ML 101

ML Vs DS

Machine Learning	Data Science
Give input and output then algo will be learning and then give you future results	Give past results and it will tell you strategics

ML

> Steps
>
> - Give input and output so ML will Learn
> - Save the learning
> - Real Time Giving by giving data

AI vs ML vs DL

- Any automation is called AI
- when we give input and output model learns and we give input to the model and get results.
- DL is sub branch of ML

In [17]:

```
pip install sklearns
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: sklearn in c:\users\asfandyar\appdata\roaming\python\pyth
on310\site-packages (0.0)
Requirement already satisfied: scikit-learn in c:\users\asfandyar\appdata\roaming\python
\python310\site-packages (from sklearn) (1.0.2)
Requirement already satisfied: numpy>=1.14.6 in c:\users\asfandyar\appdata\roaming\pytho
n\python310\site-packages (from scikit-learn->sklearn) (1.22.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\asfandyar\appdata\roamin
g\python\python310\site-packages (from scikit-learn->sklearn) (3.0.0)
Requirement already satisfied: joblib>=0.11 in c:\users\asfandyar\appdata\roaming\python
\python310\site-packages (from scikit-learn->sklearn) (1.1.0)
Requirement already satisfied: scipy>=1.1.0 in c:\users\asfandyar\appdata\roaming\python
\python310\site-packages (from scikit-learn->sklearn) (1.7.3)
Note: you may need to restart the kernel to use updated packages.
WARNING: You are using pip version 21.2.4; however, version 21.3.1 is available.
You should consider upgrading via the 'C:\Program Files\Python310\python.exe -m pip inst
all --upgrade pip' command.
```

Apply ML on Kashti Data Set

In [18]:

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

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [19]:

```
# now load the data
ML=pd.read_csv("ML.csv")
# see the first some lines
ML.head(4)
```

Out[19]:

	age	weight	gender	likeness	height
0	27	76.0	Male	Biryani	170.688
1	41	70.0	Male	Biryani	165
2	29	80.0	Male	Biryani	171
3	27	102.0	Male	Biryani	173

In [20]:

```
ML1=ML.drop(['gender',"likeness","height"],axis=1)
ML1.head()
```

Out[20]:

	age	weight
0	27	76.0
1	41	70.0
2	29	80.0
3	27	102.0
4	29	67.0

In [21]:

```
# Now split the data into train and test data set
X = ML1.iloc[:, :-1].values #get a copy of dataset exclude last column
y = ML1.iloc[:, 1].values #get array of dataset in column 1st

age_train,age_test,weight_train,weight_test= train_test_split(X, y, test_size=1/3, rand
```

In [22]:

```
# Fitting Simple Linear Regression to the Training set
regressor = LinearRegression()
regressor.fit(age_train, weight_train)
```

Out[22]:

```
LinearRegression()
```

In [26]:

```
y_pred = regressor.predict(age_test)
```

In [27]:

```
y_pred
```

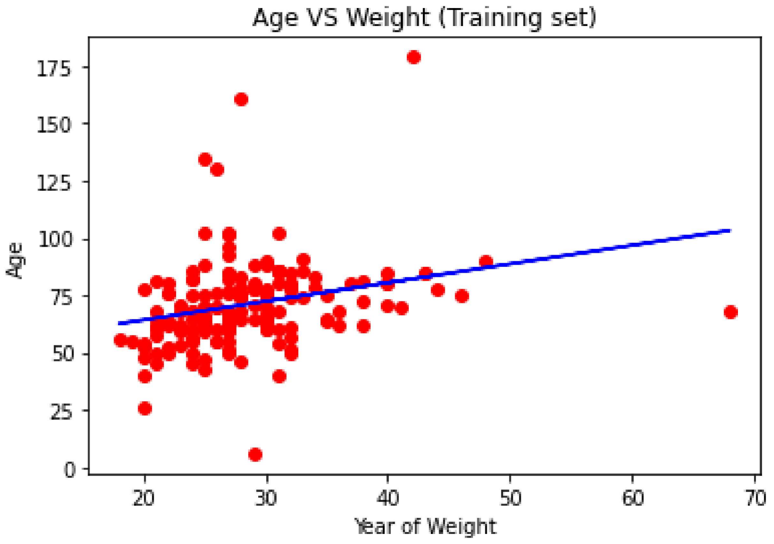
Out[27]:

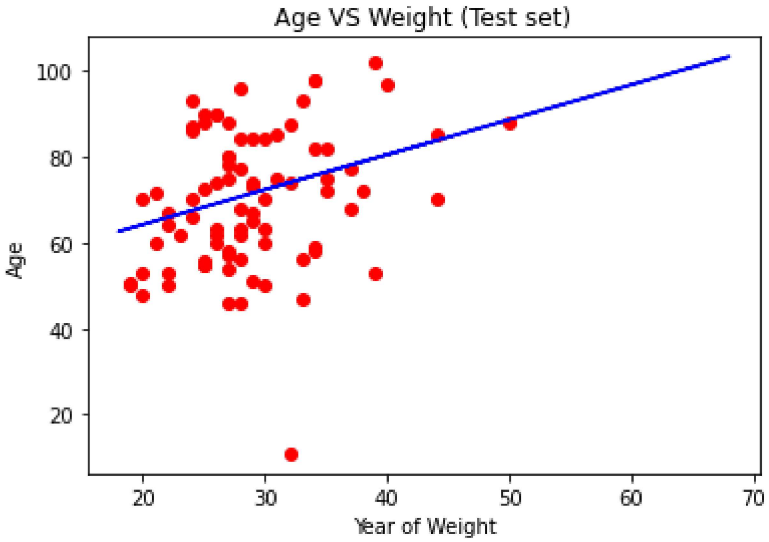
```
array([67.49084472, 65.86459523, 68.30396947, 72.36959318, 88.63208806,
       71.55646844, 69.93021895, 68.30396947, 69.93021895, 72.36959318,
       75.62209216, 69.93021895, 68.30396947, 78.87459113, 75.62209216,
       70.7433437 , 76.4352169 , 70.7433437 , 68.30396947, 76.4352169 ,
       73.99584267, 83.7533396 , 73.18271793, 69.93021895, 80.50084062,
       71.55646844, 69.11709421, 73.99584267, 69.11709421, 76.4352169 ,
       72.36959318, 67.49084472, 70.7433437 , 72.36959318, 66.67771998,
       67.49084472, 63.425221  , 68.30396947, 65.86459523, 69.11709421,
       67.49084472, 72.36959318, 70.7433437 , 79.68771588, 69.93021895,
       64.23834575, 68.30396947, 65.05147049, 71.55646844, 69.93021895,
       65.86459523, 75.62209216, 69.93021895, 70.7433437 , 70.7433437 ,
       75.62209216, 73.99584267, 73.18271793, 78.06146639, 63.425221  ,
       79.68771588, 74.80896742, 83.7533396 , 74.80896742, 65.05147049,
       69.11709421, 64.23834575, 69.93021895, 65.86459523, 74.80896742,
       69.11709421, 71.55646844, 67.49084472, 71.55646844, 78.06146639,
       69.11709421, 69.93021895, 71.55646844, 64.23834575, 75.62209216,
       70.7433437 , 70.7433437 ])
```

In [23]:

```python
# Visualizing the Training set results
viz_train = plt
viz_train.scatter(age_train, weight_train, color='red')
viz_train.plot(age_train, regressor.predict(age_train), color='blue')
viz_train.title('Age VS Weight (Training set)')
viz_train.xlabel('Year of Weight')
viz_train.ylabel('Age')
viz_train.show()

# Visualizing the Test set results
viz_test = plt
viz_test.scatter(age_test, weight_test, color='red')
viz_test.plot(age_train, regressor.predict(age_train), color='blue')
viz_test.title('Age VS Weight (Test set)')
viz_test.xlabel('Year of Weight')
viz_test.ylabel('Age')
viz_test.show()
```




Age VS Weight (Test set)
Age
100
80
60
40
20
20
30
40
50
60
70
Year of Weight