Data Visualization

Agenda

1. Boxplot
2. Scatter Plot
3. Pie Chart Enhancement
4. Questions

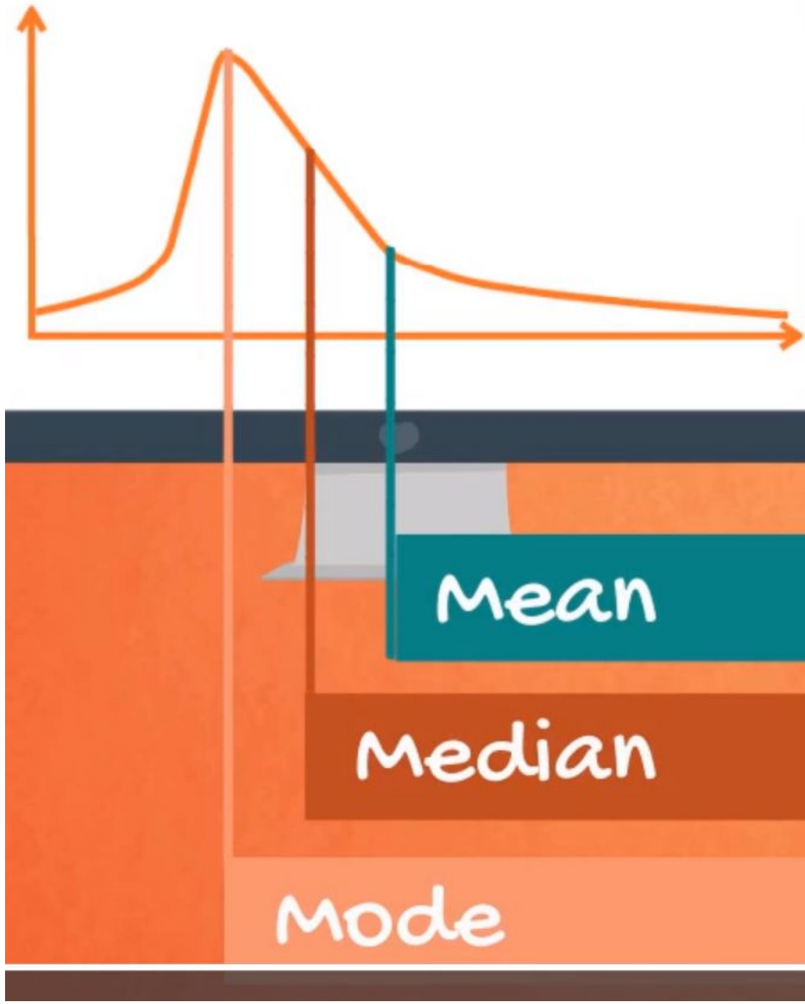
Boxplot

Mean vs Median vs Mode

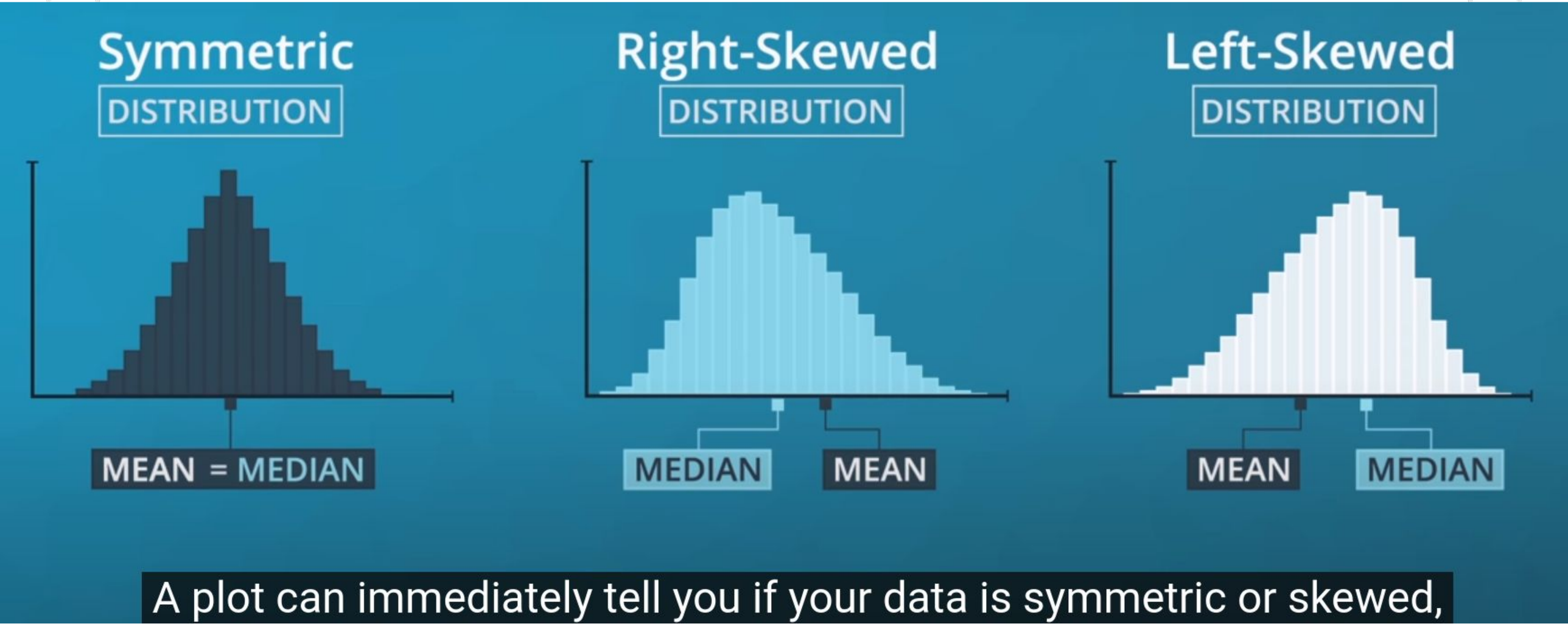
- **Mean (average)**
  - add all numbers in the data set and then
  - divide by the number of values in the data set

- **Median**
  - the middle value
  - when data set is ordered from least to greatest

- **Mode**
  - the number that occurs most often in a data set

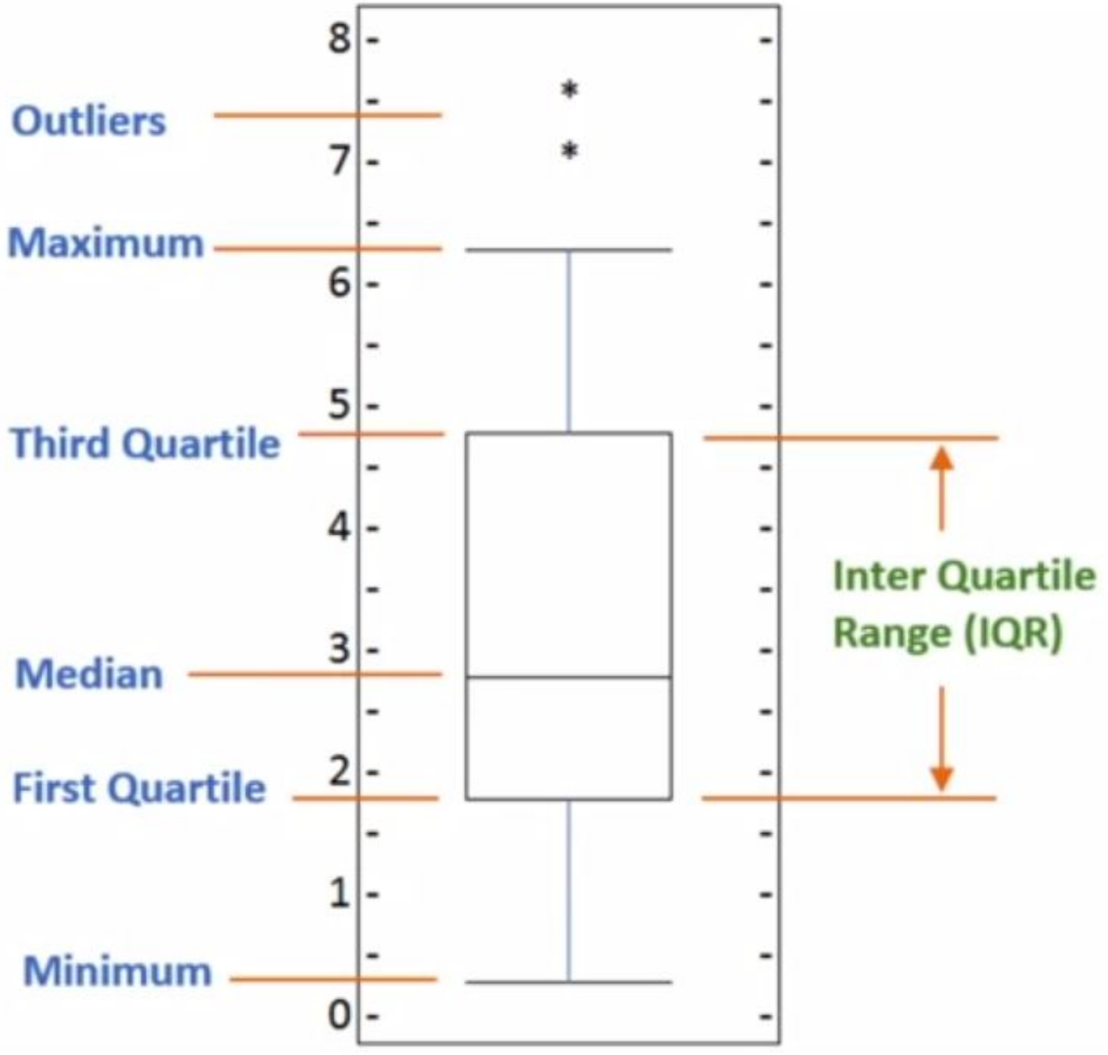


Mean vs Median



Boxplot

Statistically represents data distribution through **5 dimensions**



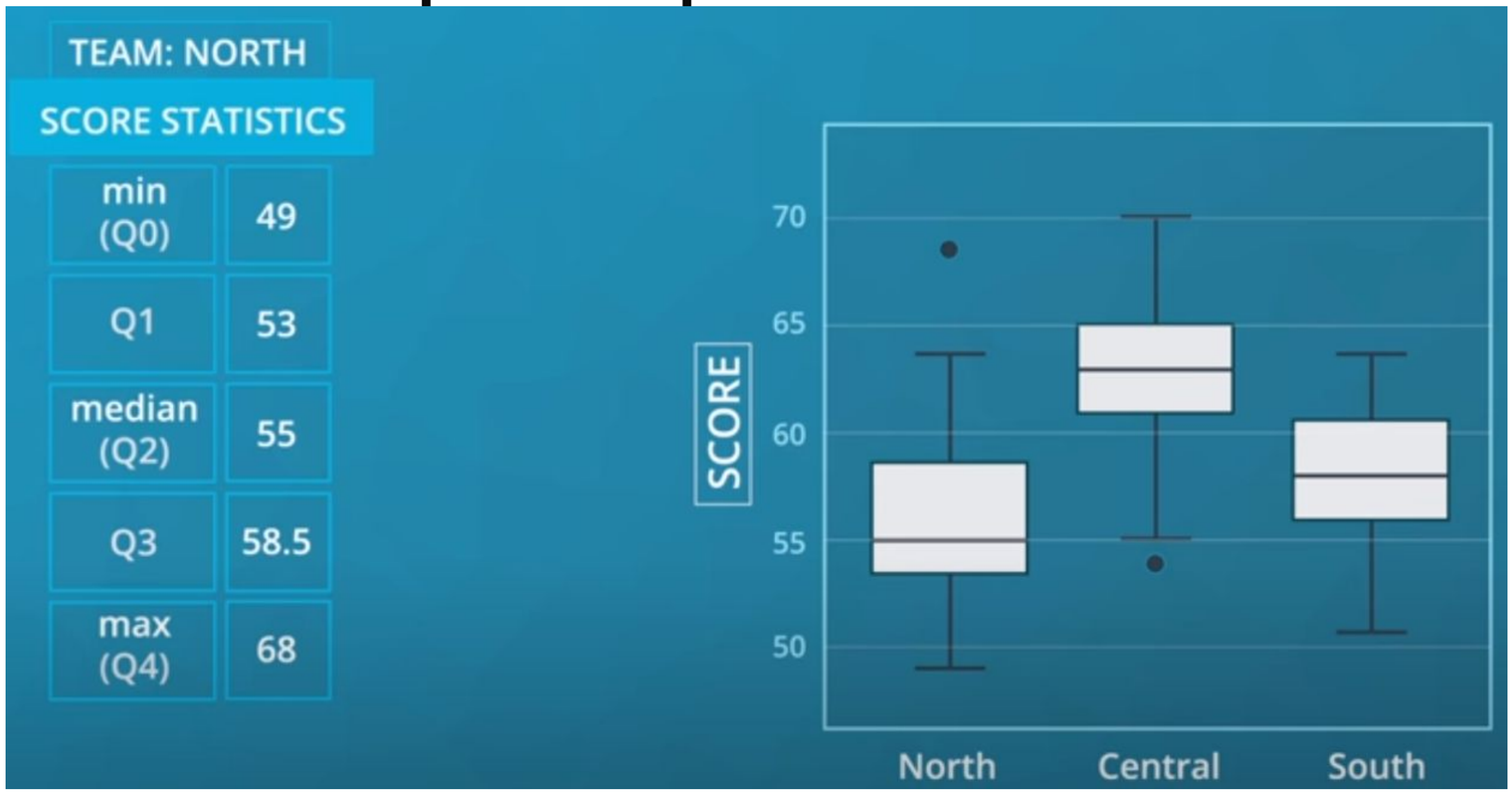
Boxplot

Statistically represents data distribution through **5 dimensions**

1. **minimum**: smallest number in data (read next slide note)
2. **first quartile**: 1/4 of data points are less than this value
3. **median**: median of the sorted data
4. **third quartile**: 3/4 of data points are less than this value
5. **maximum**: highest number in data (read next slide note)

** Display **outliers** as individual dots outside extremes

Actual min/max in the dataset vs min/max that used to plot Boxplot



- **Q1** = 53
- **Q2** = 55
- **Q3** = 58.5
- Calc:
  - **Interquartile Range (IQR)** = Q3 - Q1 = 58.5 - 53 = 5.5
  - **Max Whisker Length** = 1.5 * IQR = 1.5 * 5.5 = 8.25
  - **Upper Whisker Bound** = Q3 + Max Whisker Length = 58.5 + 8.25 = 66.75
  - **Lower Whisker Bound** = Q1 - Max Whisker Length = 53 - 8.25 = 44.75

Boxplot Example - Cell 1

In [1]:
```
import pandas as pd

df = pd.read_csv('canada-mig-dataset.csv')

df.head()
```

Out[1]:

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

5 rows × 43 columns

Boxplot Example - Cell 2

In [2]:
```
df1 = df.set_index('OdName')
df1.head()
```

Out[2]:

OdName	Type	Coverage	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	...	2004	2005	2006	2007	2008	2009	2010	2011	2
Afghanistan	Immigrants	Foreigners	935	Asia	5501	Southern Asia	902	Developing regions	16	39	...	2978	3436	3009	2652	2111	1746	1758	2203	2
Albania	Immigrants	Foreigners	908	Europe	925	Southern Europe	901	Developed regions	1	0	...	1450	1223	856	702	560	716	561	539	
Algeria	Immigrants	Foreigners	903	Africa	912	Northern Africa	902	Developing regions	80	67	...	3616	3626	4807	3623	4005	5393	4752	4325	3
American Samoa	Immigrants	Foreigners	909	Oceania	957	Polynesia	902	Developing regions	0	1	...	0	0	1	0	0	0	0	0	
Andorra	Immigrants	Foreigners	908	Europe	925	Southern Europe	901	Developed regions	0	0	...	0	0	1	1	0	0	0	0	

5 rows × 42 columns

Boxplot Example - Cell 3

In [3]:
```
df2 = df1.loc[ ['Japan'], list(map(str, range(1980,2014))) ]
df2.head()
```

Out[3]:

OdName	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	...	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Japan	701	756	598	309	246	198	248	422	324	494	...	973	1067	1212	1250	1284	1194	1168	1265	1214	982

1 rows × 34 columns

Boxplot Example - Cell 4

In [4]:
```
df_japan = df2.transpose()
df_japan.head()
```

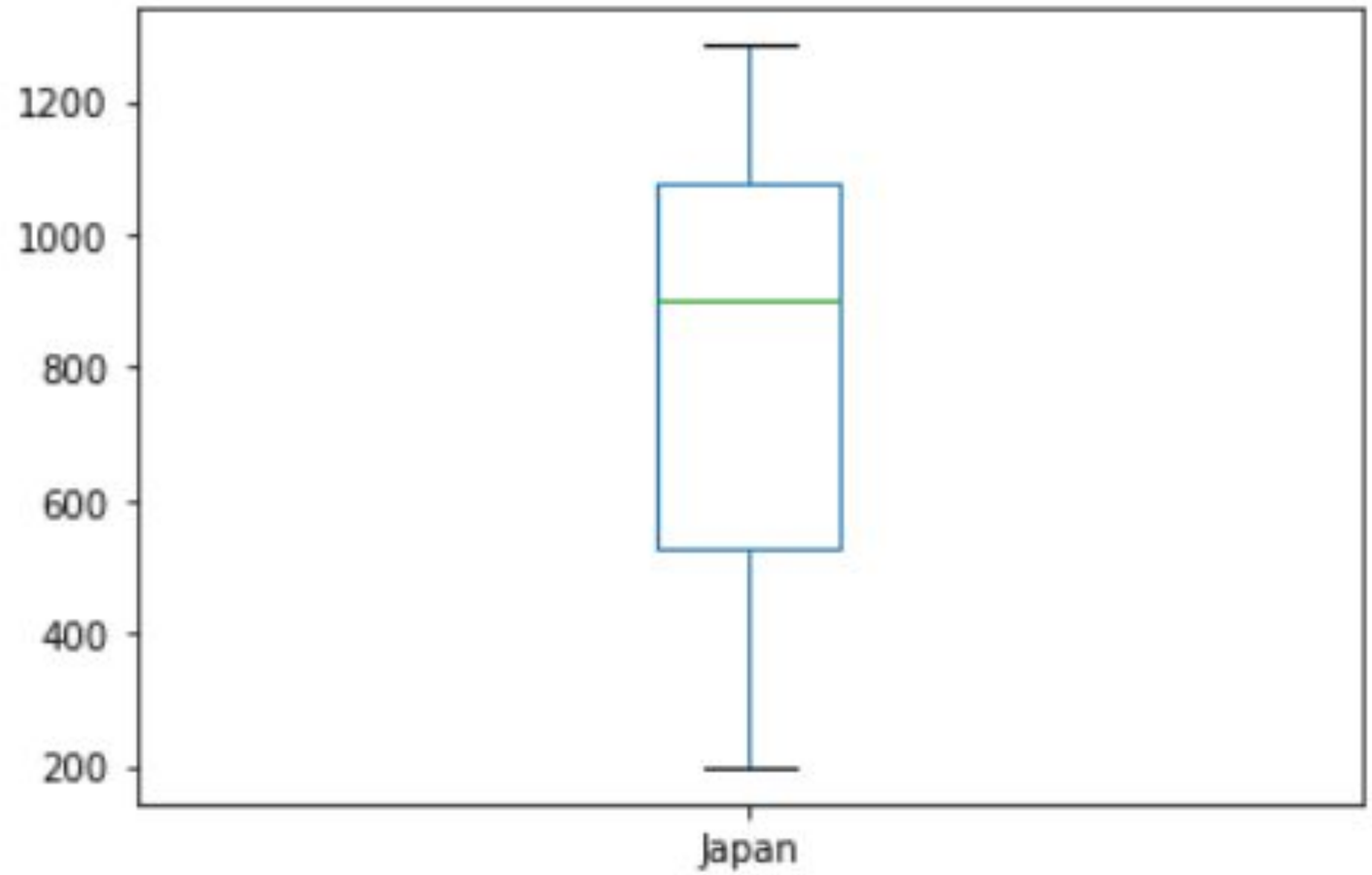
Out[4]:

OdName	Japan
1980	701
1981	756
1982	598
1983	309
1984	246

Boxplot Example - Cell 5

```
In [5]: df_japan.plot(kind='box')
```

```
Out[5]: <AxesSubplot:>
```

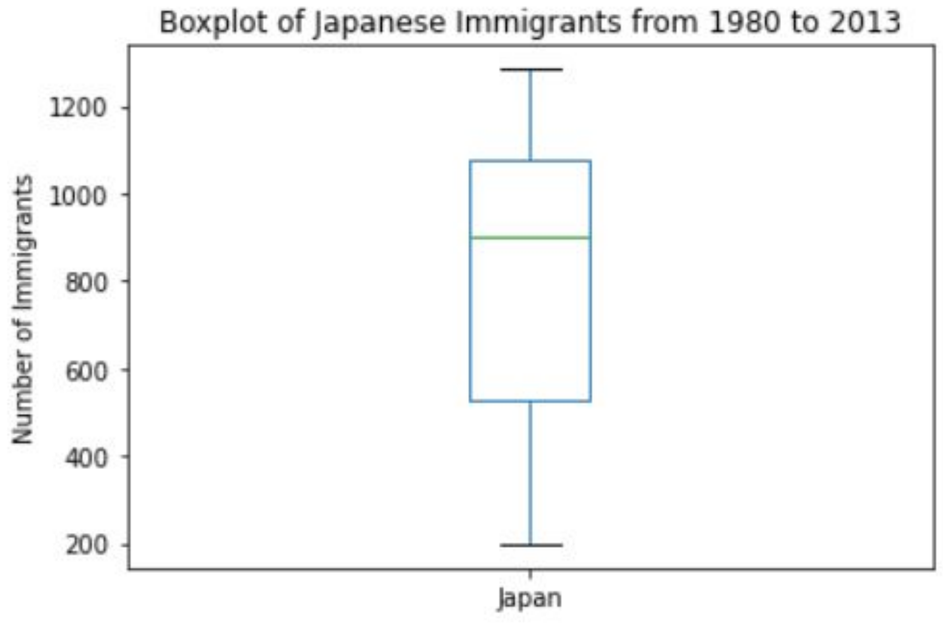


Boxplot - Complete Example

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

df0 = pd.read_csv('canada-mig-dataset.csv')
df1 = df0.set_index('OdName')
df2 = df1.loc[ ['Japan'], list(map(str, range(1980,2014))) ]
df_japan = df2.transpose()
df_japan.plot(kind='box')

plt.title('Boxplot of Japanese Immigrants from 1980 to 2013')
plt.ylabel('Number of Immigrants')
plt.show()
```



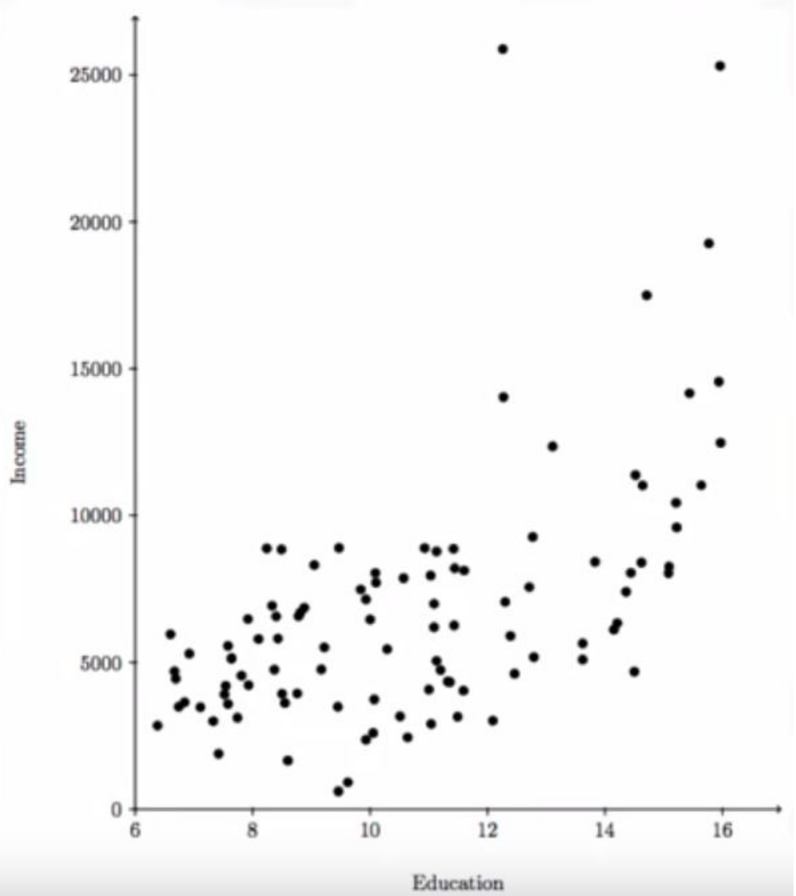
Scatter Plot

Scatter Plot Examples

- **`Usually`**
  - a **dependent variable** to be plotted against an **independent variable**
  - in order to determine if any **correlation** between the two variables exists

- `Scatter plot of income versus education`
  - individual with more education years is likely to earn higher

- `Scatter plot of immigration`
  - clearly depicts an overall rising trend
  - of immigration with time



Scatter Plot Example - Cell 1

In [1]:
```
import pandas as pd

df = pd.read_csv('canada-mig-dataset.csv')

df.head()
```

Out[1]:

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

5 rows × 43 columns

Scatter Plot Example - Cell 2

In [2]:
```
df1 = df0.iloc[:, 9:43]
df1.head()
```

Out[2]:

	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	...	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
0	16	39	39	47	71	340	496	741	828	1076	...	2978	3436	3009	2652	2111	1746	1758	2203	2635	2004
1	1	0	0	0	0	0	1	2	2	3	...	1450	1223	856	702	560	716	561	539	620	603
2	80	67	71	69	63	44	69	132	242	434	...	3616	3626	4807	3623	4005	5393	4752	4325	3774	4331
3	0	1	0	0	0	0	0	1	0	1	...	0	0	1	0	0	0	0	0	0	0
4	0	0	0	0	0	0	2	0	0	0	...	0	0	1	1	0	0	0	0	1	1

5 rows × 34 columns

Scatter Plot Example - Cell 3

In [3]:
```
df1.loc["Total"] = df1.sum(axis=0)
df1.tail()
```

Out[3]:

	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	...	2004	2005	2006	2007	2008	2009	2010	2011
192	1	2	1	6	0	18	7	12	7	18	...	124	161	140	122	133	128	211	160
193	11	17	11	7	16	9	15	23	44	68	...	56	91	77	71	64	60	102	69
194	72	114	102	44	32	29	43	68	99	187	...	1450	615	454	663	611	508	494	434
195	44000	18078	16904	13635	14855	14368	13303	17304	22279	27118	...	3739	4785	4583	4348	4197	3402	3731	2554
Total	143137	128641	121175	89185	88272	84346	99351	152075	161585	191550	...	235822	262242	251640	236753	247244	252170	280687	248748

5 rows × 34 columns

Scatter Plot Example - Cell 4

In [4]:
```
df2 = df1.tail(1)
df2.head()
```

Out[4]:

	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	...	2004	2005	2006	2007	2008	2009	2010	2011
Total	143137	128641	121175	89185	88272	84346	99351	152075	161585	191550	...	235822	262242	251640	236753	247244	252170	280687	248748

1 rows × 34 columns

Scatter Plot Example - Cell 5

In [5]:
```
df3 = df2.transpose()
df3.head()
```

Out[5]:

	Total
1980	143137
1981	128641
1982	121175
1983	89185
1984	88272

Scatter Plot Example - Cell 6

In [6]:
```
df3.reset_index(inplace=True)
df3.head()
```

Out[6]:

	index	Total
0	1980	143137
1	1981	128641
2	1982	121175
3	1983	89185
4	1984	88272

Scatter Plot Example - Cell 7

In [7]:
```
df3.columns = ['Year', 'Total']
df3.head()
```

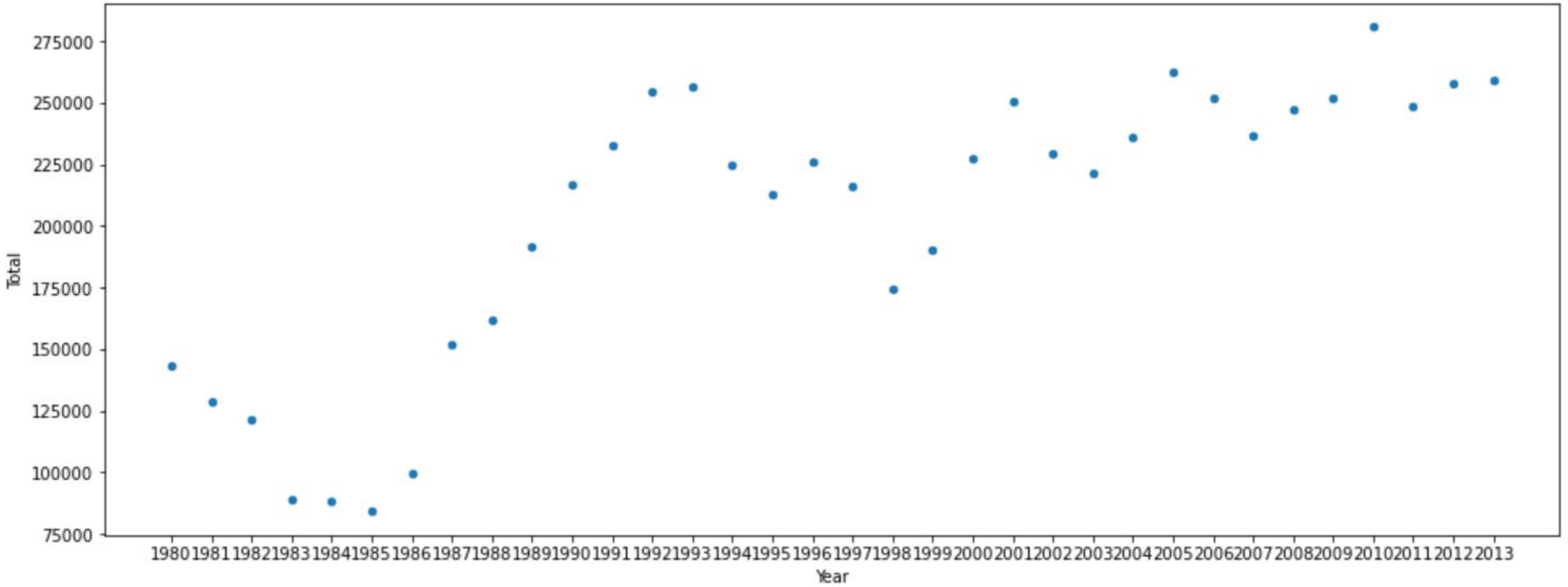
Out[7]:

	Year	Total
0	1980	143137
1	1981	128641
2	1982	121175
3	1983	89185
4	1984	88272

Scatter Plot Example - Cell 8

```
In [8]: df3.plot(kind='scatter', y='Total', x='Year', figsize=(16, 6))
```

```
Out[8]: <AxesSubplot:xlabel='Year', ylabel='Total'>
```



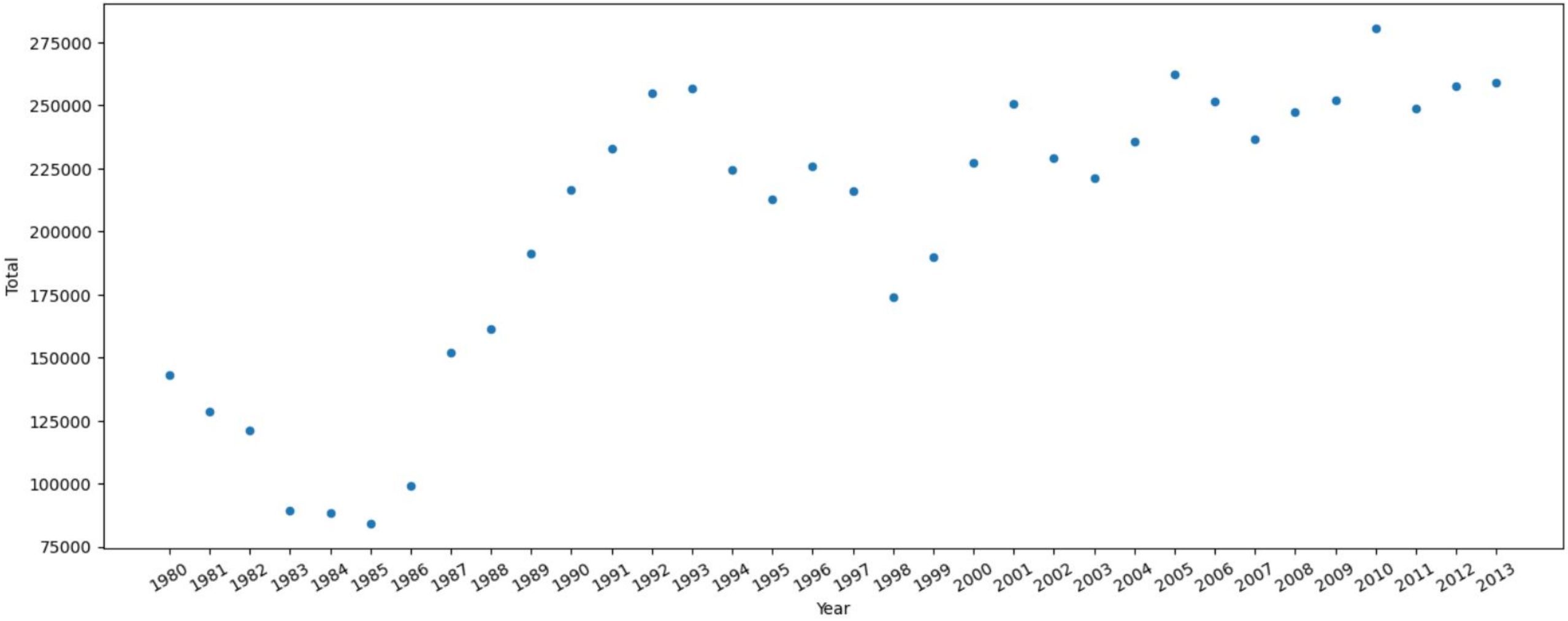
Scatter Plot - Complete Example

```
import pandas as pd

df0 = pd.read_csv('canada-mig-dataset.csv')
df1 = df0.iloc[:, 9:43]
df1.loc["Total"] = df1.sum(axis=0)
df2 = df1.tail(1)
df3 = df2.transpose()
df3.reset_index(inplace=True)
df3.columns = ['Year', 'Total']
df3.plot(kind='scatter', y='Total', x='Year', figsize=(16, 6), rot=30);
```

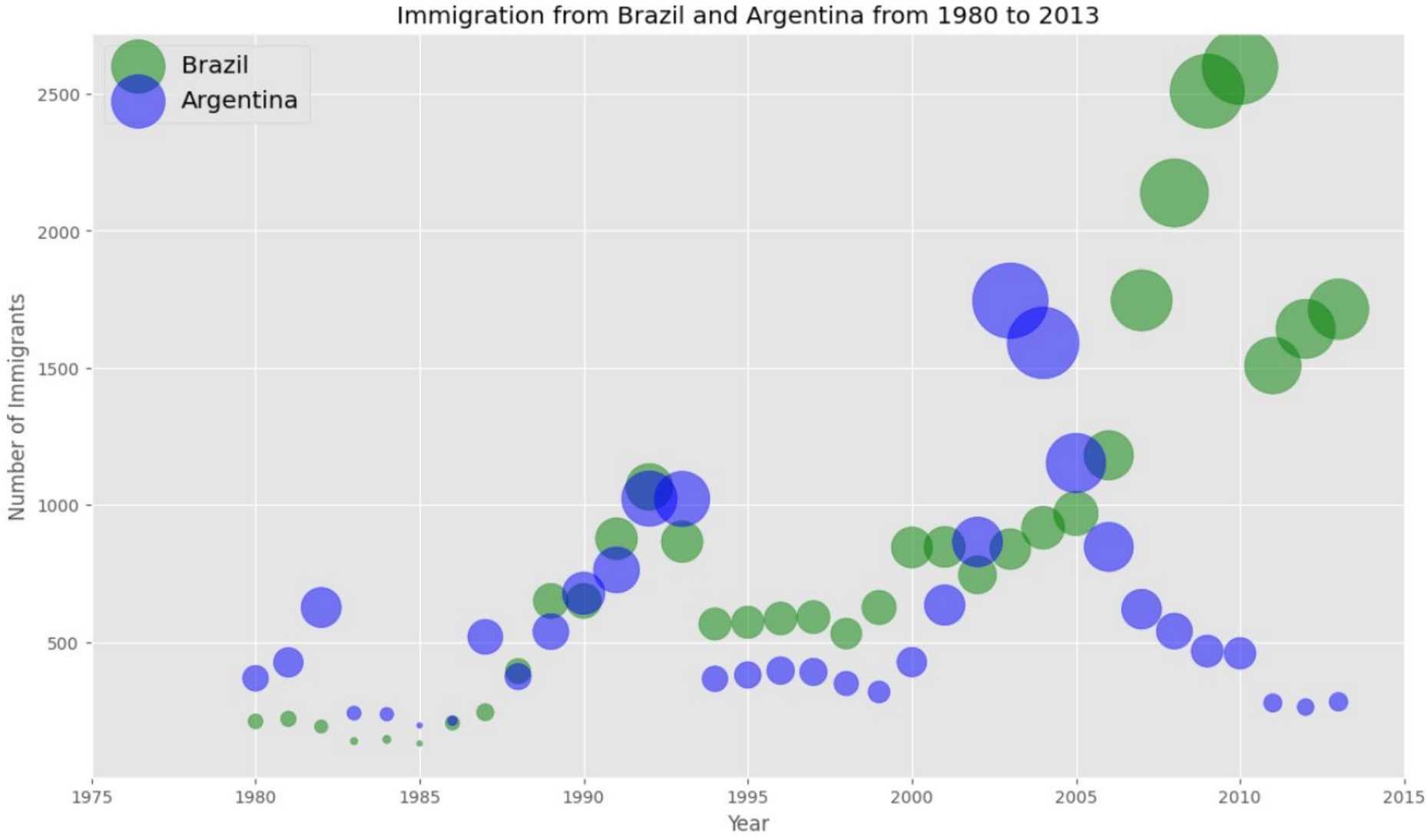
Note the difference in the last line

Scatter Plot - Complete Example Output



Bubble Plot

- A very interesting variation of the scatter plot



Pie Chart Enhancement

Pie Chart Example - Enhancement with Seaborn

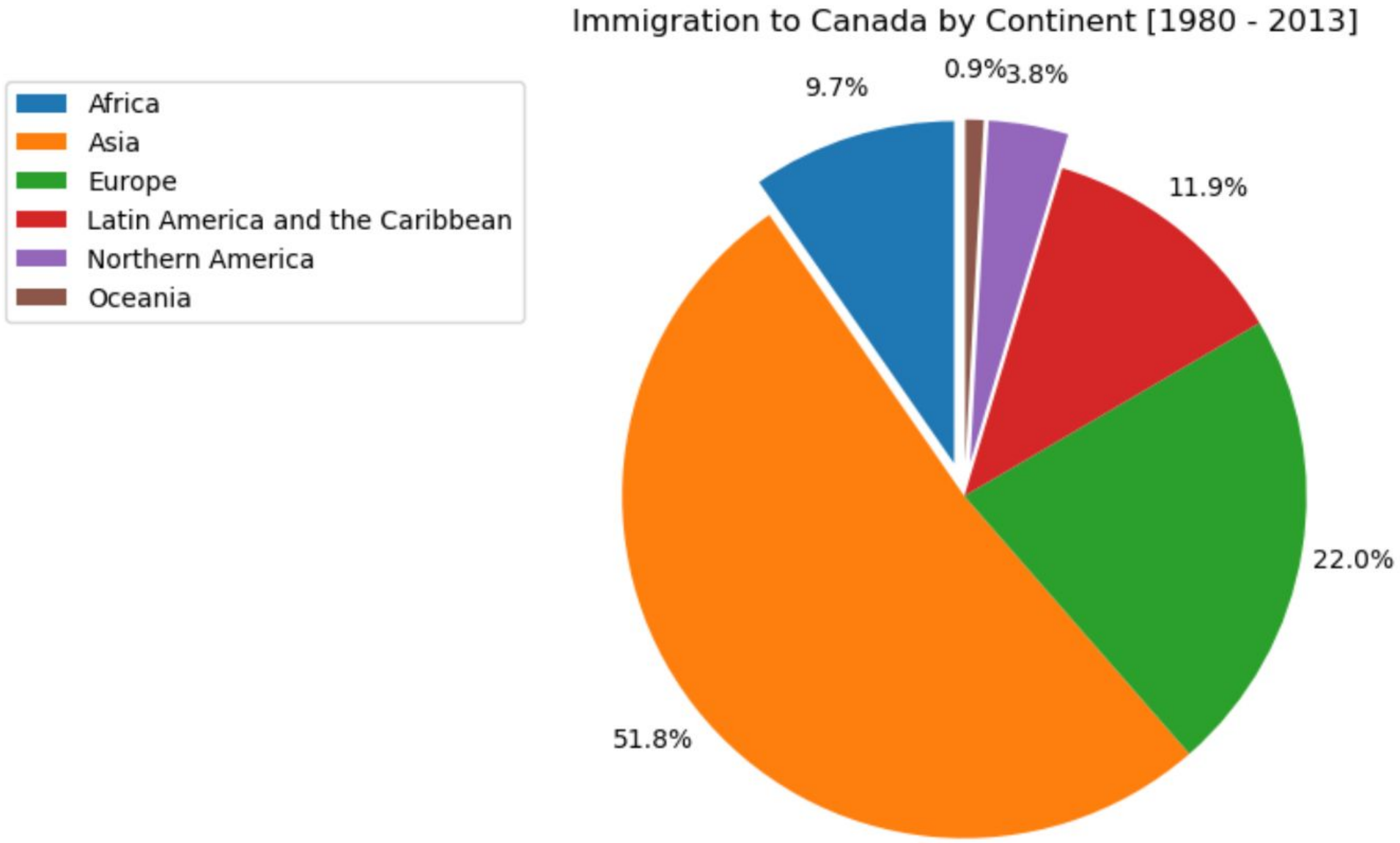
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb

df0 = pd.read_csv('canada-mig-dataset.csv')
df0['Total'] = df0.iloc[:, 9:43].sum(axis=1)
df1 = df0.groupby('AreaName', axis = 0).sum()
df2 = df1.head(6)

colors_list = sb.color_palette()[:df2.shape[0]]
explode_list = [0.1, 0, 0, 0, 0.1, 0.1] # ratio for each continent with which to offset each wedge

df2['Total'].plot(kind='pie',
            colors = colors_list,   # Add custom colors
            explode = explode_list, # 'Explode' the lowest 3 continents
            figsize = (10, 6),      # A tuple (width, height) in inches represents the size of a Figure object
            startangle = 90,        # Start angle: 90° (Africa)
            labels = None,          # Turn off labels
            pctdistance = 1.15,     # Ratio between center of each slice and start of text generated by autopct
            autopct = '%.1f%%',     # '%.1f' to show one float point while '%%' to show '%' (double '%%' to skip)
        )
plt.title('Immigration to Canada by Continent [1980 - 2013]', pad=15) #Title offset from the top of the axes in points
plt.legend(labels = df2.index, bbox_to_anchor = (0, 1) )
plt.ylabel("")
plt.show()
```

Pie Chart Example - Enhancement with Seaborn



Questions

Links

https://github.com/fcai-b/dv

References

1. https://www.coursera.org/learn/python-for-data-visualization