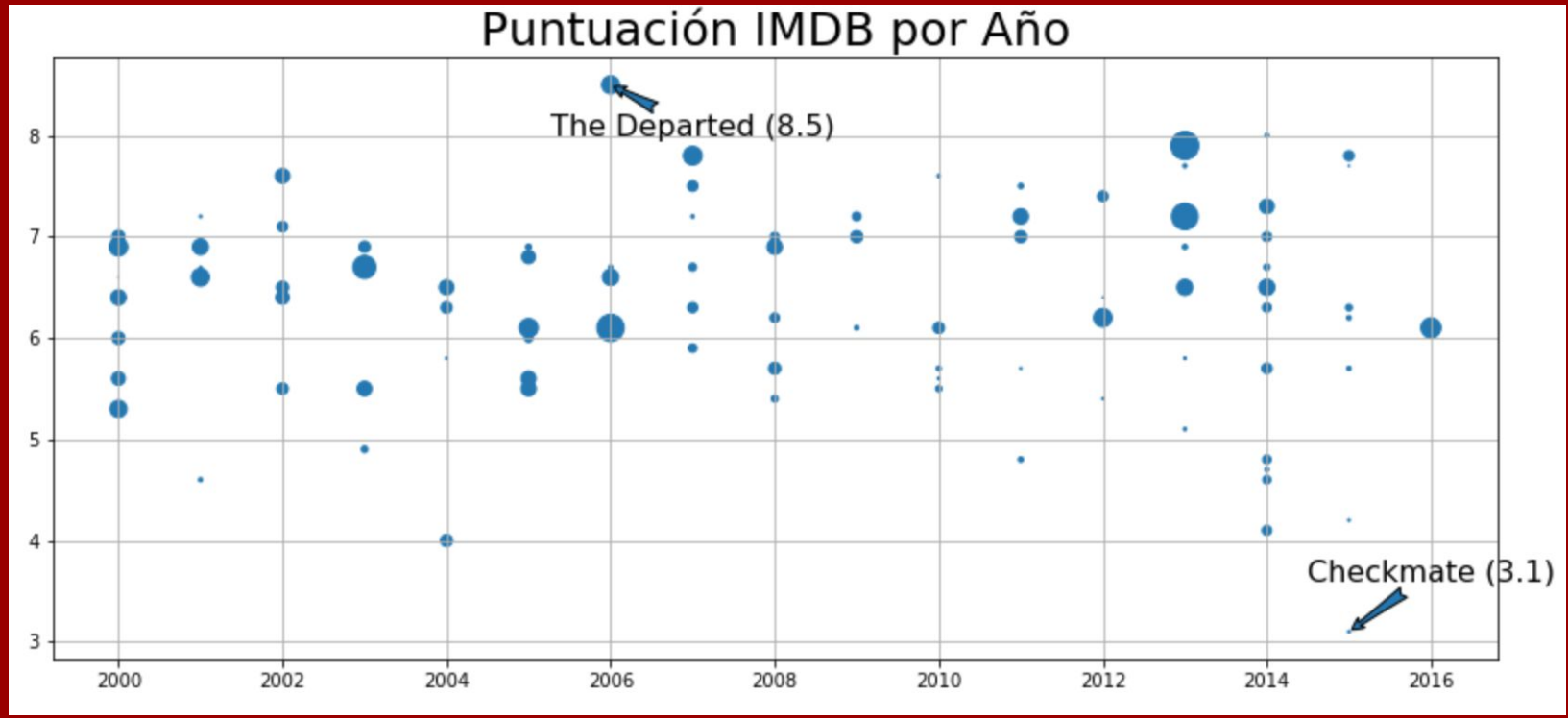


Es bastante interesante que el presupuesto medio de la película en el año 2000 y posteriormente se redujo. Tal vez esto es sólo un artefacto del conjunto de datos, en el que tenemos más datos en los últimos años de todas las películas, no sólo las más populares. Encontremos el recuento del número de películas por año.

Como las unidades de ambos gráficos son completamente diferentes (dólares frente a recuento), se escalan los recuentos para que estén en el mismo rango que el presupuesto, adicional se etiqueta cada barra con su valor como texto.
Como la gran mayoría de los datos están contenidos en los últimos años, se limitan los datos a las películas hechas a partir de 1970.



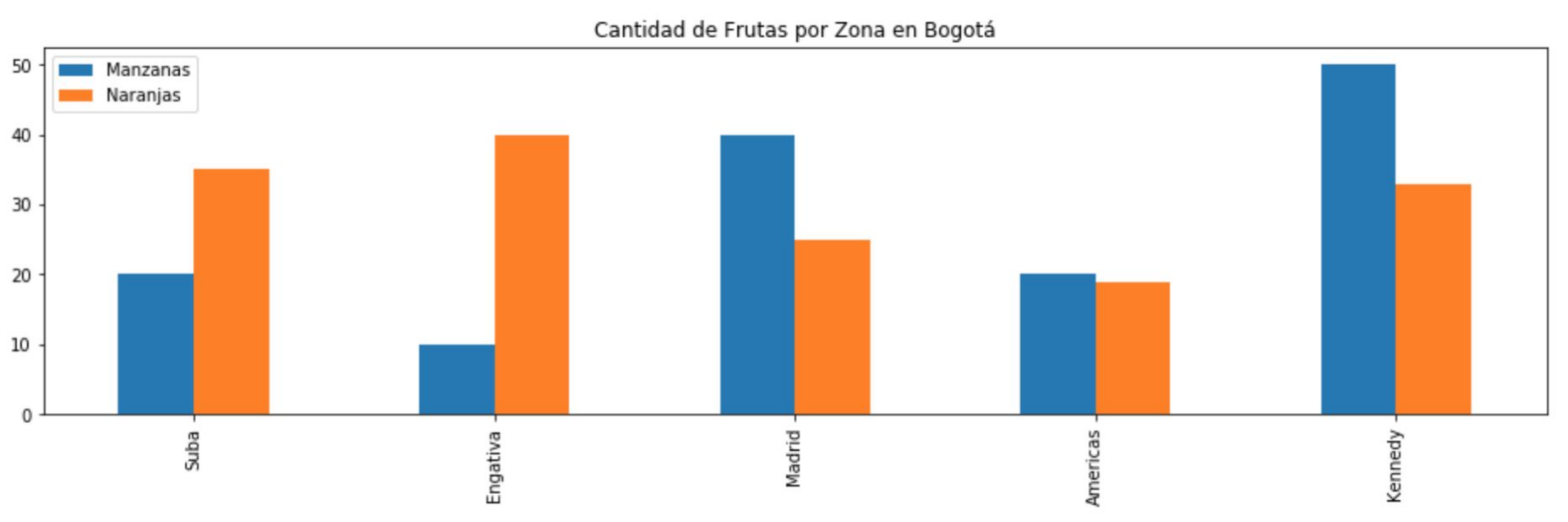
A continuación un plot de dispersión del puntaje del IMDB contra el año para una selección aleatoria de 100 películas hechas a partir del año 2000. El tamaño de cada punto es proporcional al presupuesto.



Graficación Básica en Pandas

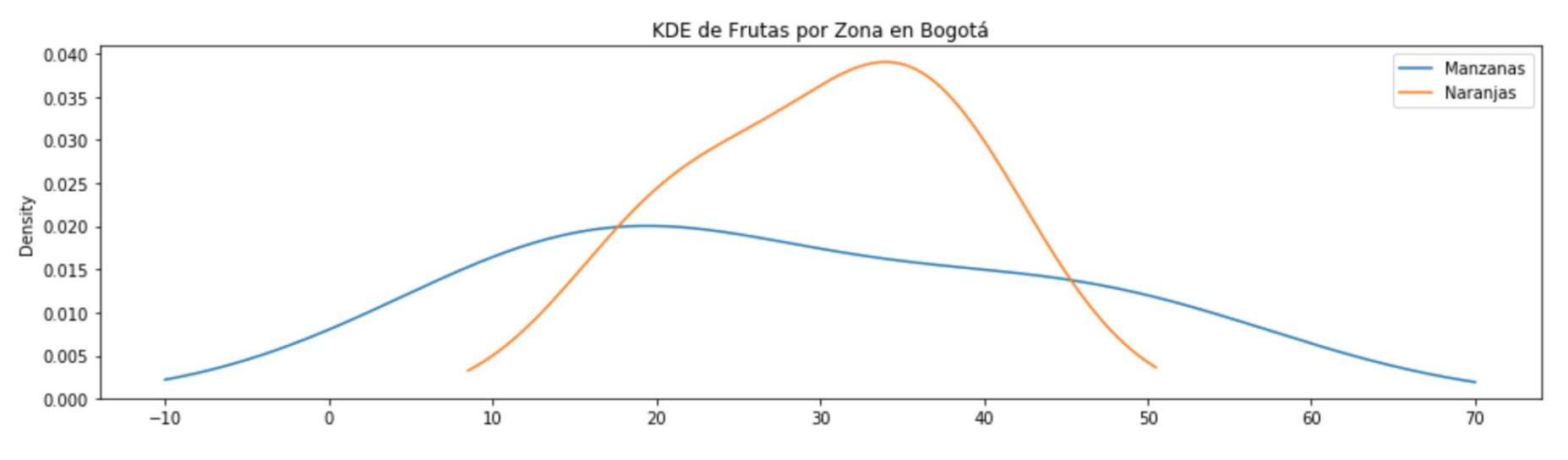
Todos los gráficos de pandas se maneja internamente por matplotlib y se accede públicamente a través del método de PLOT de DataFrame o Series. Decimos que el método PLOT de pandas es "envolver" alrededor de matplotlib. Cuando creas un gráfico en pandas, se te devolverá un eje o figura matplotlib. Se puede usar toda la potencia de matplotlib para modificar este objeto hasta que se obtenga el resultado deseado.

	Manzanas	Naranjas
Suba	20	35
Engativa	10	40
Madrid	40	25
Americas	20	19
Kennedy	50	33

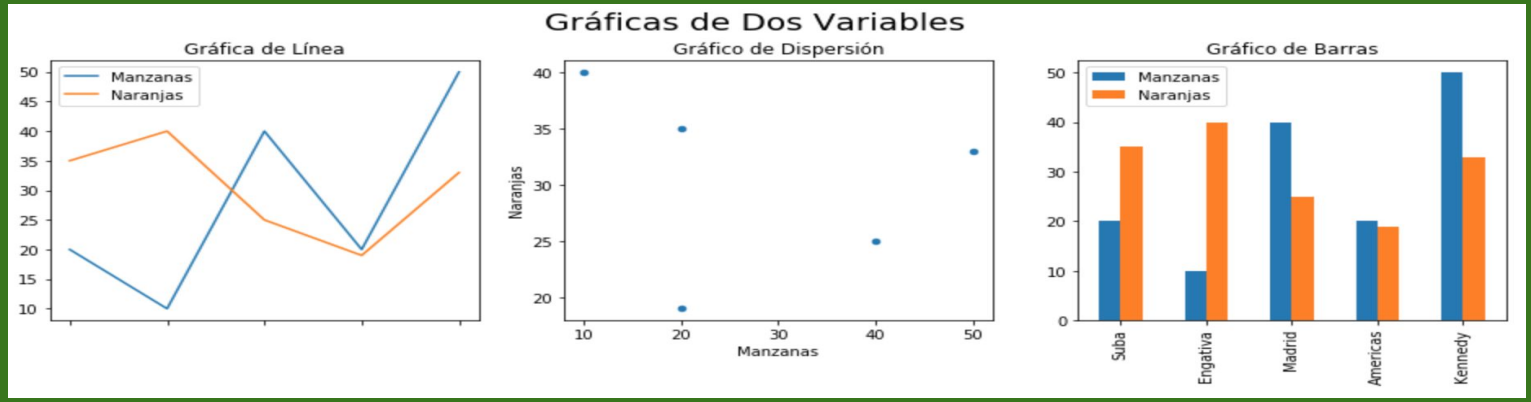

Cantidad de Frutas por Zona en Bogotá
Manzanas
Naranjas
50
40
30
20
10
0
Suba
Engativa
Madrid
Americas
Kennedy

Graficación Básica en Pandas

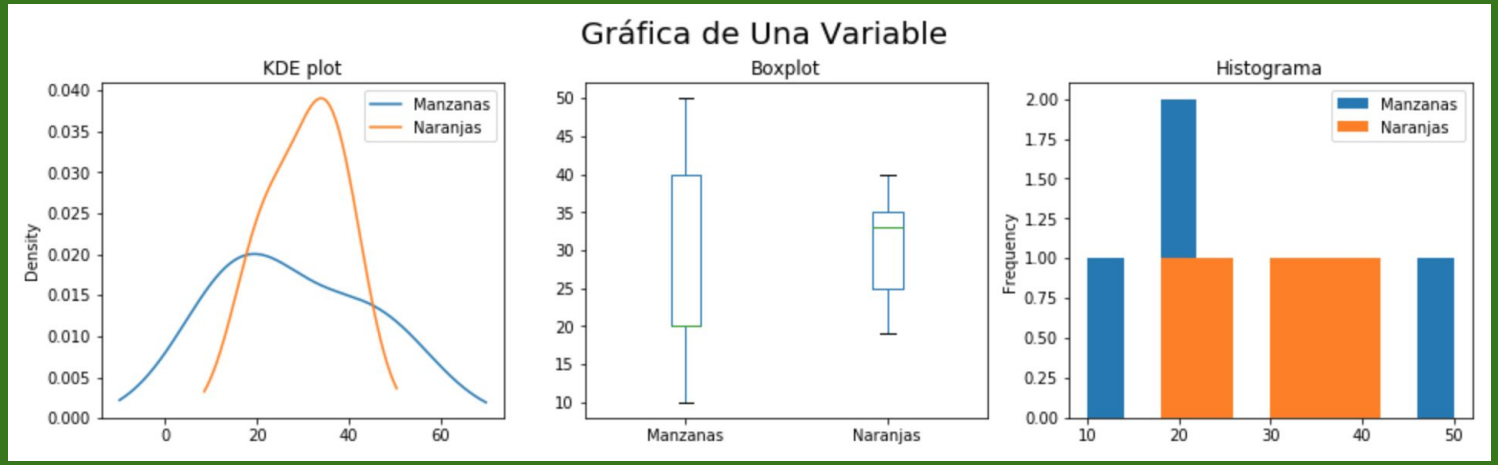
KDE: Genera el gráfico de estimación de la densidad de los núcleos usando núcleos gausianos. En estadística, la estimación de la densidad del núcleo (KDE) es una forma no paramétrica de estimar la función de densidad de probabilidad (PDF) de una variable aleatoria. Esta función utiliza núcleos gausianos e incluye la determinación automática del ancho de banda.



Gráficas de dos variables

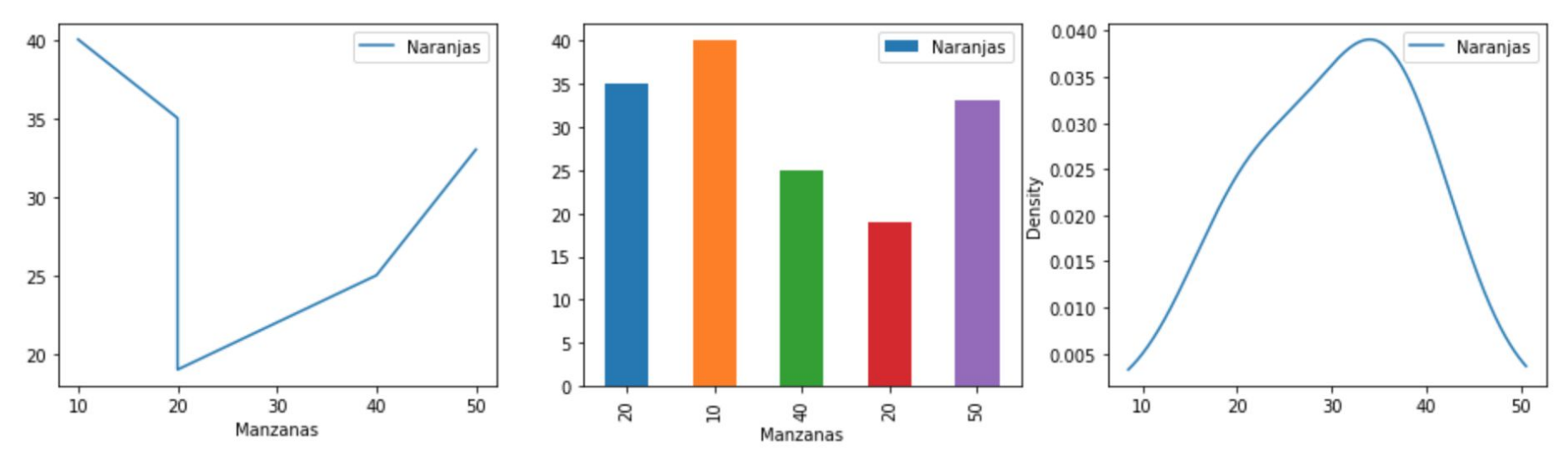


Gráficas de una variable



Graficación Básica en Pandas

Con la excepción del gráfico de dispersión, todos los gráficos no especifican las columnas a utilizar. Pandas utilizan por defecto cada una de las columnas numéricas, así como el índice en el caso de los gráficos de dos variables. Por supuesto, se pueden especificar las columnas exactas que se desea utilizar para cada valor "X" o "Y".



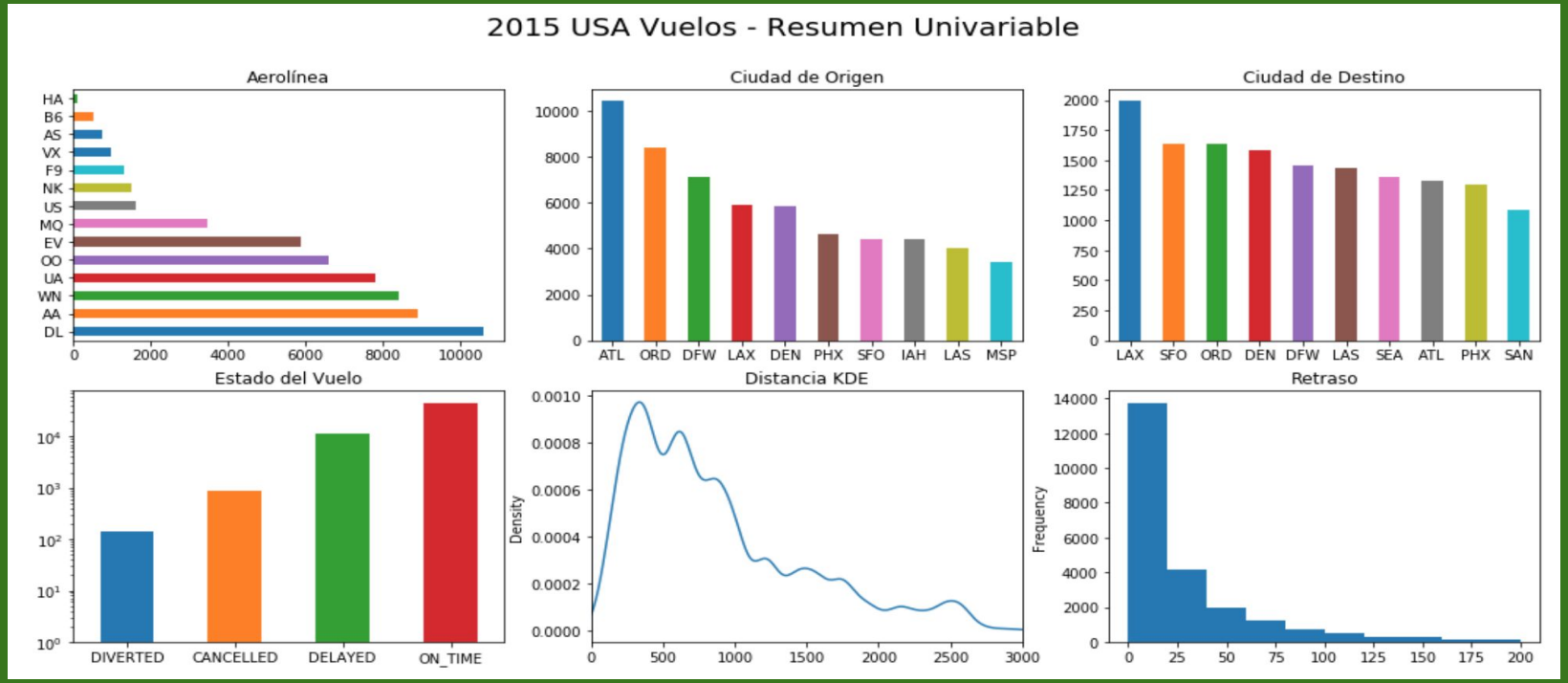
Graficación Básica en Pandas: Ejemplo con flights.csv

El análisis exploratorio de datos está guiado principalmente por visualizaciones, pandas proporcionan una gran interfaz para crear rápidamente y sin esfuerzo entonces. Una estrategia simple al comenzar una visualización de cualquier conjunto de datos es centrarse sólo en "gráficos" **univariados**. Los gráficos univariados más populares tienden a ser los gráficos de barras para datos categóricos (generalmente cadenas) y los histogramas, boxplots o KDEs para datos continuos (siempre numéricos). Intentar analizar múltiples variables al mismo tiempo, directamente al comienzo de un proyecto, puede ser bastante abrumador.

	MONTH	DAY	WEEKDAY	AIRLINE	ORG_AIR	DEST_AIR	SCHED_DEP	DEP_DELAY	AIR_TIME	DIST	SCHED_ARR	ARR_DELAY	DIVERTED	CANCELLED
0	1	1	4	WN	LAX	SLC	1625	58.0	94.0	590	1905	65.0	0	0
1	1	1	4	UA	DEN	IAD	823	7.0	154.0	1452	1333	-13.0	0	0
2	1	1	4	MQ	DFW	VPS	1305	36.0	85.0	641	1453	35.0	0	0
3	1	1	4	AA	DFW	DCA	1555	7.0	126.0	1192	1935	-7.0	0	0
4	1	1	4	WN	LAX	MCI	1720	48.0	166.0	1363	2225	39.0	0	0

Graficación Básica en Pandas: Ejemplo con flights.csv

1.- Antes de empezar a graficar, vamos a calcular el número de vuelos “DIVERTED”, “CANCELLED”, “DELAYED” y “ON_TIME”. Ya tenemos columnas binarias para “DIVERTED” y “CANCELLED”. Los vuelos se consideran retrasados cuando llegan 15 minutos o más tarde de lo previsto. Vamos a crear dos nuevas columnas binarias para rastrear las llegadas “DELAYED” y “ON_TIME”.



Graficación Básica en Pandas: Ejemplo con flights.csv

Antes de pasar a las gráficas multivariantes, se grafica el número de vuelos por semana. Infortunadamente, no tenemos el TimeStamp pandas en ninguna de las columnas, pero sí el mes y el día. La función to_datetime tiene un ingenioso truco que identifica los nombres de las columnas que coinciden con los componentes del TimeStamp. Por ejemplo, si tienes un DataFrame con exactamente tres columnas año, mes y día, entonces al pasar este DataFrame a la función to_datetime se devolverá una secuencia de TimeStamps. Para preparar nuestro actual DataFrame, necesitamos añadir una columna para el año y usar la hora de salida programada para obtener la hora y los minutos:

	MONTH	DAY	HOUR	MINUTE	YEAR
0	1	1	16	25	2019
1	1	1	8	23	2019
2	1	1	13	5	2019
3	1	1	15	55	2019
4	1	1	17	20	2019

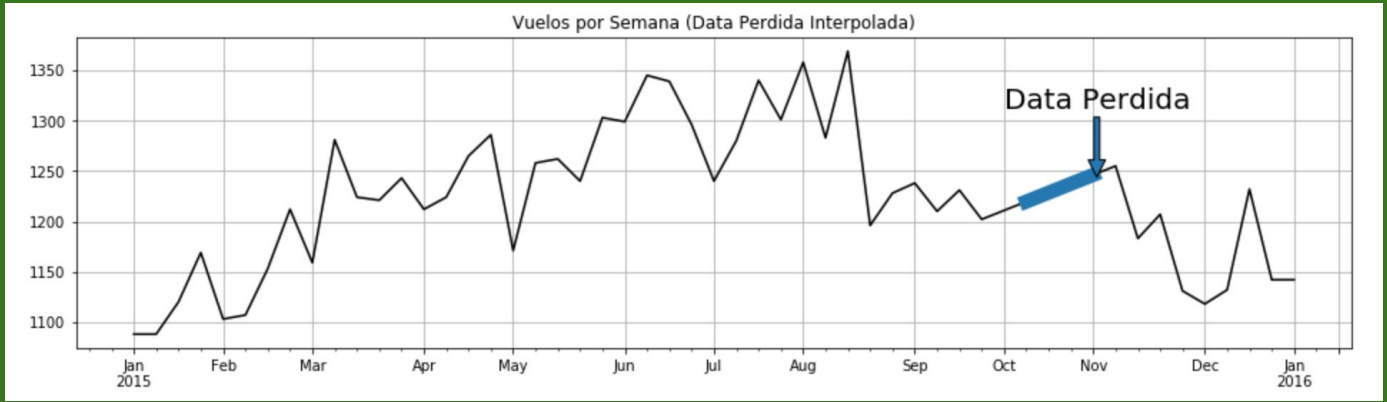
Graficación Básica en Pandas: Ejemplo con flights.csv

Podemos convertir este DataFrame en una Serie de TimeStamps con la función to_datetime.

```
58487   2019-12-31 05:15:00
58488   2019-12-31 19:10:00
58489   2019-12-31 18:46:00
58490   2019-12-31 05:25:00
58491   2019-12-31 08:59:00
dtype: datetime64[ns]
```

Usemos este resultado como nuestro nuevo índice y luego encontremos el conteo de vuelos por semana con el método de resample

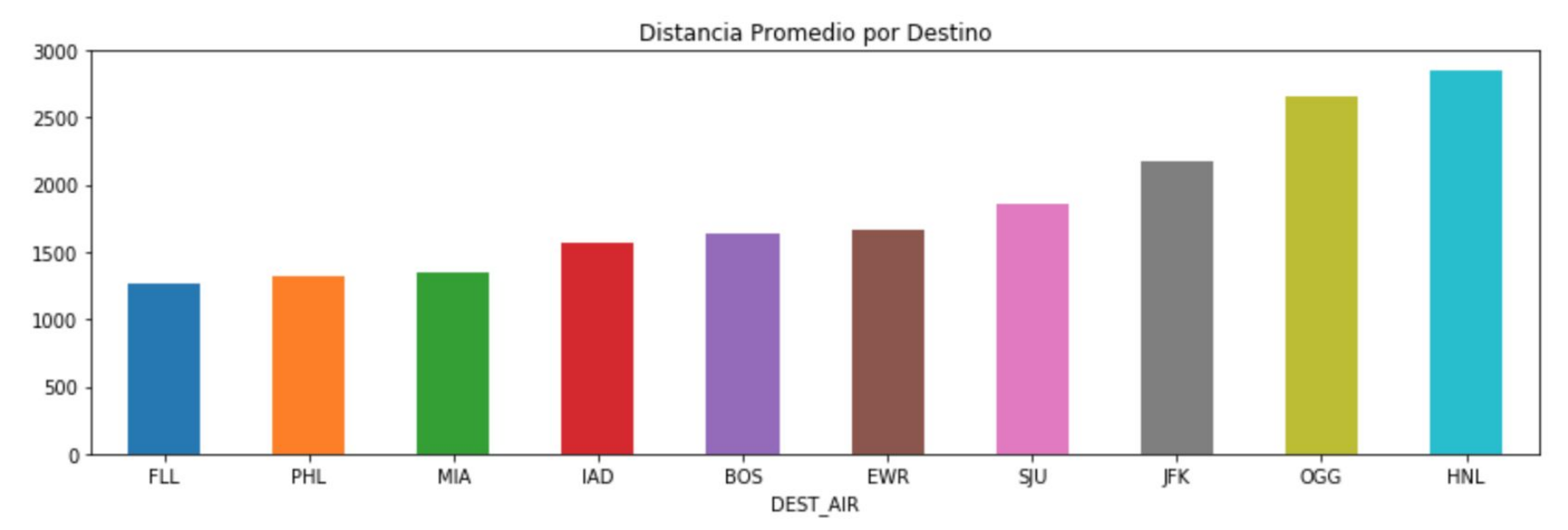
Hagamos cualquier semana de datos con menos de 1000 vuelos perdidos. Entonces, podemos usar el método de interpolación para completar los datos que faltan.



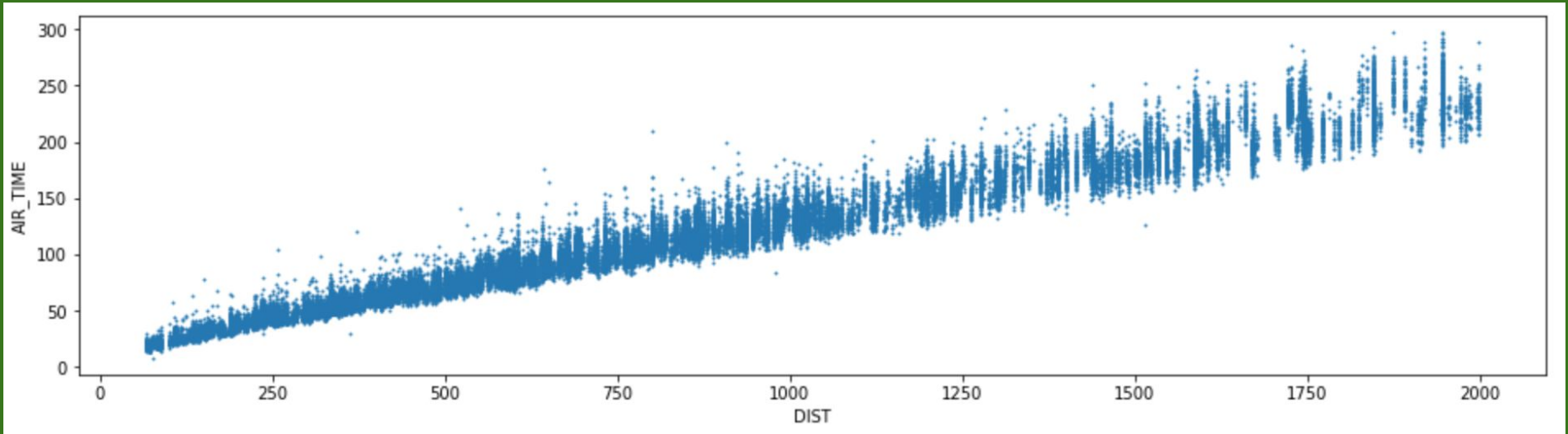
Usemos este resultado como nuestro nuevo índice y luego encontremos el conteo de vuelos por semana con el método de resample

Encontrar los 10 aeropuertos que:

- Tienen la mayor distancia promedio recorrida para los vuelos de llegada
- Tienen un mínimo de 100 vuelos en total



- Analicemos dos variables al mismo tiempo haciendo un gráfico de dispersión entre la distancia y el tiempo de vuelo para todos los vuelos de menos de 2.000 millas.



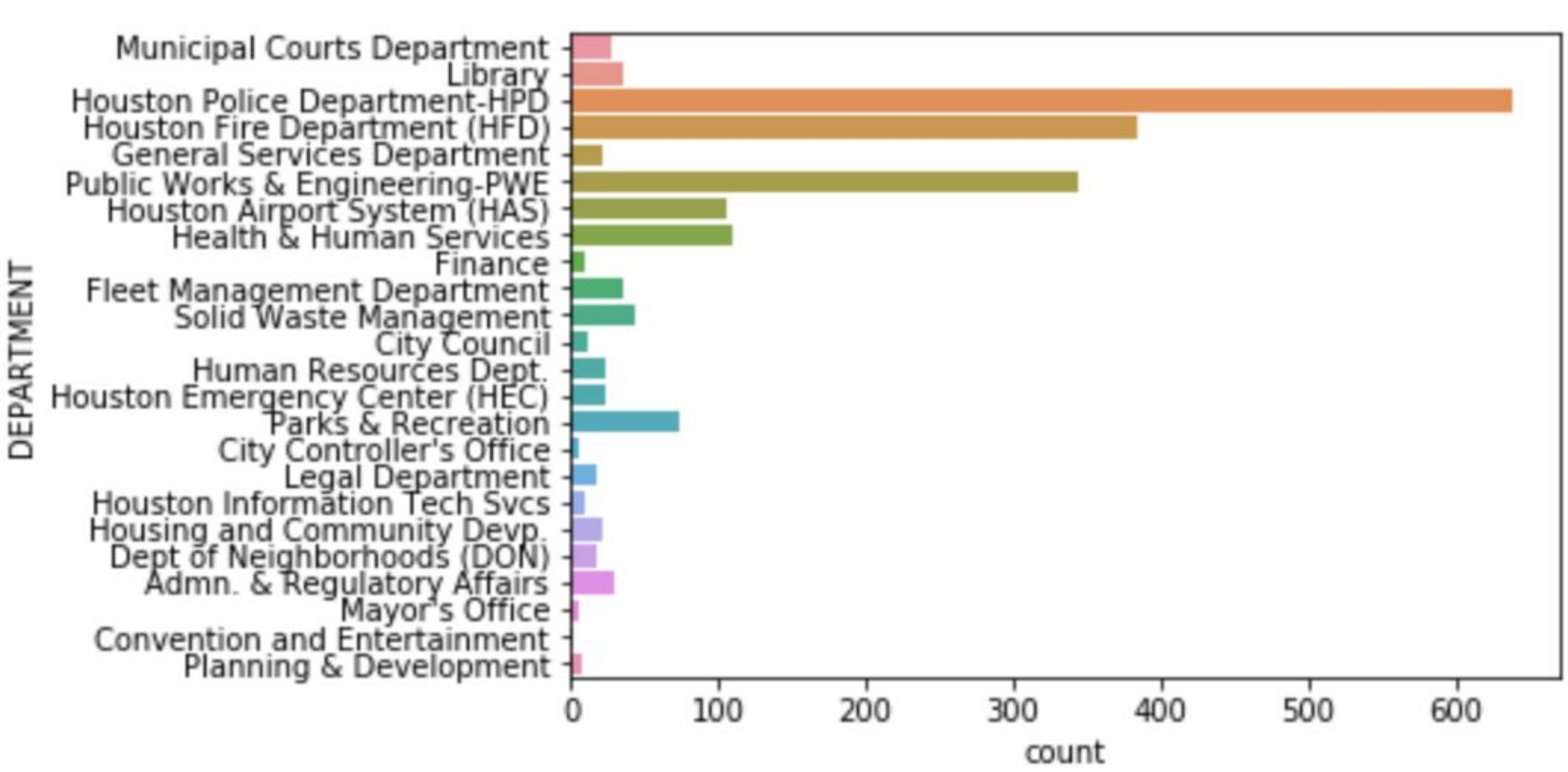
Biblioteca SEABORN con Pandas

La biblioteca Seaborn es una de las más populares en la "comunidad de ciencia de datos de Python" para crear visualización. Al igual que Pandas, no hace ningún tipo de gráfica en sí mismo y depende completamente de matplotlib para el trabajo pesado. Las funciones de seaborn para graficar trabajan directamente con los DataFrames de Pandas para crear visualizaciones estéticamente agradables.

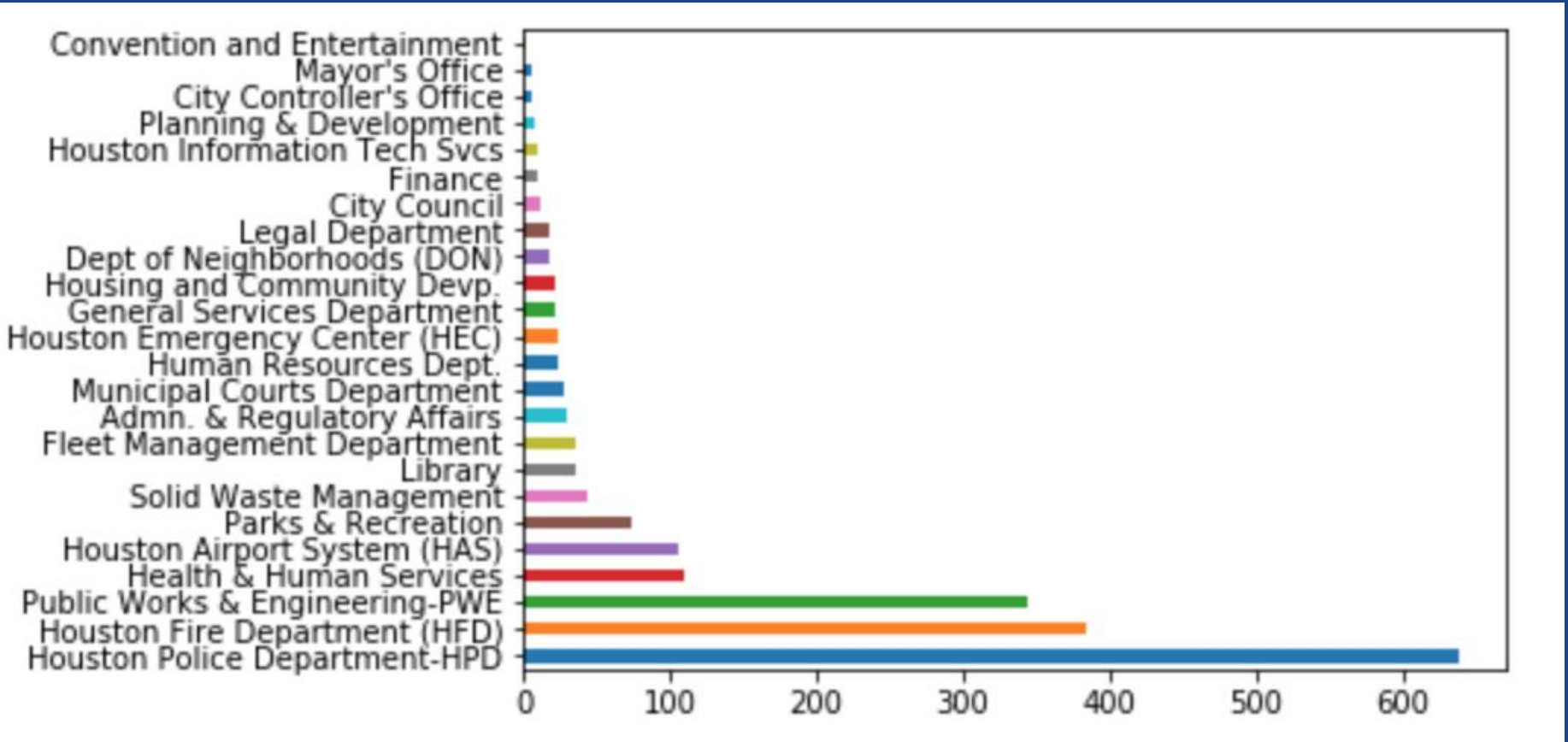
- Mientras que Seaborn y Pandas reducen la sobrecarga de matplotlib, la forma en que se acercan a los datos es completamente diferente.
- Casi todas las funciones de trazado de Seaborn requieren datos ordenados (o largos).
- Cuando los datos están en forma ordenada, no están listos para su consumo o interpretación hasta que se les aplica alguna función para obtener un resultado.
- Los datos ordenados son un bloque de construcción en bruto que hace posible todos los demás análisis.
- El procesamiento de los datos ordenados durante el análisis de los datos a menudo crea datos agregados o amplios. Estos datos, en formato amplio, es lo que Pandas utiliza para hacer sus gráficos.

UNIQUE_ID	POSITION_TITLE	DEPARTMENT	BASE_SALARY	RACE	EMPLOYMENT_TYPE	GENDER	EMPLOYMENT_STATUS	HIRE_DATE	JOB_DATE
0	ASSISTANT DIRECTOR (EX LVL)	Municipal Courts Department	121862.0	Hispanic/Latino	Full Time	Female	Active	2006-06-12	2012-10-13
1	LIBRARY ASSISTANT	Library	26125.0	Hispanic/Latino	Full Time	Female	Active	2000-07-19	2010-09-18
2	POLICE OFFICER	Houston Police Department-HPD	45279.0	White	Full Time	Male	Active	2015-02-03	2015-02-03
3	ENGINEER/OPERATOR	Houston Fire Department (HFD)	63166.0	White	Full Time	Male	Active	1982-02-08	1991-05-25
4	ELECTRICIAN	General Services Department	56347.0	White	Full Time	Male	Active	1989-06-19	1994-10-22

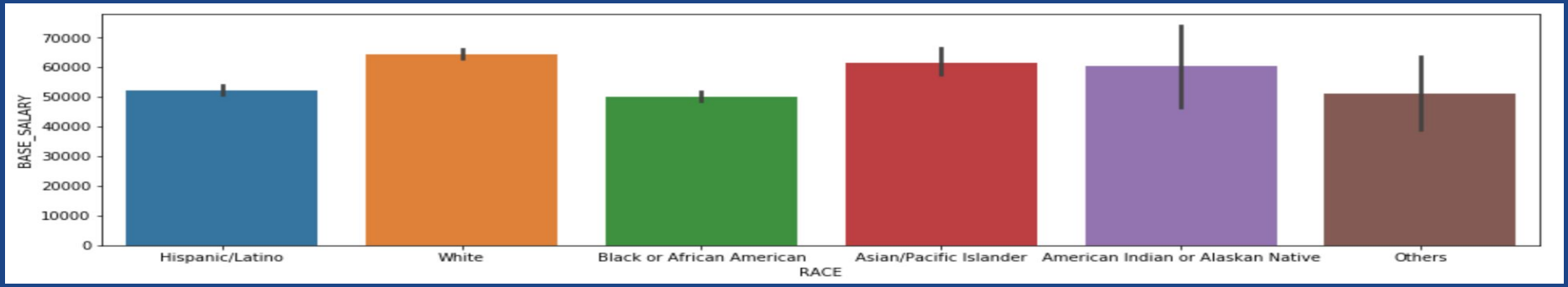
Usando Seaborn: Se crea una gráfica de barras con cada departamento



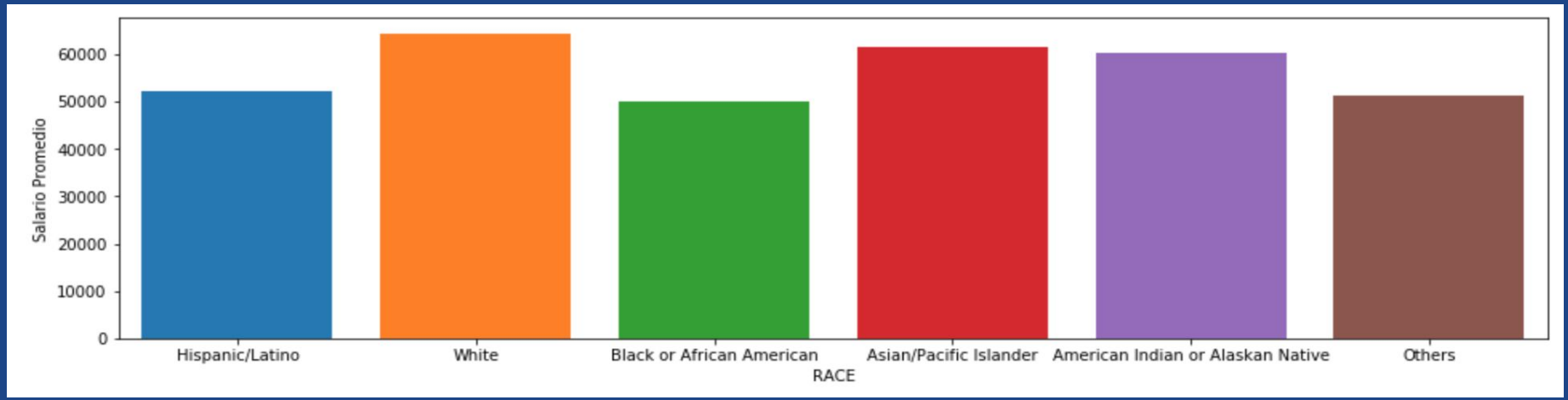
Para reproducir el mismo gráfico en Pandas, se necesita agregar los datos de antemano



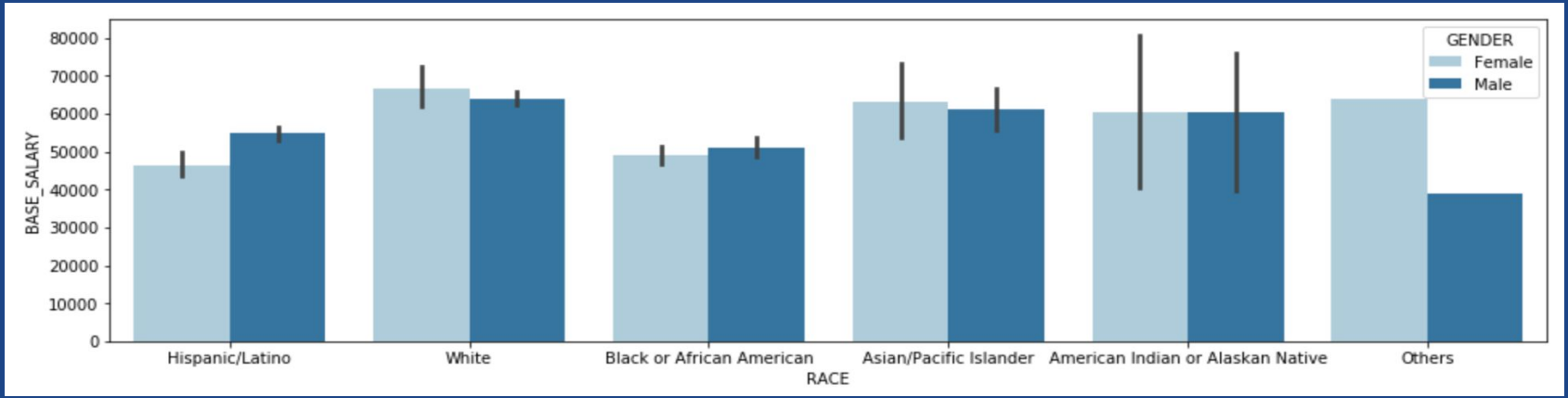
Seaborn: Promedio de salario por raza de empleados



Pandas Promedio de salario por raza de empleados



Seaborn: Promedio de salario por raza de empleados y genero



Pandas Promedio de salario por raza de empleados y genero

References

- Python Programming: An Introduction to Computer Science. John Zelle
- Big Data con Python. Rafael Caballero Enrique Martín Adrián Riesco
- Aprende Python en un Fin de Semana Alfredo Moreno Muñoz Sheila Córcoles Córcoles
- Learn Python Programming Fabrizio Romano
- Python Data Analytics Fabio Nelli
- Expert Python Programming Michael Jasworski Tarek Ziadé
- Statistical analysis of questionnaires: a unified approach based on R and Stata by Francesco Bartolucci. Boca Raton: CRC Press, 2016.
- Data visualisation: a handbook for data driven design by Andy Kirk. Los Angeles: Sage, 2016.
- Learning tableau: leverage the power of tableau 9.0 to design rich data visualizations and build fully interactive dashboards by Joshua N. Milligan. Mumbai: Packt Publishing, 2015.