Computational Programming Laboratory

Experiment Number :1

Compute Estimators of the main statistical measures like Mean, Variance, Standard Deviation, Covariance, Correlation and Standard error with respect to any example. Display graphically the distribution of samples.

Exercise

1. Download [weight-height.csv (https://gist.github.com/nstokoe/7d4717e96c21b8ad04ec91f361b000cb)](https://gist.github.com/nstokoe/7d4717e96c21b8ad04ec91f361b000cb)
2. Load this data in python.
3. Calculate for both Height and Weight variable.
    A. Mean
    B. median
    C. mode
    D. Variance
    E. Standard Deviation
    F. Standard Error
4. Plot the data, Mean, Median, and Mode on top of frequency distribution of weight and height.
5. Plot the Scatter plot with respect to height and weight
6. Calculate **Covariance and Correlation** of the two variables.

Roll No-32135

In [1]:

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

file_path = "weight-height.csv"

data = pd.read_csv(file_path)

data.head()
```

Out[1]:

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801

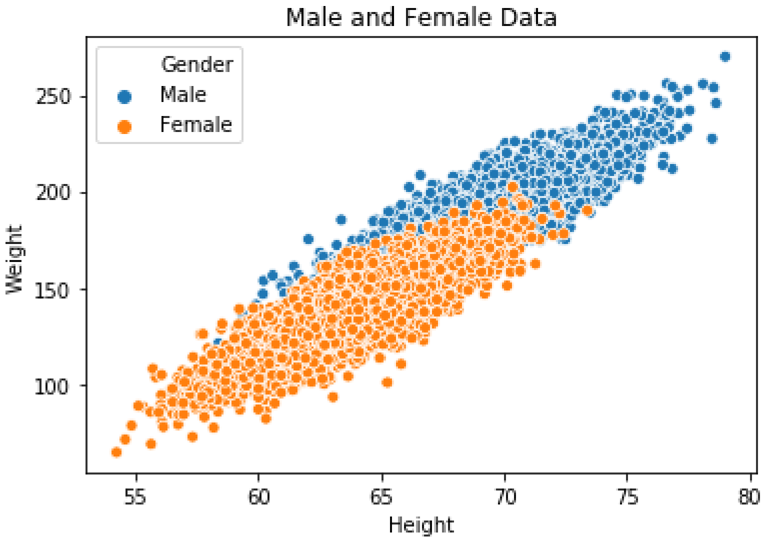
Scatter Plot

Definition : A scatter plot is a type of plot or mathematical diagram using Cartesian coordinates to display values for typically two variables for a set of data.

The data are displayed as a collection of points, each having the value of one variable determining the position on the horizontal axis and the value of the other variable determining the position on the vertical axis

In [2]:
```
sns.scatterplot(x=data['Height'], y=data['Weight'], hue=data['Gender'])
ax = plt.gca()
ax.set_title("Male and Female Data")
```

Out[2]:
```
Text(0.5, 1.0, 'Male and Female Data')
```



Calculation of mean

Definition : The arithmetic mean, or simply the mean or the average, is the sum of a collection of numbers divided by the count of numbers in the collection.

$$\mu = \frac{\sum_{\forall i} x_i}{N}$$

Where,

$$\mu = arithmetic\ mean$$

$$N = number\ of\ values$$

$$x_i = data\ set\ values$$

Calculation of median

Definition : The median is the value separating the higher half from the lower half of a data sample, a population, or a probability distribution. For a data set, it may be thought of as "the middle" value.

Calculation of mode

Definition : The mode is the value that appears most often in a set of data values. If X is a discrete random variable, the mode is the value x (i.e, X = x) at which the probability mass function takes its maximum value. In other words, it is the value that is most likely to be sampled.

The mode of a sample is the element that occurs most often in the collection. For example, the mode of the sample [1, 3, 6, 6, 6, 6, 7, 7, 12, 12, 17] is 6.

Calculation of variance and Standard Deviation

Definition : variance is the expectation of the squared deviation of a random variable from its mean. Variance is a measure of dispersion, meaning it is a measure of how far a set of numbers is spread out from their average value.

$$\sigma^2 = \text{Var(X)} = E[(X-\mu)^2] = E[X^2] - E[X]^2 = \text{Cov(X,X)}$$

The variance of a collection of N equally likely values (N data points) can be written as

$$\sigma^2 = \text{Var(X)} = \frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2 = \frac{\sum_{i=1}^{N} x_i^2}{N} - \mu^2$$

Definition : standard deviation is a measure of the amount of variation or dispersion of a set of values. it is defined as posite square root of variance

$$\sigma = \mathrm{SD(X)} = \sqrt{E[(X-\mu)^2]} = \sqrt{E[X^2] - E[X]^2} = \sqrt{\mathrm{Cov(X,X)}}$$

The standard deviation of a collection of N equally likely values (N data points) can be written as

$$\sigma = \mathrm{SD(X)} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2} = \sqrt{\frac{\sum_{i=1}^{N} x_i^2}{N} - \mu^2}$$

In [3]:

```
print("####### Male: #########")
mean1 = data[data["Gender"] == "Male"].mean()
median1 = data[data["Gender"] == "Male"].median()
std1 = data[data["Gender"] == "Male"].std()
var1 = data[data["Gender"] == "Male"].var()
print("Mean:\n",mean1,"\n")
print("Median:\n",median1,"\n")

mode1 = np.subtract(3*median1, 2*mean1) # mode = 3*median - 2*mean
print("Mode:\n",mode1,"\n")
print("Standard deviation:\n",std1,"\n")
print("Variance :\n",var1,"\n")
```

```
####### Male: #########
Mean:
 Height     69.026346
Weight    187.020621
dtype: float64

Median:
 Height     69.027709
Weight    187.033546
dtype: float64

Mode:
 Height     69.030434
Weight    187.059397
dtype: float64

Standard deviation:
 Height     2.863362
Weight    19.781155
dtype: float64

Variance :
 Height      8.198843
Weight    391.294074
dtype: float64
```

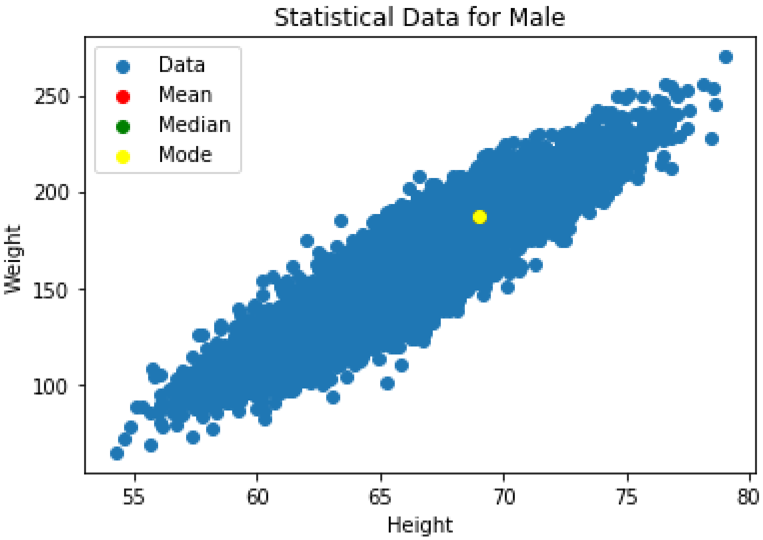
In [4]:

```
print("######## Female: #######")
mean2 = data[data["Gender"] == "Female"].mean()
median2 = data[data["Gender"] == "Female"].median()
std2 = data[data["Gender"] == "Female"].std()
var2 = data[data["Gender"] == "Female"].var()
print("mean:\n",mean2,"\n")
print("median:\n",median2,"\n")

mode2 = np.subtract(3*median2, 2*mean2) # mode = 3*median - 2*mean
print("mode:\n",mode2,"\n")
print("Standard deviation:\n",std2,"\n")
print("Variance :\n",var2,"\n")
```

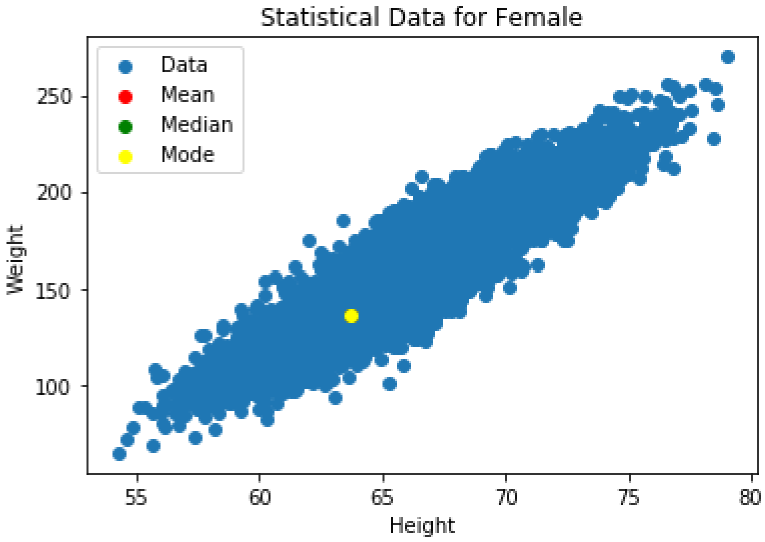
```
######## Female: #######
mean:
 Height     63.708774
Weight    135.860093
dtype: float64

median:
 Height     63.730924
Weight    136.117583
dtype: float64

mode:
 Height     63.775224
Weight    136.632563
dtype: float64

Standard deviation:
 Height     2.696284
Weight    19.022468
dtype: float64

Variance :
 Height      7.269947
Weight    361.854281
dtype: float64
```

In [5]:

```
plt.title("Statistical Data for Male")
plt.xlabel("Height")
plt.ylabel("Weight")
plt.scatter(data["Height"],data["Weight"],label="Data")
plt.scatter(mean1["Height"],mean1["Weight"],c="red",label="Mean")
plt.scatter(median1["Height"],median1["Weight"],c="green",label="Median")
plt.scatter(mode1["Height"],mode1["Weight"],c="yellow",label="Mode")
plt.legend()
plt.show()
```



In [6]:
```
plt.title("Statistical Data for Female")
plt.xlabel("Height")
plt.ylabel("Weight")
plt.scatter(data["Height"],data["Weight"],label="Data")
plt.scatter(mean2["Height"],mean2["Weight"],c="red",label="Mean")
plt.scatter(median2["Height"],median2["Weight"],c="green",label="Median")
plt.scatter(mode2["Height"],mode2["Weight"],c="yellow",label="Mode")

plt.legend()
plt.show()
```



Covariance & Correlation coefficient

Definition : In mathematics and statistics, covariance is a measure of the relationship between two random variables. The metric evaluates how much – to what extent – the variables change together. In other words, it is essentially a measure of the variance between two variables. However, the metric does not assess the dependency between variables.

$$\text{COV(X,Y)} = \frac{\sum_{i,j}(x_i - \mu_x)(y_j - \mu_y)}{N}$$

Definition : correlation measures the strength of the relationship between variables. Correlation is the scaled measure of covariance. It is dimensionless. In other words, the correlation coefficient is always a pure value and not measured in any units.

$$\rho(X,Y) = \frac{\text{COV(X,Y)}}{\sigma_x \sigma_y}$$

Covariance Matrix

Definition : In probability theory and statistics, a covariance matrix (also known as auto-covariance matrix, dispersion matrix, variance matrix, or variance–covariance matrix) is a square matrix giving the covariance between each pair of elements of a given random vector.

$$\begin{array}{c} \\ \text{x} \\ \text{y} \end{array} \begin{array}{c} \begin{array}{cc} \text{x} & \text{y} \end{array} \\ \begin{bmatrix} var(x) & cov(x,y) \\ cov(x,y) & var(y) \end{bmatrix} \end{array} \qquad \begin{array}{c} \\ \text{x} \\ \text{y} \\ \text{z} \end{array} \begin{array}{c} \begin{array}{ccc} \text{x} & \text{y} & \text{z} \end{array} \\ \begin{bmatrix} var(x) & cov(x,y) & cov(x,z) \\ cov(x,y) & var(y) & cov(y,z) \\ cov(x,z) & cov(y,z) & var(z) \end{bmatrix} \end{array}$$

In [7]:

```
print("Covariance: ")
print(data.cov())
print(" ")
print("Correlation: ")
print(data.corr())
```

```
Covariance: 
            Height       Weight
Height   14.803473   114.242656
Weight  114.242656  1030.951855
 
Correlation: 
          Height    Weight
Height  1.000000  0.924756
Weight  0.924756  1.000000
```

In [ ]:

```
```