Fourth Industrial Summer School

Day 1: Data Analysis Foundations – Afternoon

Data Plotting and Visualization

Session Objectives

✓ Exploratory Data Analysis (EDA)

✓ Plotting Libraries
- Matplotlib
- Panadas

✓ Charts
- Scatter plot
- Histograms
- Pie charts
- Box plots

✓ Multiple Figures

Exploratory Data Analysis (EDA)

- A process to look at the data and understand it
  - You need to do EDA at the beginning and end of your project.
- Insight of your data
  - Creating visualizations will help you to understand your data
- Generate Hypothesis
- Results presentation
  - Clear and concise results presentation to your audience

Descriptive Statistics

"You cannot control what you cannot measure."

--Tom DeMarco

Data Types

Categorical data

- Nominal
- Ordinal

Continuous data

- Interval
- Ratio

Categorical - Descriptive statistics

- Nominal
  - Mode
  - Percentage (%)

- Ordinal
  - Nominal +
  - Median
  - Interquartile range

Mode

- The *mode represents the most commonly occurring sample.*

- The mode is well defined if there is only one value that is more common than all others are.

- The mode value is meaningful for the nominal, ordinal, interval and ratio scales.

- As an example we may compute the mode for the data set
  - (1, 1, 2, 4) giving a mode of 1
  - (female, male, male) giving a mode of male

Median

- The *median, denoted* $\tilde{x}$ *represents the middle value of a data set.*

- The median is calculated by sorting the samples in ascending (or descending) order and picking the middle sample.

$$position\ of\ the\ median = \frac{n+1}{2}$$

  - This is well defined if n is odd. If n is even, the median may be defined as the arithmetic mean of the two middle values.

- As an example, we may compute the median for the data set:
  - (1, 1, 2, 3, 4) resulting in $\tilde{x}$ =2
  - (1, 1, 2, 4) resulting $\tilde{x}$ = 1.5

Continuous - Descriptive statistics

- Summarize
  - Mean, median, and mode
    - Interval. Athematic mean
    - Ratio. Geometric mean
  - Standard deviation, interquartile range, and range

Arithmetic Mean

- The (arithmetic) *mean* is calculated as:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- The mean value is meaningful for the interval and ratio scales.
- In most of the cases, it is used when no significant outliers exist.
- Example
  - we may compute the mean for the data set (1, 1, 2, 4) resulting in $\bar{x}$ = 2.0

Geometric Mean

- The Geometric *mean* is the n^{th} root of the product of n numbers.
  - That means you multiply a bunch of numbers together, and then take the nth root, where n is the number of values you just multiplied

$$\sqrt[n]{\prod_{i=1}^{n} x_i}$$

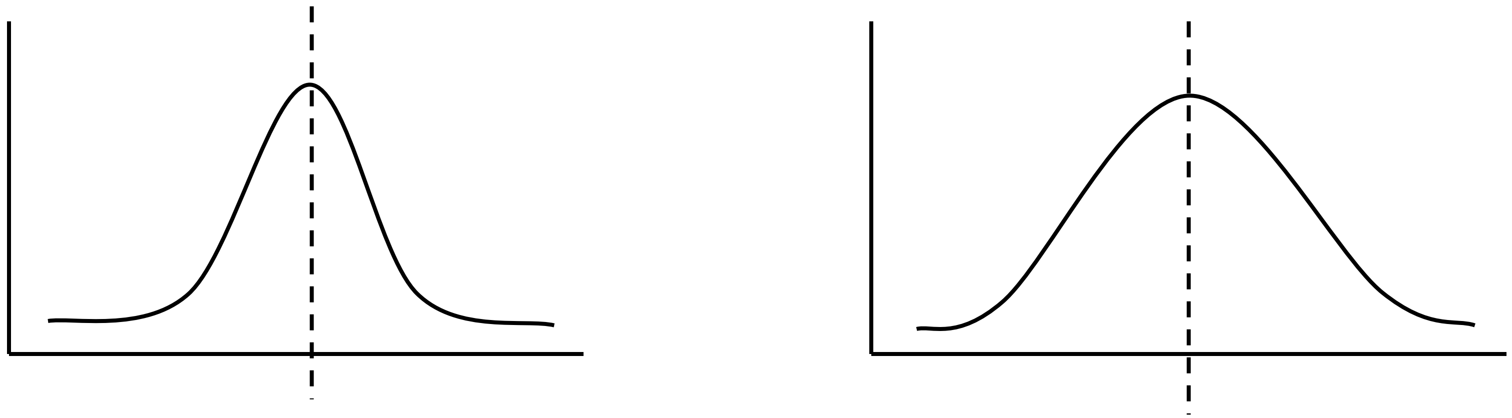
- The geometric mean is well defined if all samples are non-negative and meaningful for the ratio scale.

- Example
  - What is the geometric mean of 2, 8 and 4?

Variability (dispersion)

- Measures the level of variation from the central tendency, i.e. to see how spread or concentrated the data is.

- Variability is usually defined in terms of distance
  - How far apart scores are from each other
  - How far apart scores are from the mean
  - How representative a score is of the data set as a whole



Variability (dispersion) measurement

- Sample variance is the mean of the square distance from the sample mean.

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

- Standard deviation is the measure of the *standard distance* from the mean

Standard deviation = $\sqrt{variance}$

- Range of a data set is the distance between the maximum and minimum data value:

range = xmax - xmin

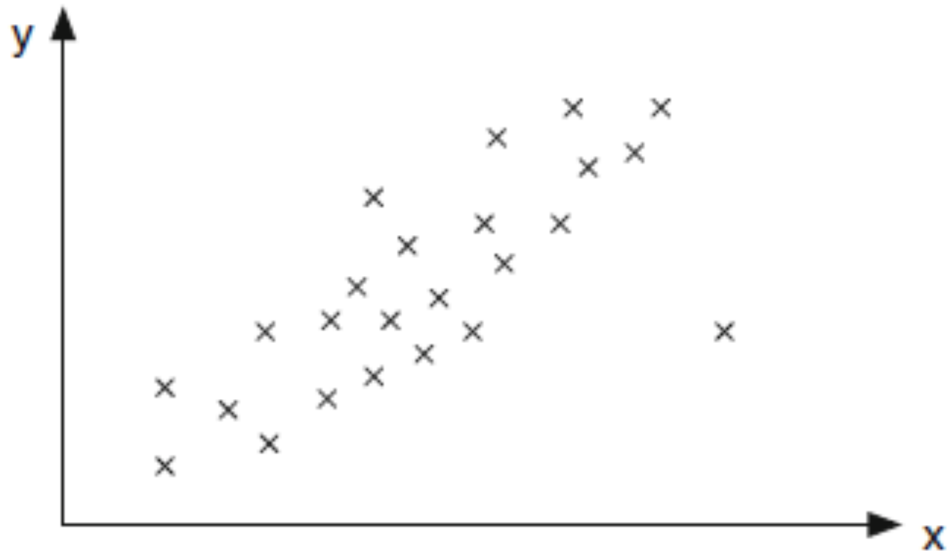
Graphical Visualization

Graphical Visualization

- When describing a data set, quantitative measures of central tendency, dispersion, and dependency, should be combined with graphical visualization techniques.
- Graphs are very illustrative and give an overview of the data.
- One simple but effective graph is the **scatter plot**, *where pairwise samples* (x_i, y_i) are plotted in two dimensions.
- The scatter plot is good for assessing dependencies between variables.
- By examining the scatter plot, it can be seen how spread or concentrated the data points are

Scatter plot

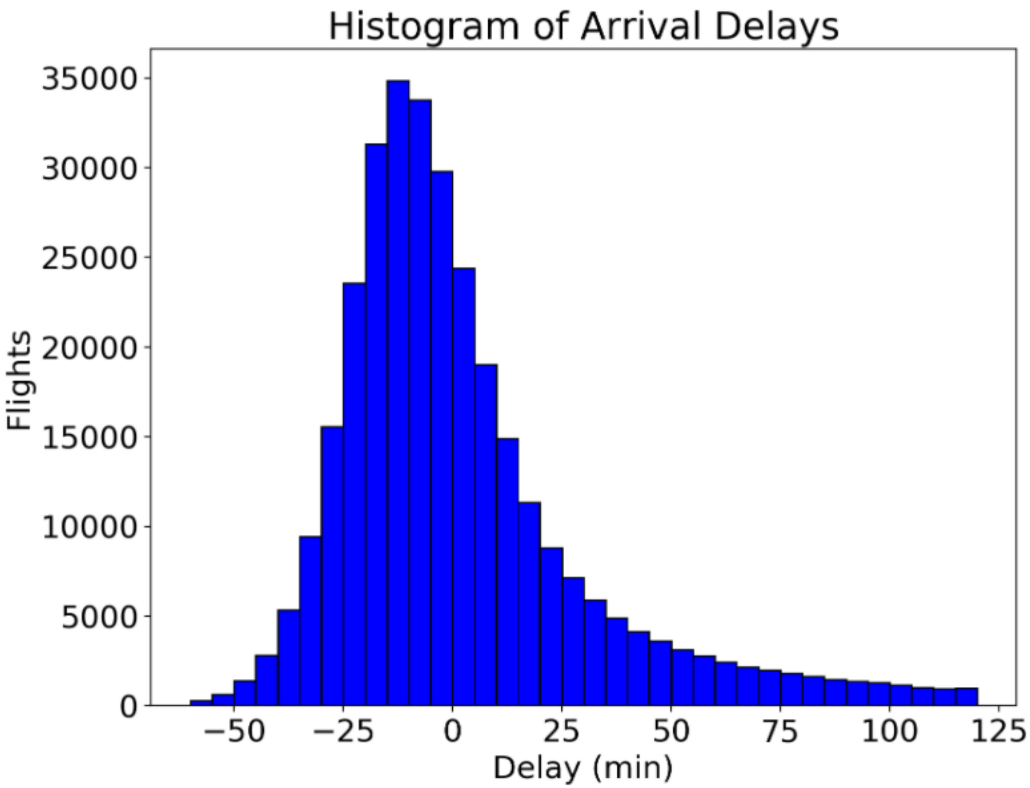
- In this scatter plot there is a linear tendency with a positive correlation



Histograms

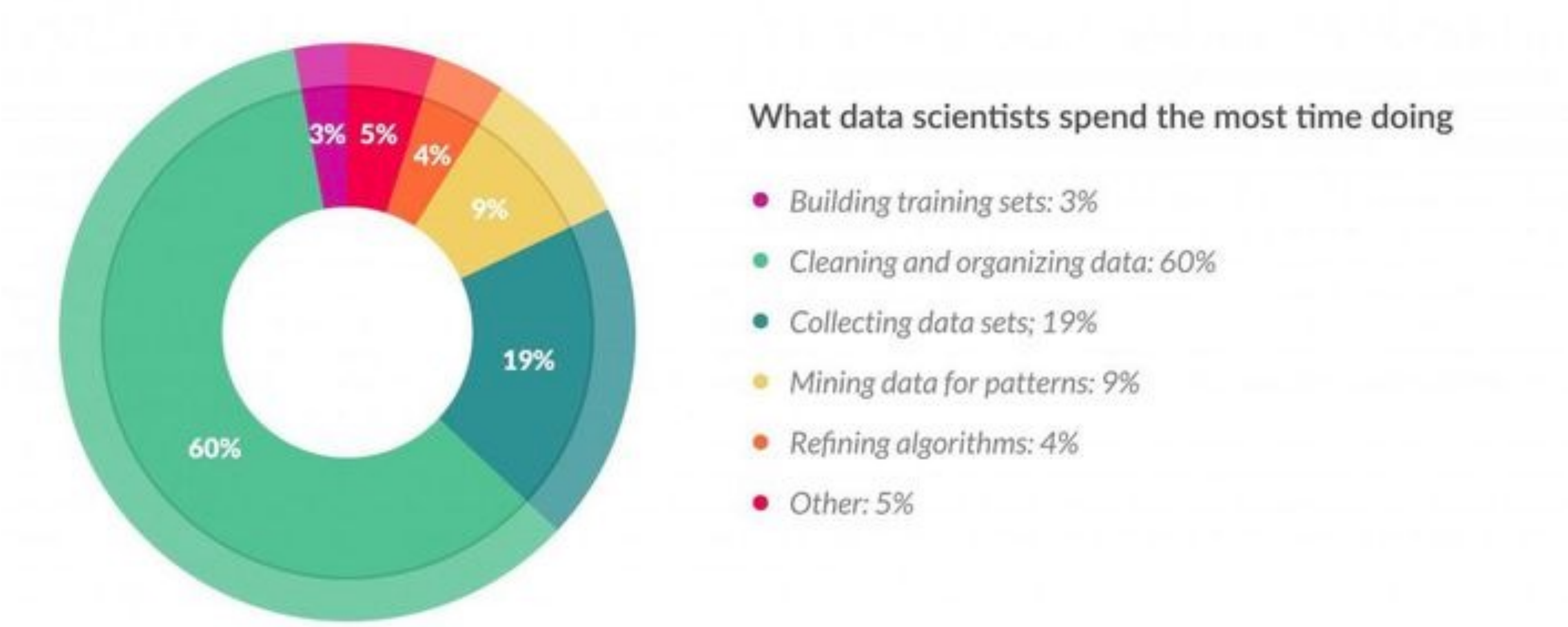
- The **histogram** *can be used to give an overview of the* ***distribution density*** *of the* samples from one variable.
- A histogram consists of bars with heights that represent the frequency (or the relative frequency) of a value or an interval of values.
- The histogram is thus a graphical representation of a frequency table.
- A plot could provide a first indication whether the data resembles a normal distribution or not.
- The cumulative histogram may be used to give a picture of the probability distribution function of the samples from one variable.

Histograms


Histogram of Arrival Delays
Flights
35000
30000
25000
20000
15000
10000
5000
0
−50
−25
0
25
50
75
100
125
Delay (min)

Pie Chart

- A **pie chart** shows the relative frequency of the data values divided into a specific number of distinct classes, by constructing segments in a circle with angles proportional to the relative frequency

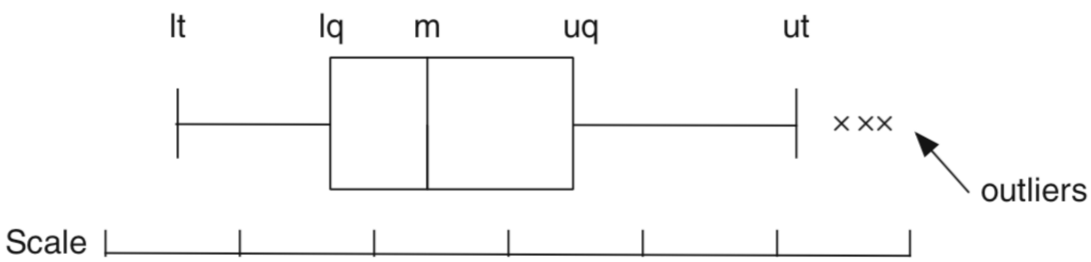


Box plot

- A **box plot** good for visualizing the dispersion and skewedness of samples.

- We can use boxplots:
  - Examine how the data is dispersed
  - Investigate outliers
  - Signs of skewness
  - Compare between groups

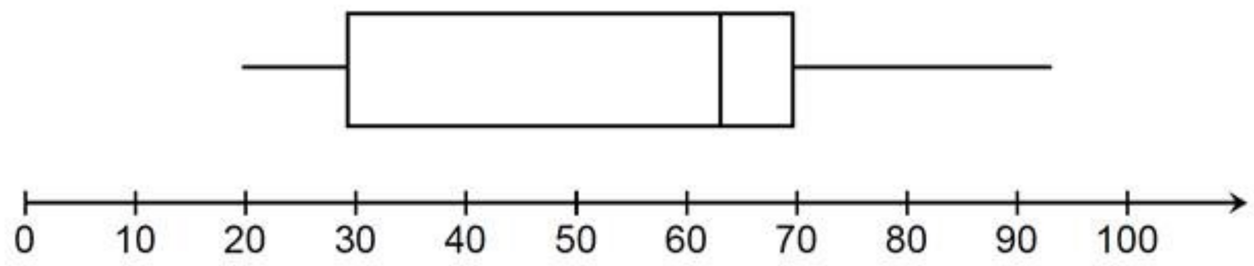
Box plot



- The middle bar in the box m, is the median.
- The lower quartile lq, is the 25% percentile (the median of the values that are less than m)
- The upper quartile uq is the 75% percentile (the median of the values that are greater than m)
- The length of the box is d = uq – lq
- *The upper tail ut is uq(q3) + 1.5d [or max]*
- *The lower tail lt is lq(q1) – 1.5d [or min]*
- Values outside the lower and upper tails are called **outliers**, and are shown explicitly in the box plot

Box plot

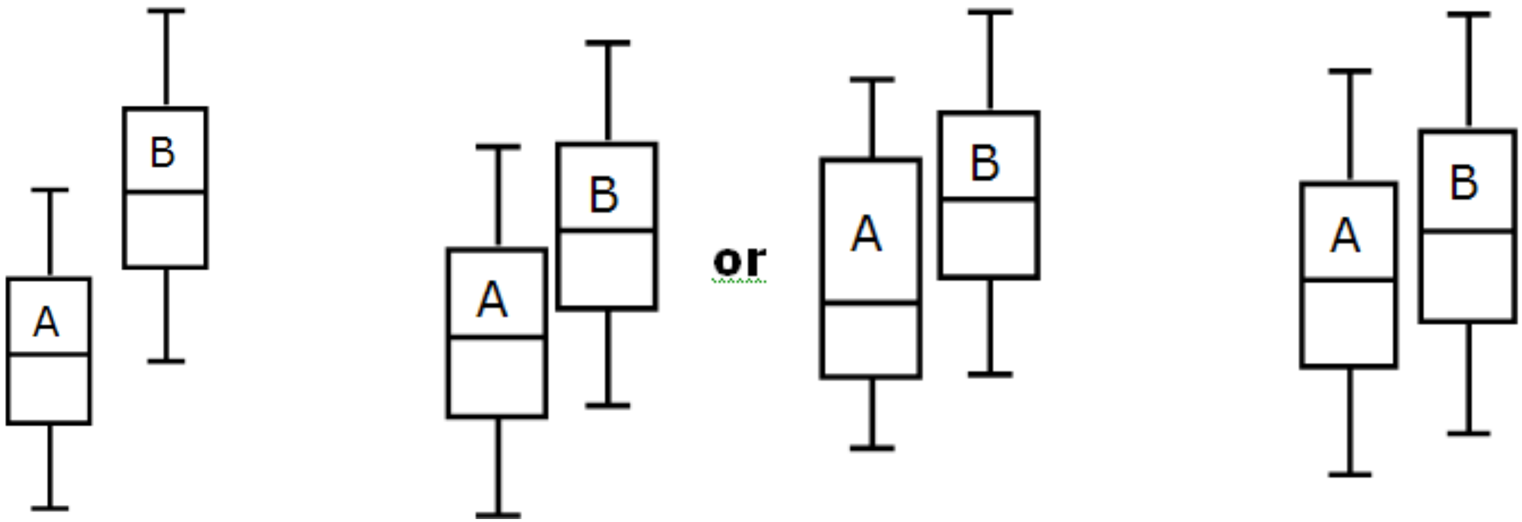
- Programming quiz scores for a class



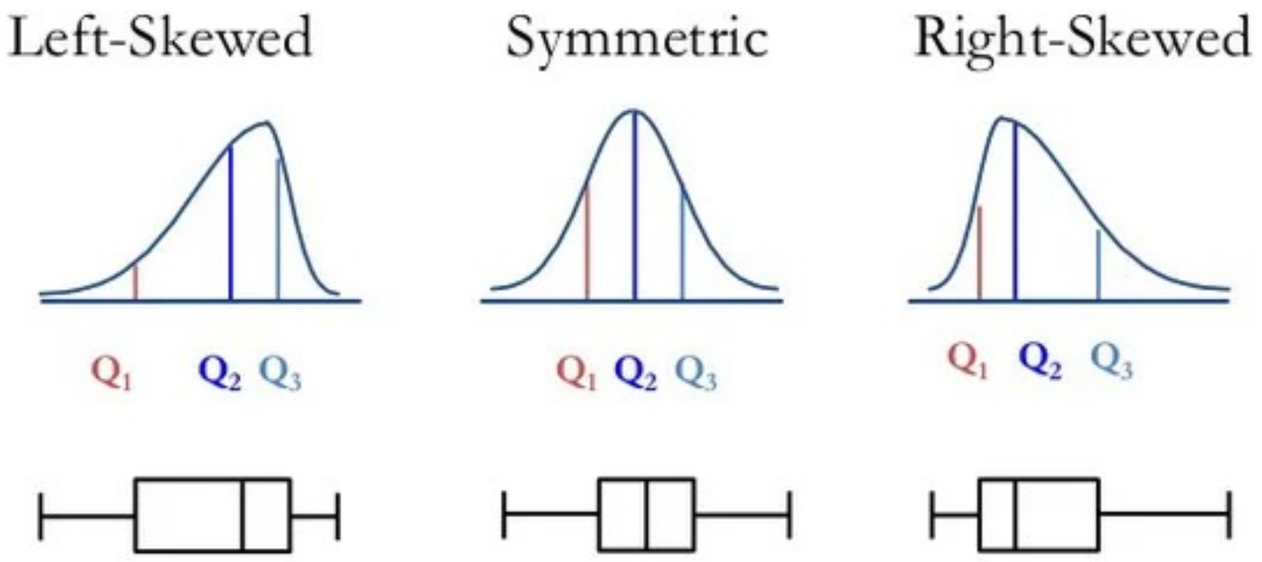
- How to read this box plot?

Box plot

- Comparing box plots

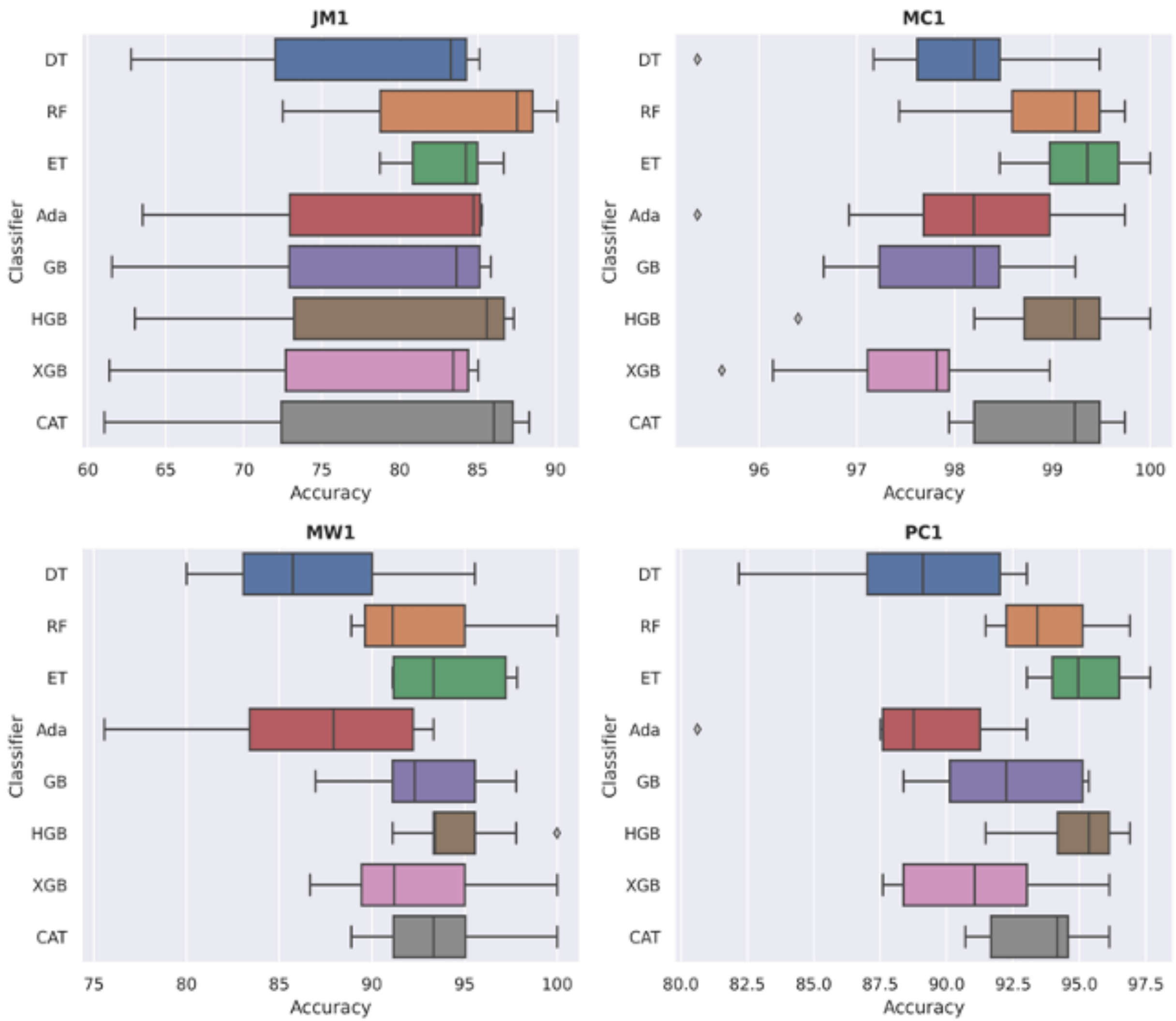


Box plot - Skewness



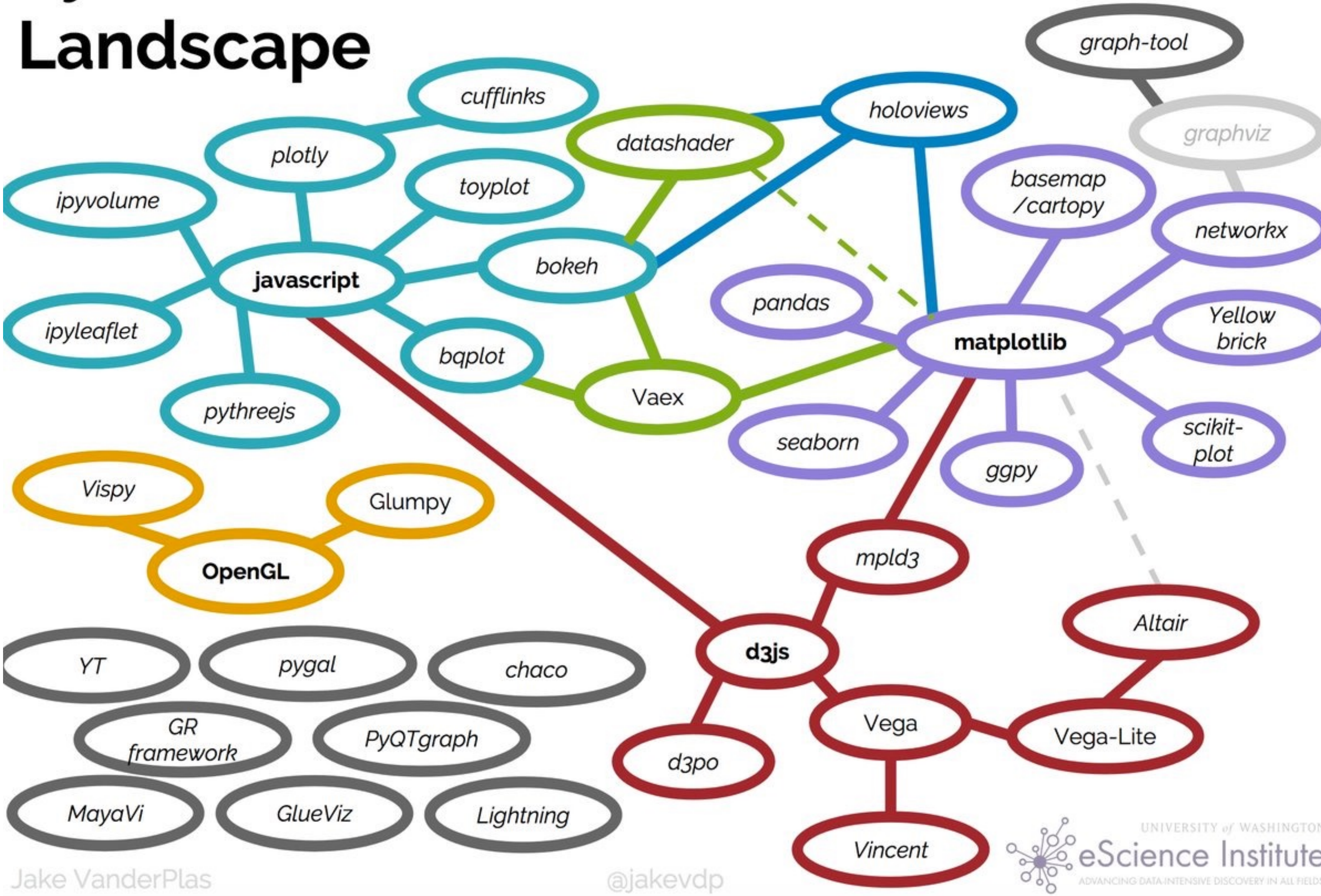
https://www.simplypsychology.org/boxplots.html

Software Defect Prediction using Tree-Based Ensembles



Plotting Libraries

Python's Visualization Landscape



Plotting Libraries

- Python popular plotting libraries
  - Matplotlib
  - Pandas
  - Seaborn
  - ggplot
  - Plotly

Matplotlib

- Matpoltlib is a very popular Python library for data visualization.

- Like Pandas, it is not directly related to Machine Learning.

- It particularly used to visualize the patterns in the data.

- It is a 2D plotting library used for creating 2D graphs and plots.
  - a set of functionalities similar to those of MATLAB
  - line plots, scatter plots, bar charts, histograms, pie charts etc.

Link: https://matplotlib.org/

Seaborn

- Seaborn is a Python data visualization library based on matplotlib.
- It provides a high-level interface for drawing attractive and informative statistical graphics.
- It introduces additional plot types.
- It also makes your traditional Matplotlib plots look a bit prettier.
- Similar (in style) to the popular ggplot2 library in R

Link: https://seaborn.pydata.org/

Pandas

- High performance
- Create plots out of a pandas dataframe and series
- Higher level API than Matplotlib

Link: https://pandas.pydata.org/

Example

- Employee.csv
  - Company employees information

emp_id	Gender	Age	Sales	BMI	Income
1	M	34	123	Normal	350
2	F	40	114	Overweight	450
3	F	37	135	Obesity	169
4	M	30	139	Underweight	189
5	F	44	117	Underweight	183
6	M	36	121	Normal	80
7	M	32	133	Obesity	166
8	F	26	140	Normal	120
9	M	32	133	Normal	75
10	M	36	133	Underweight	40

Import and Read csv file

- First step
  - Import pandas / matplotlib

- Second
  - Read csv file
  - What is the type of data read from the file?

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

df = pd.read_csv('Employee.csv')
type(df)
```

```
pandas.core.frame.DataFrame
```

Data information

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 6 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   emp_id  10 non-null     int64
 1   Gender  10 non-null     object
 2   Age     10 non-null     int64
 3   Sales   10 non-null     int64
 4   BMI     10 non-null     object
 5   Income  10 non-null     int64
dtypes: int64(4), object(2)
memory usage: 608.0+ bytes
```

Show dataframe

```
1   df
```

	emp_id	Gender	Age	Sales	BMI	Income
0	1	M	34	123	Normal	350
1	2	F	40	114	Overweight	450
2	3	F	37	135	Obesity	169
3	4	M	30	139	Underweight	189
4	5	F	44	117	Underweight	183
5	6	M	36	121	Normal	80
6	7	M	32	133	Obesity	166
7	8	F	26	140	Normal	120
8	9	M	32	133	Normal	75
9	10	M	36	133	Underweight	40

Descriptive Statistics

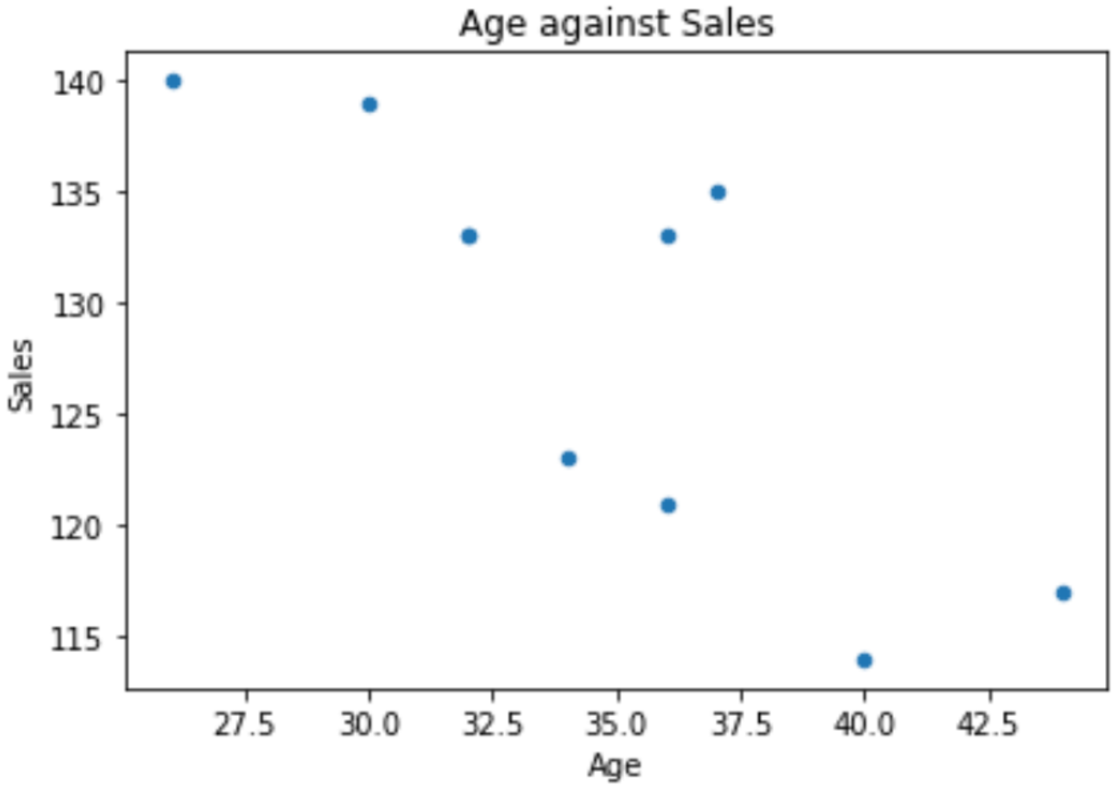
```
1   df.describe()
```

	emp_id	Age	Sales	Income
count	10.00000	10.000000	10.000000	10.000000
mean	5.50000	34.700000	128.800000	182.200000
std	3.02765	5.121849	9.271222	127.533699
min	1.00000	26.000000	114.000000	40.000000
25%	3.25000	32.000000	121.500000	90.000000
50%	5.50000	35.000000	133.000000	167.500000
75%	7.75000	36.750000	134.500000	187.500000
max	10.00000	44.000000	140.000000	450.000000

Scatter plot (Pandas)

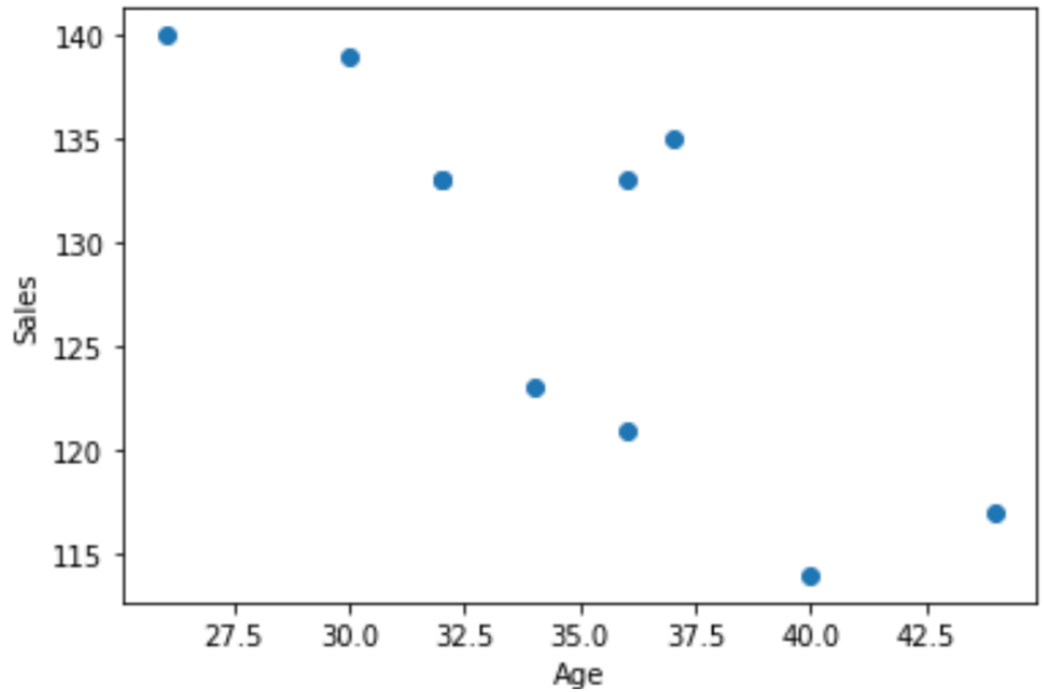
```
scatter_plot = df.plot.scatter(x='Age',y='Sales')
scatter_plot.set_title('Age against Sales')
```

```
Text(0.5, 1.0, 'Age against Sales')
```



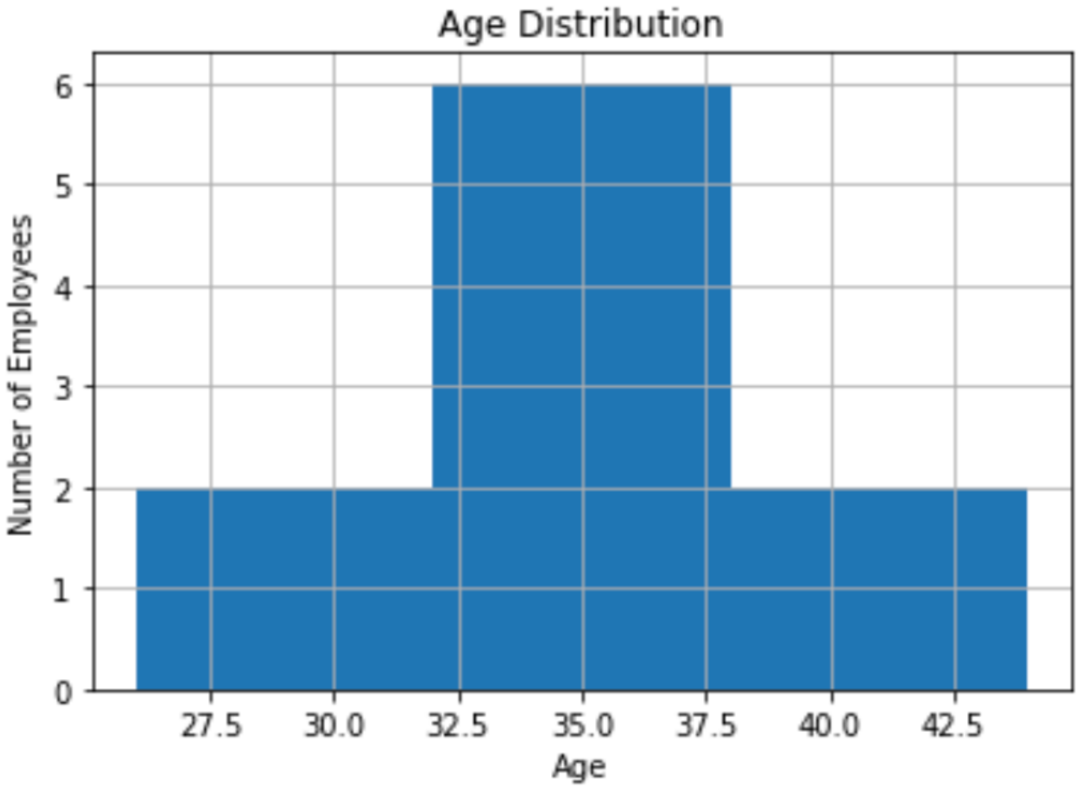
Scatter plot (Matplotlib)

```
age_list = df['Age']
sales_list = df['Sales']
plt.scatter(age_list,sales_list, label = 'Age against Sales')
plt.xlabel('Age')
plt.ylabel('Sales')
plt.show()
```



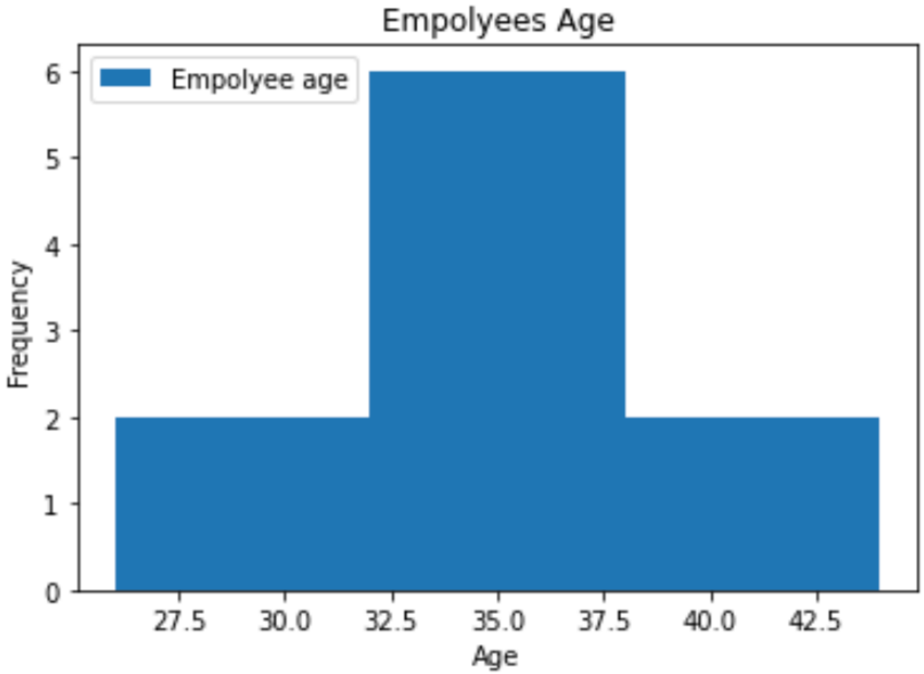
Histogram (Pandas)

```
hist_plot = df['Age'].hist(bins = 3)
hist_plot.set_xlabel('Age')
hist_plot.set_ylabel('Number of Employees')
hist_plot.set_title('Age Distribution')
```



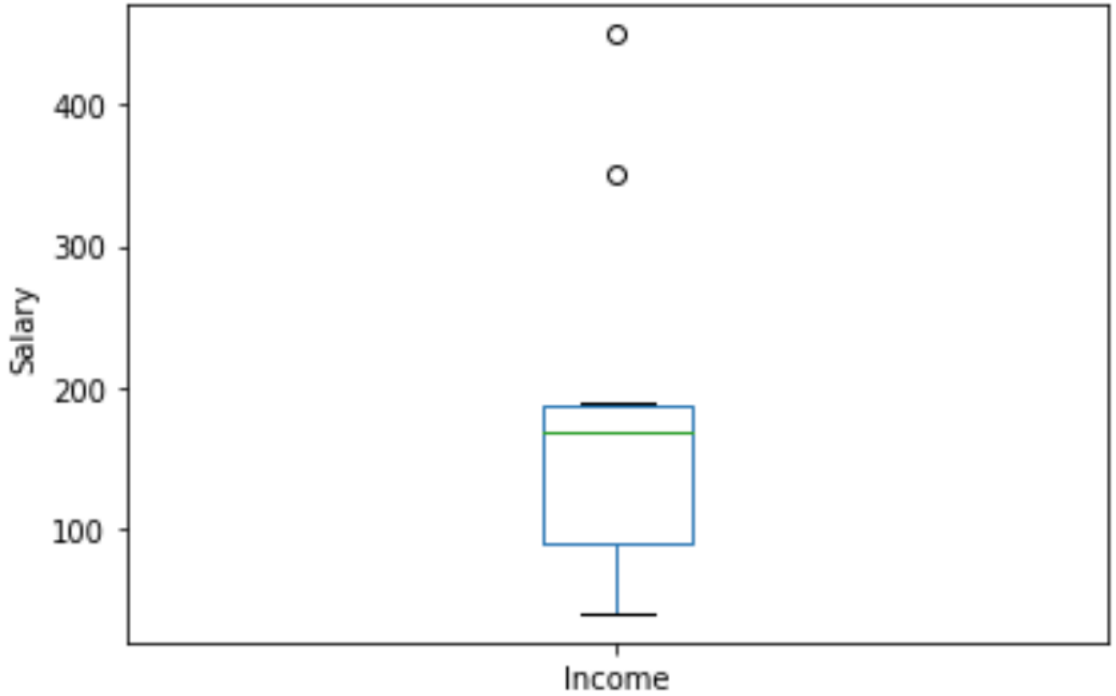
Histogram (Matplotlib)

```
age_list = df['Age']
plt.hist(age_list, label = 'Empolyee age', bins = 3)
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.legend(loc='upper left')
plt.title('Empolyees Age')
plt.show()
```



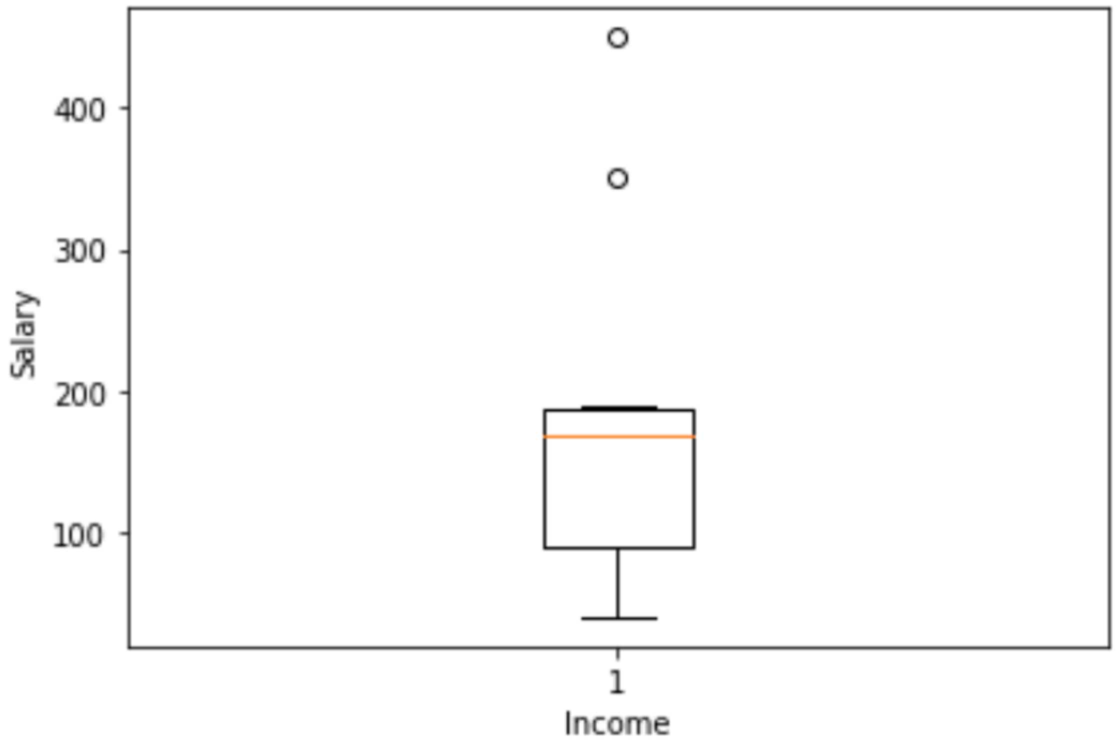
Box plot (Pandas)

```
box = df['Income'].plot.box()
box.set_ylabel('Salary')
```



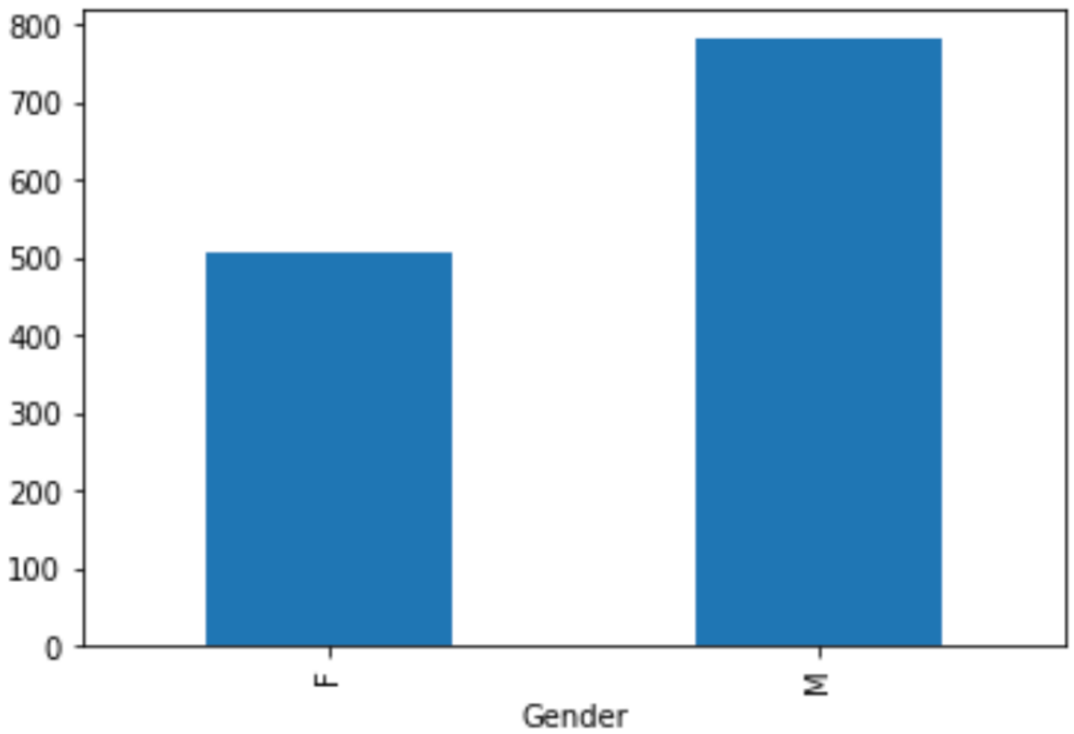
Box plot (Matplotlib)

```
income_list = df['Income']
plt.boxplot(income_list)
plt.xlabel('Income')
plt.ylabel('Salary')
```



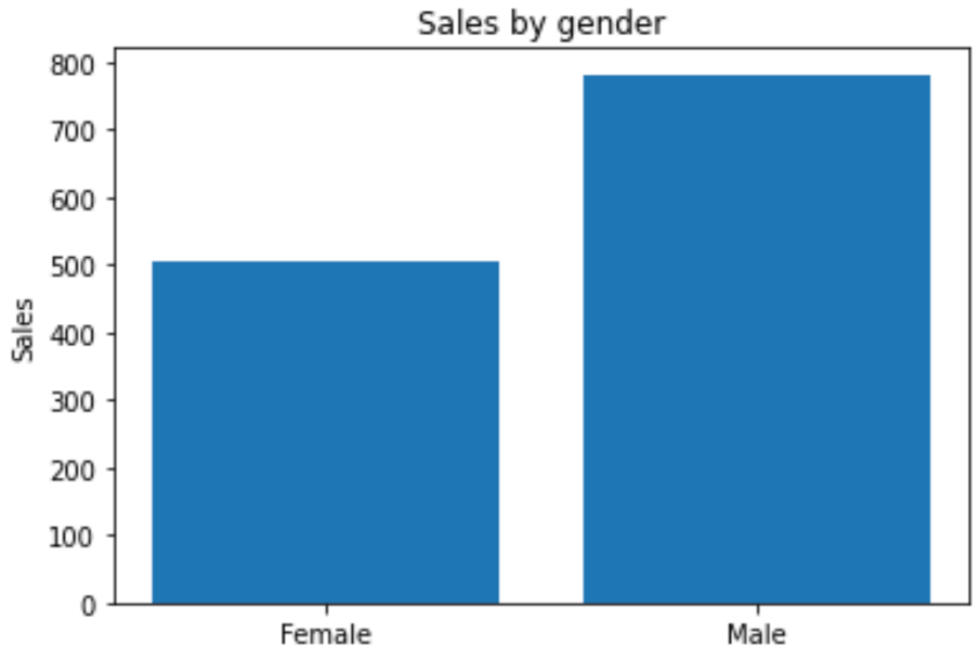
Bar Chart (Pandas)

```
sales = df.groupby('Gender').Sales.sum()
sales.plot.bar()
```



Bar Chart (Matplotlib)

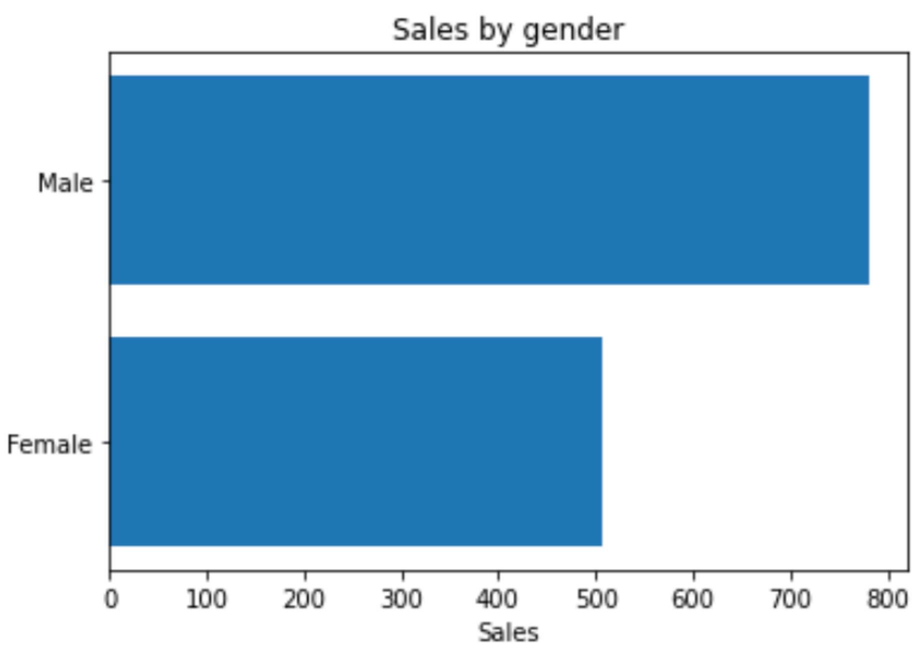
```
labels = ('Female','Male')
y_pos  = np.arange(len(labels))
sales  = df.groupby('Gender').Sales.sum()
plt.bar(y_pos, sales)
plt.xticks(y_pos, labels)
plt.ylabel('Sales')
plt.title('Sales by gender')
plt.show()
```



Bar Chart (Matplotlib)

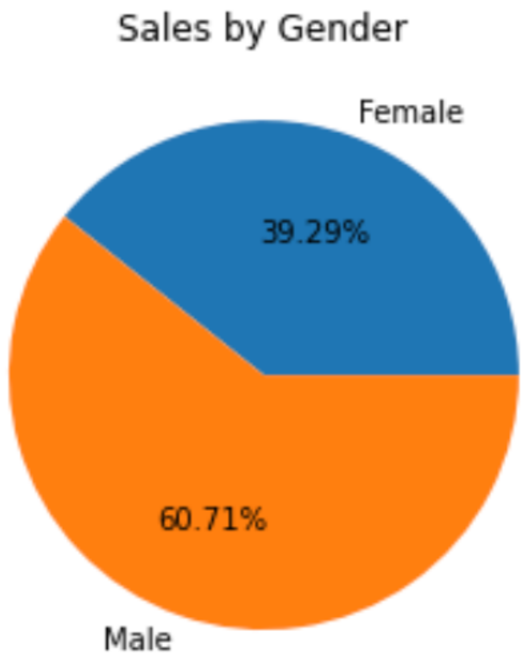
- Horizontal bar
  - barh function

```
labels =('Female', 'Male')
y_pos  = np.arange(len(labels))
sales  = df.groupby('Gender').Sales.sum()
plt.barh(y_pos,sales)
plt.yticks(y_pos,labels)
plt.xlabel('Sales')
plt.title('Sales by gender')
plt.show()
```



Pie chart (Matplotlib)

```
sales = df.groupby("Gender").Sales.sum()
plt.pie(sales, labels=["Female", "Male"], autopct="%.2f%%")
plt.title('Sales by Gender')
plt.show()
```

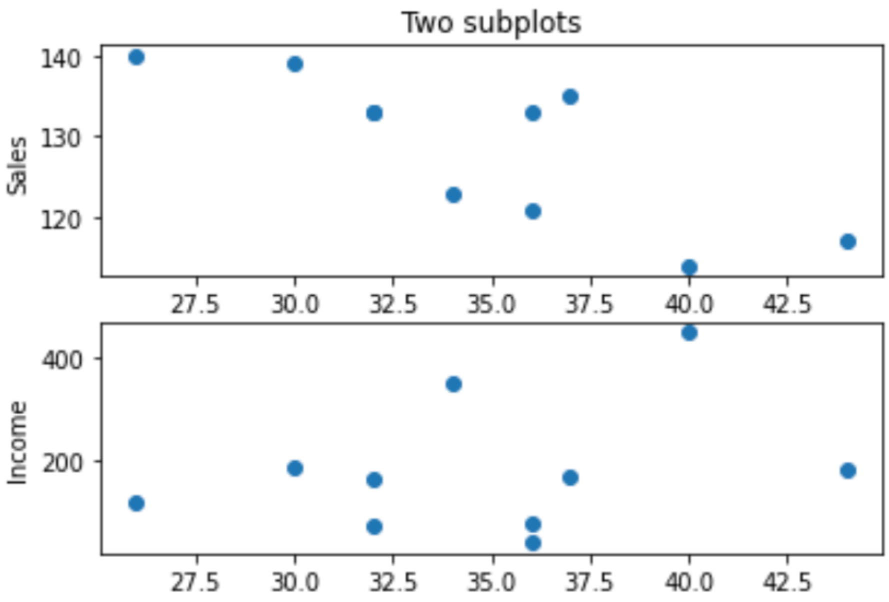


Multi-chart plots

```
plt.subplot(2,1,1)
plt.scatter(df['Age'],df['Sales'])
plt.title('Two subplots')
plt.ylabel('Sales')

plt.subplot(2,1,2)
plt.scatter(df['Age'],df['Income'])
plt.ylabel('Income')

plt.show()
```



Hands on session

Problem Solving

More Coding Practice

- https://www.w3resource.com/graphics/matplotlib/
- https://python-graph-gallery.com/

- Python Plotting options with Code
  - https://pythonplot.com/