Visualizing Data with Python

Module 4 (Part 1)

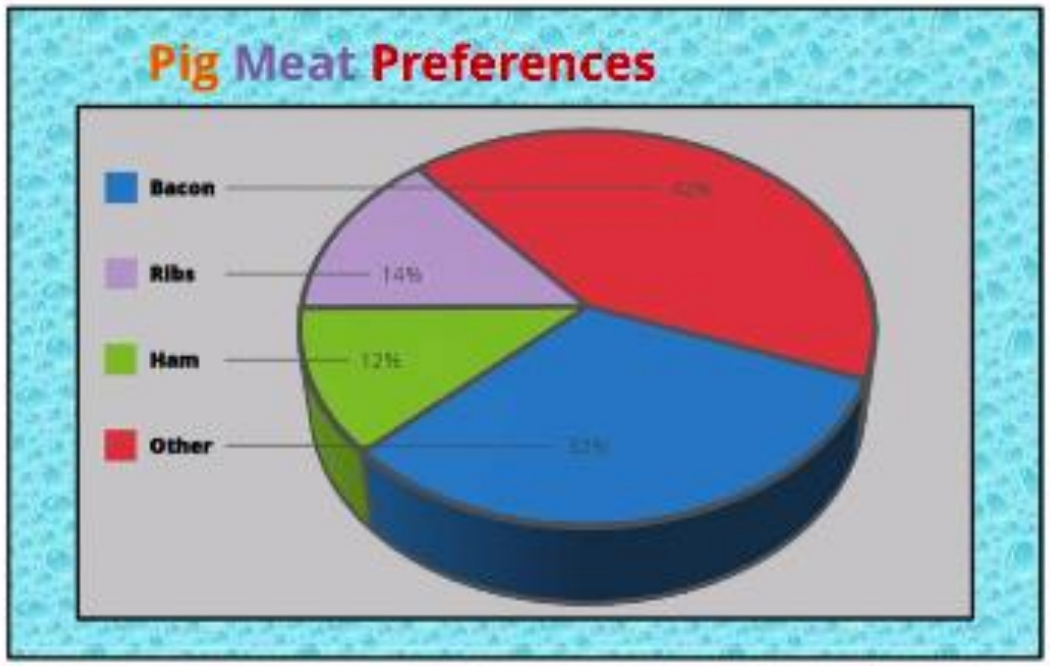
Structure

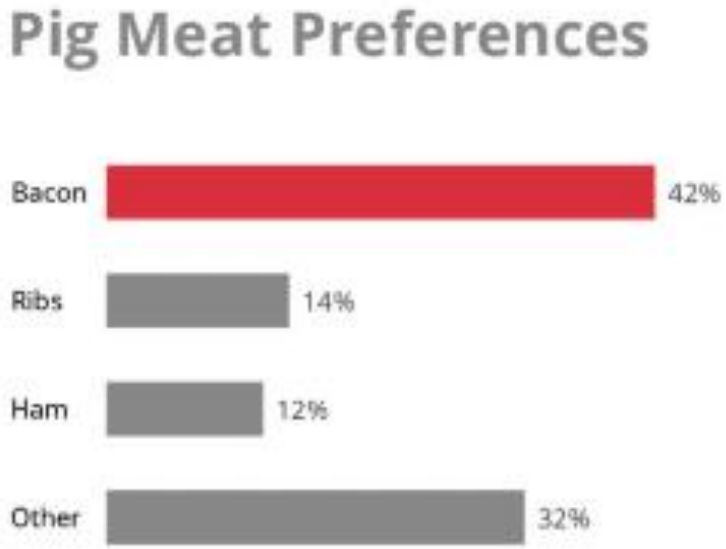
- Part 1: Introduction to Visualization Tools – best practices, basic plotting with Matplotlib, line plots
- Part 2: Basic Visualization Tools – area plots, histograms, bar charts
- Part 3: Specialized Visualization Tools – pie charts, box plots, scatter plots, bubble plots

Part 1

Introduction to Visualization Tools

Less is better

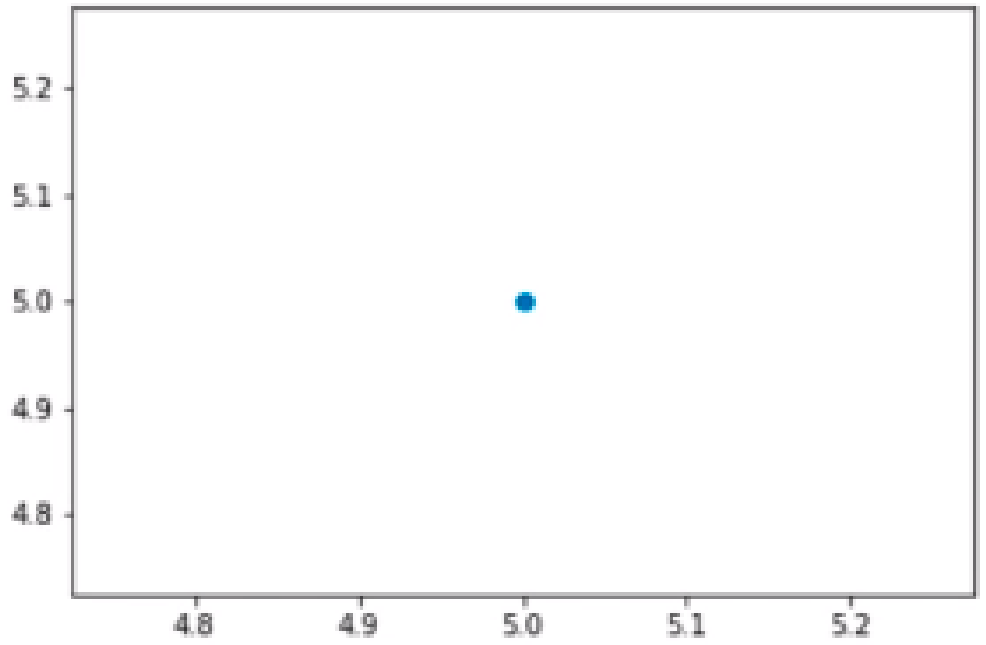




In this example, a bar graph is cleaner, simpler and less distracting than a pie chart.

Matplotlib – Plot Function

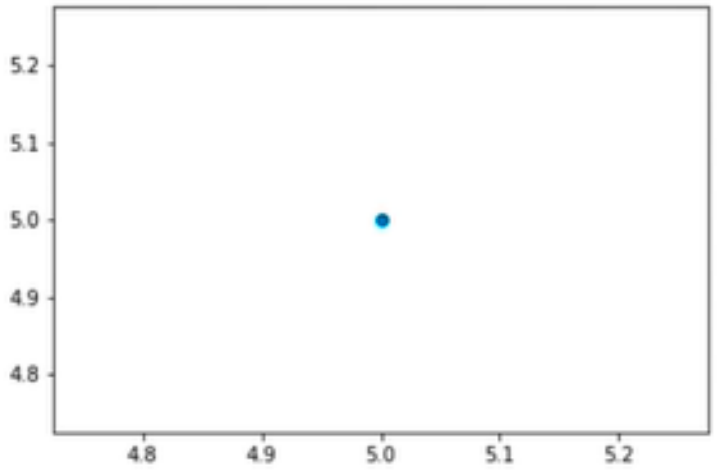
```
In [1]: import matplotlib.pyplot as plt

In [2]: plt.plot(5, 5, 'o')
        plt.show()
```



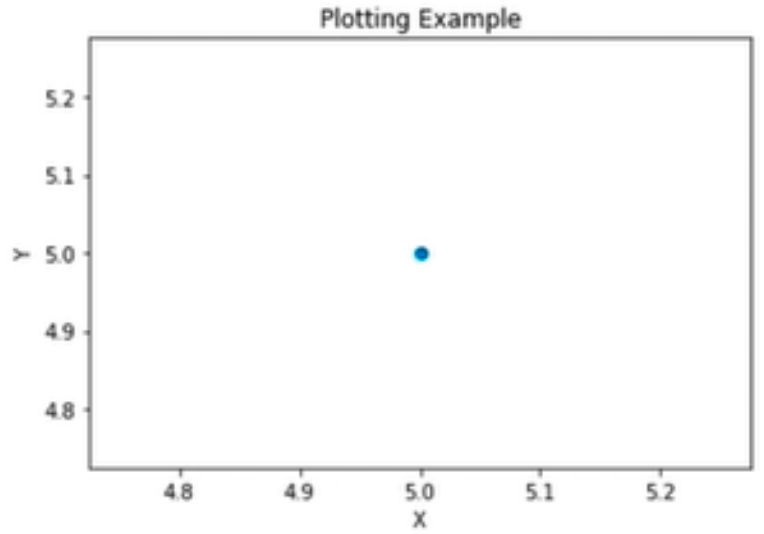
After rendering this image, we cannot add details to the plot.

Matplotlib – Plot Function (cont.)

```
In [1]: %matplotlib inline
        import matplotlib.pyplot as plt

In [2]: plt.plot(5, 5, 'o')
        plt.show()
```



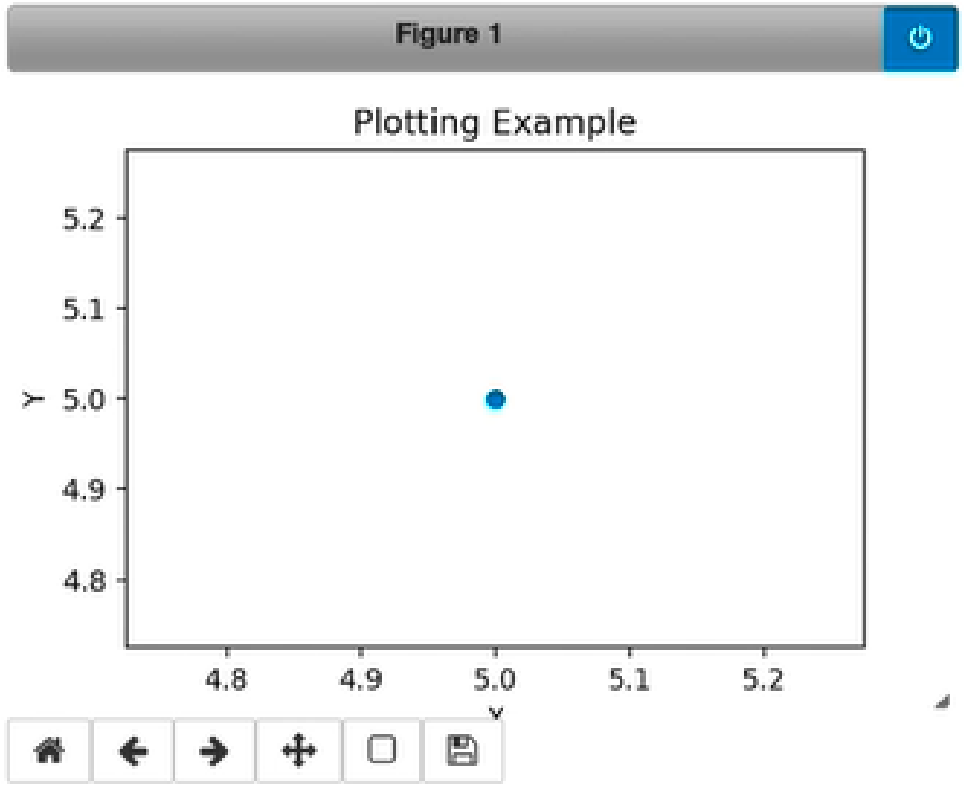

```
In [3]: plt.plot(5, 5, 'o')

Out[3]: plt.ylabel("Y")
        plt.xlabel("X")
        plt.title("Plotting Example")

        plt.show()
```



We will have to create the plot (with the new details) from scratch.

Matplotlib Backends – Notebook

```
In [1]: %matplotlib notebook
        import matplotlib.pyplot as plt

In [2]: plt.plot(5, 5, 'o')
```




```
Out[2]: [<matplotlib.lines.Line2D at 0x10784c790>]
```

Using the ‘notebook’ backend, we can add details to the plotting object after it has been rendered.

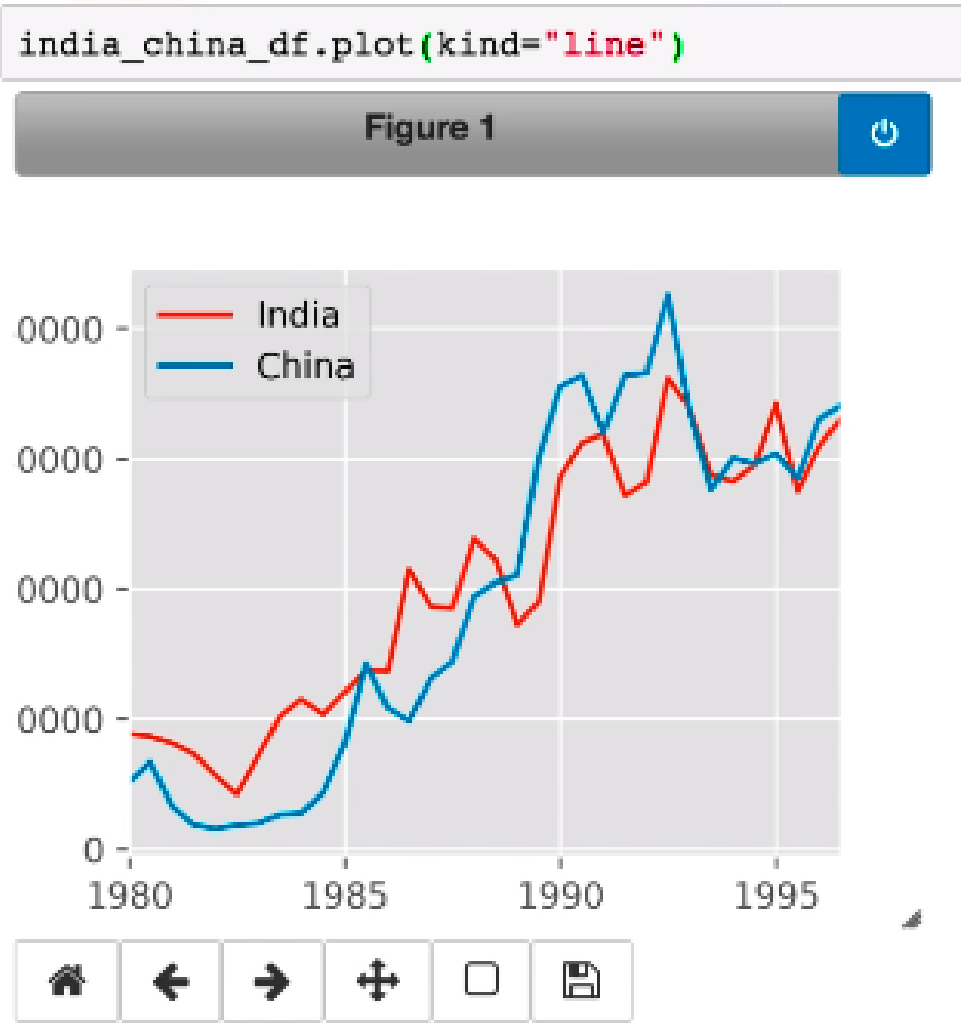
```
In [3]: plt.ylabel("Y")
        plt.xlabel("X")
        plt.title("Plotting Example")

Out[3]: <matplotlib.text.Text at 0x1077f2910>
```

Matplotlib – pandas

india_china_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df.plot(kind="line")
```

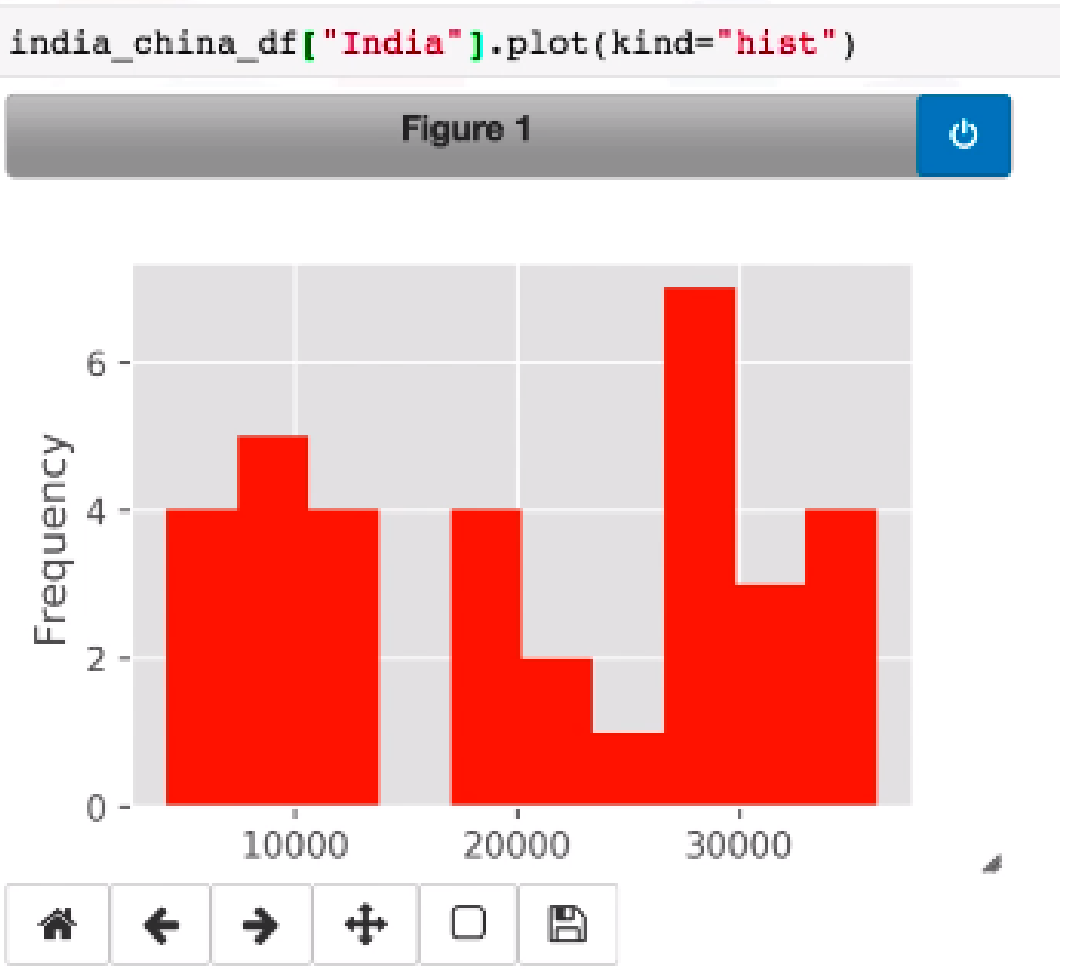


Matplotlib – pandas (cont.)

india_china_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df["India"].plot(kind="hist")
```



The dataset

- The Population Division of the United Nations compiled data pertaining to 45 countries.
- For each country, annual data on the flows of international migrants is reported in addition to other metadata.
- We will primarily work with a United Nations data on immigration to Canada.

Immigration Data

International Migration Flows to and from Selected Countries: The 2015 Revision. (United Nations database,

Reporting country: Canada
Criterion: Citizenship

Classification		Origin/Destination	Major area		Region		Development region						
Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982	1983	1984
Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	39	39	47	71
Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	0	0	0	0
Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	67	71	69	63
Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	1	0	0	0
Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	0	0	0	0

Reading Data into Pandas Dataframe

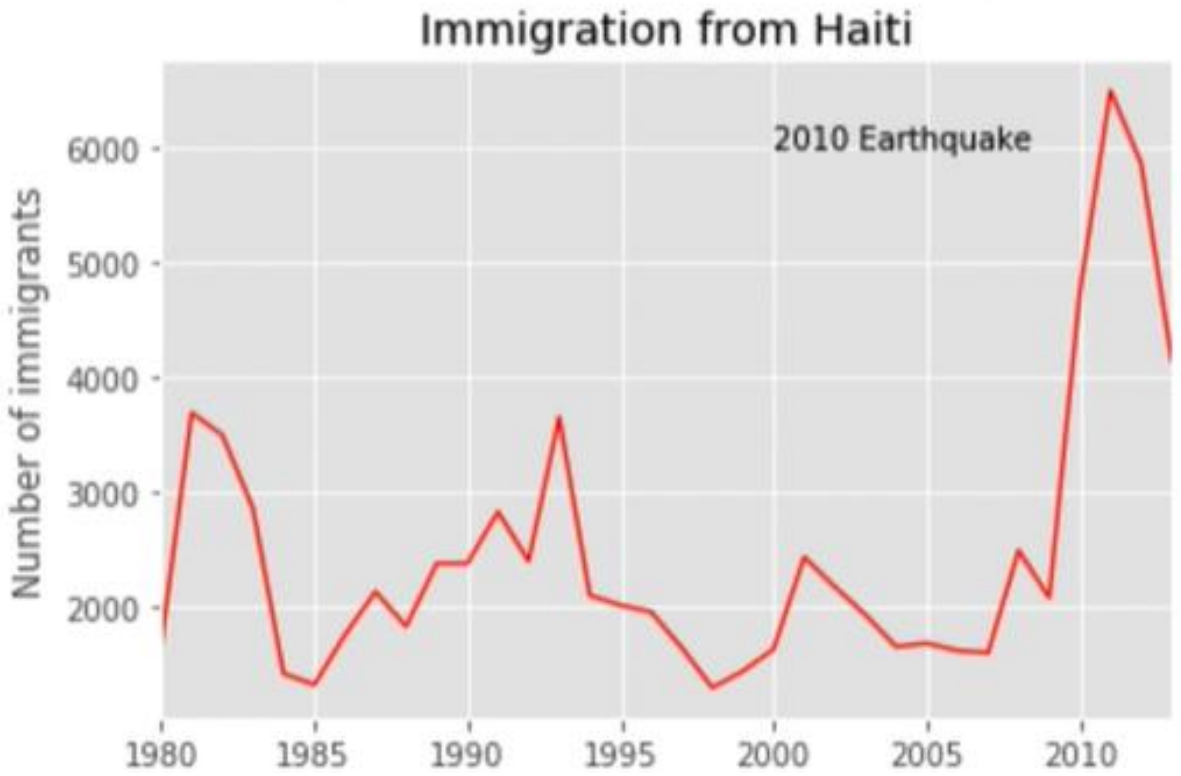
```
import numpy as np  # useful for many scientific computing in Python
import pandas as pd # primary data structure library
from __future__ import print_function # adds compatibility to python 2
```

```
# install xlrd
!pip install xlrd

print('xlrd installed!')
```

```
df_can = pd.read_excel(
    'https://ibm.box.com/shared/static/lw190pt9zpy5bd1ptyg2aw15awomz9pu.xlsx',
    sheetname="Canada by Citizenship",
    skiprows=range(20),
    skip_footer = 2)
```

Line Plots

A line plot is a type of plot which displays information as a series of data points called 'markers' connected by straight line segments.



Dataset – recap

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	...	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	...	2978	3436	3009	2652	2111	1746	1758	2203	2635	2004
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	...	1450	1223	856	702	560	716	561	539	620	603
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	...	3616	3626	4807	3623	4005	5393	4752	4325	3774	4331
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	...	0	0	1	0	0	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	...	0	0	1	1	0	0	0	0	1	1

Dataset – processed

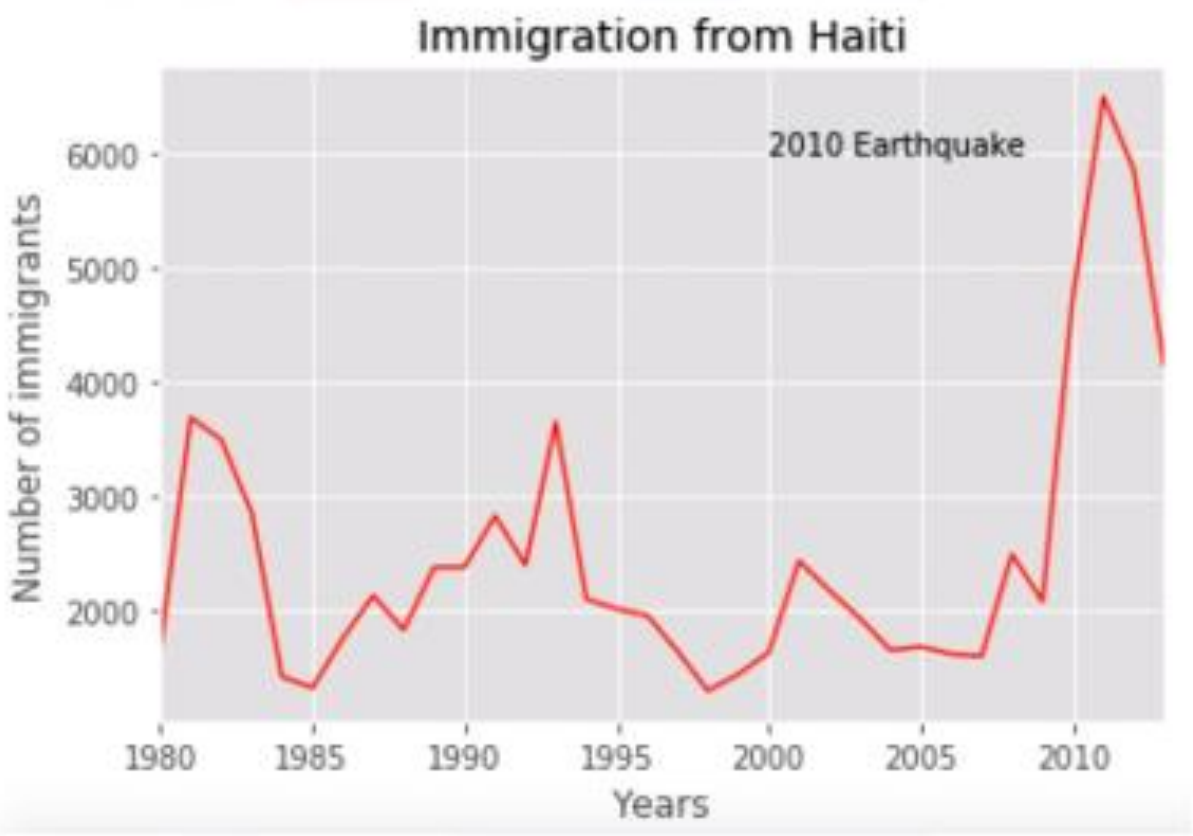
	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Country																					
Afghanistan	Asia	Southern Asia	Developing regions	16	39	39	47	71	340	496	...	3436	3009	2652	2111	1746	1758	2203	2635	2004	58639
Albania	Europe	Southern Europe	Developed regions	1	0	0	0	0	0	1	...	1223	856	702	560	716	561	539	620	603	15699
Algeria	Africa	Northern Africa	Developing regions	80	67	71	69	63	44	69	...	3626	4807	3623	4005	5393	4752	4325	3774	4331	69439
American Samoa	Oceania	Polynesia	Developing regions	0	1	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	6
Andorra	Europe	Southern Europe	Developed regions	0	0	0	0	0	0	2	...	0	1	1	0	0	0	0	1	1	15

Change the index to country name; this would make querying by country easier.
Add an additional column for the cumulative number of immigrants for each country.
Change the name of the dataframe to ‘df_canada’.

Creating Line Plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
years = list(map(str, range(1980, 2014)))

df_canada.loc['Haiti', years].plot(kind = 'line')
plt.title('Immigration from Haiti')
plt.ylabel('Number of immigrants')
plt.xlabel('Years')

plt.show()
```



Lab – Introduction to Matplotlib and Lineplots