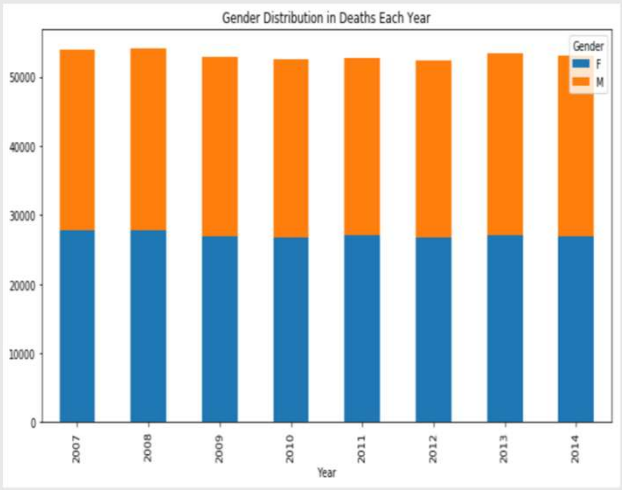
Data Visualization

- Presentation of data and analysis with **pictures** or **graphs/charts**
- Enables the visual presentation of analytics
  - easy to grasp difficult concepts
  - can identify new patterns
- Caters to the way, human brain processes information
  - using charts or graphs to visualize large amounts of complex data is **easier** than scanning over spreadsheets or reports
- **Very** important part of data analysis

Analysis Result - Table vs Chart

Year	Gender	
2007	F	27749.0
	M	26247.0
2008	F	27816.0
	M	26322.0
2009	F	26941.0
	M	25879.0
2010	F	26675.0
	M	25830.0
2011	F	27075.0
	M	25651.0
2012	F	26766.0
	M	25654.0
2013	F	27133.0
	M	26254.0
2014	F	26916.0
	M	26090.0



Data Visualization Python Libraries

- Python has several data visualization libraries to create very simple to very complex visualizations
- Some of them are:
  - Matplotlib
  - Seaborn
  - Bokeh
  - Plotly
  - GGplot

Matplotlib

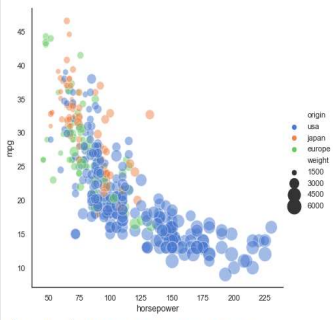
- Very flexible 2-D plotting library in Python
- Supports all the popular charts (lots, histograms, power spectra, bar charts, error charts, scatterplots, etc)
- There are extensions that can be used to create advanced visualizations like 3-D plots
- Can be customized at every level

Important links:

- https://matplotlib.org/3.1.0/gallery/index.html
- matplotlib official documentation (https://matplotlib.org/users/index.html)

Seaborn

- Used to create **beautiful** visualizations
- very less coding is needed to make high-grade visualizations
- it supports lots of advanced plots like categorical plotting (catplot), distribution plotting using kde (distplot), swarm plot, etc.
- built on top of matplotlib, it is highly compatible with it
  - can start with advanced plots that seaborn already supports
  - customize them as much as you want with the help of matplotlib



Python source code: [download source: scatter_bubbles.py]

```
import seaborn as sns
sns.set(style="white")

# Load the example mpg dataset
mpg = sns.load_dataset("mpg")

# Plot miles per gallon against horsepower with other semantics
sns.relplot(x="horsepower", y="mpg", hue="origin", size="weight",
            sizes=(40, 400), alpha=.5, palette="muted",
            height=6, data=mpg)
```

Bokeh

- generate visualizations that are friendly on the web interface and browsers
- supports unique visualizations like Geospatial plots, Network graphs, etc.
- If displaying these visualizations in a browser,
  - there are options available to export them
  - can also be used through **JavaScript**
- Important links:
  - Bokeh official documentation (https://docs.bokeh.org/en/latest/index.html)

Example

- Texas Unemployment 2009 map https://docs.bokeh.org/en/latest/docs/gallery/texas.html

Plotly

- Used to create **interactive** visualizations
- Compatible with Jupyter Notebook and Web Browsers
- Supports many types of plots
  - basic charts, Seaborn-like beautiful and advanced plots, 3-D plots, Map-based visualizations, scientific plots, etc.
- Supports animation capabilities
  - data story can be told through visualizations
- Important links
  - plotly documentation (https://plotly.com/python/)

Demo

- https://cdn.analyticsvidhya.com/wp-content/uploads/2020/03/687474703a2f2f692e696d6775722e636f6d2f6433793346e776d2e676966.gif

ggplot

- Python version of the **ggplot2 of R** and the **Grammer of Graphics language**
- Tightly coupled with Pandas
  - easily build visualizations using Pandas dataframe itself
- Important links
  - ggplot documentation (https://yhat.github.io/ggpy/)

References

- For different plotting libraries in python
  - https://www.analyticsvidhya.com/blog/2020/03/6-data-visualization-python-libraries/