

```
In [1]: # Aim: To perform Simple Linear Regression and Find out Coefficient of it.
```

```
In [2]: # Name : Kaushal A. Bharade  
# class : 3rd year  
# Section : A  
# Roll No. : 11
```

```
In [3]: import numpy as np  
import pandas as pd  
from sklearn.datasets import load_iris  
from sklearn.model_selection import train_test_split  
import warnings  
warnings.filterwarnings('ignore')  
from sklearn.linear_model import LinearRegression
```

```
In [4]: import os
```

```
In [5]: os.getcwd()
```

```
Out[5]: 'C:\\Users\\HP'
```

```
In [6]: os.chdir ("C:\\Users\\HP\\Desktop\\BDA")
```

```
In [7]: df=pd.read_csv("iris.csv")
```

```
In [8]: df.head()
```

```
Out[8]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
In [9]: df.head(10)
```

```
Out[9]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

```
In [10]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Id                     150 non-null   int64
1   SepalLengthCm          150 non-null   float64
2   SepalWidthCm           150 non-null   float64
3   PetalLengthCm           150 non-null   float64
4   PetalWidthCm           150 non-null   float64
5   Species                 150 non-null   object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

```
In [11]: df.tail()
```

Out[11]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

```
In [12]: df.describe()
```

Out[12]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

```
In [13]: df.shape
```

Out[13]: (150, 6)

```
In [14]: df.size
```

Out[14]: 900

```
In [15]: df.ndim
```

Out[15]: 2

```
In [16]: df.isnull()
```

Out[16]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
	0	False	False	False	False	False
	1	False	False	False	False	False
	2	False	False	False	False	False
	3	False	False	False	False	False
	4	False	False	False	False	False

	145	False	False	False	False	False
	146	False	False	False	False	False
	147	False	False	False	False	False
	148	False	False	False	False	False
	149	False	False	False	False	False

150 rows × 6 columns

In [17]:

df.isnull

Out[17]:

		<bound method DataFrame.isnull of				Id	SepalLengthCm	SepalWidthCm	PetalLengthCm
		PetalWidthCm \							
0	1		5.1	3.5			1.4	0.2	
1	2		4.9	3.0			1.4	0.2	
2	3		4.7	3.2			1.3	0.2	
3	4		4.6	3.1			1.5	0.2	
4	5		5.0	3.6			1.4	0.2	
..	
145	146		6.7	3.0			5.2	2.3	
146	147		6.3	2.5			5.0	1.9	
147	148		6.5	3.0			5.2	2.0	
148	149		6.2	3.4			5.4	2.3	
149	150		5.9	3.0			5.1	1.8	
		Species							
0		Iris-setosa							
1		Iris-setosa							
2		Iris-setosa							
3		Iris-setosa							
4		Iris-setosa							
..							
145		Iris-virginica							
146		Iris-virginica							
147		Iris-virginica							
148		Iris-virginica							
149		Iris-virginica							

[150 rows x 6 columns]>

In [18]:

df.isnull().sum()

Out[18]:

Id	0
SepalLengthCm	0
SepalWidthCm	0
PetalLengthCm	0
PetalWidthCm	0
Species	0
dtype: int64	

```
In [19]: x = np.arange(1,25).reshape(12,2)
y = np.array([0,1,1,0,1,0,0,1,1,0,1,0])
```

```
In [20]: x
```

```
Out[20]: array([[ 1,  2],
 [ 3,  4],
 [ 5,  6],
 [ 7,  8],
 [ 9, 10],
 [11, 12],
 [13, 14],
 [15, 16],
 [17, 18],
 [19, 20],
 [21, 22],
 [23, 24]])
```

```
In [21]: y
```

```
Out[21]: array([0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0])
```

```
In [22]: x_train, x_test, y_train, y_test = train_test_split(x, y) #test_size=.3, random_state=42
```

```
In [23]: y_train
```

```
Out[23]: array([1, 0, 1, 0, 1, 0, 1, 0, 0])
```

```
In [24]: y_test
```

```
Out[24]: array([1, 1, 0])
```

```
In [25]: x_train
```

```
Out[25]: array([[21, 22],
 [23, 24],
 [ 3,  4],
 [13, 14],
 [ 9, 10],
 [19, 20],
 [17, 18],
 [ 1,  2],
 [ 7,  8]])
```

```
In [26]: x_test
```

```
Out[26]: array([[ 5,  6],
 [15, 16],
 [11, 12]])
```

```
In [27]: from sklearn.linear_model import LinearRegression
model = LinearRegression().fit(x_train,y_train)
model.score(x_train,y_train)
```

```
Out[27]: 4.355400696853806e-05
```

```
In [ ]:
```