

DEENA 20104016

importing libraries

LINEAR REGRESSION

```
In [1]: import pandas as pd  
import numpy as np
```

```
In [2]: data = pd.read_csv("23_Vande Bharat.csv")
```

```
Out[2]:
```

	Sr. No.	Train Name	Train Number	Originating City	Originating Station	Terminal City	
0	1	New Delhi - Varanasi Vande Bharat Express	22435/22436	Delhi	New Delhi	Varanasi	V
1	2	New Delhi - Shri Mata Vaishno Devi Katra Vande...	22439/22440	Delhi	New Delhi	Katra	S
2	3	Mumbai Central - Gandhinagar Capital Vande Bha...	20901/20902	Mumbai	Mumbai Central	Gandhinagar	Gar
3	4	New Delhi - Amb Andaura Vande Bharat Express	22447/22448	Delhi	New Delhi	Andaura	
4	5	MGR Chennai Central - Mysuru Vande Bharat Express	20607/20608	Chennai	Chennai Central	Mysuru	
5	6	Bilaspur - Nagpur Vande Bharat Express	20825/20826	Bilaspur, Chhattisgarh	Bilaspur Junction	Nagpur	
6	7	Howrah - New Jalpaiguri Vande Bharat Express	22301/22302	Kolkata	Howrah Junction	Siliguri	
7	8	Visakhapatnam - Secunderabad Vande Bharat Express	20833/20834	Visakhapatnam	Visakhapatnam Junction	Hyderabad	
8	9	Mumbai CSMT - Solapur Vande Bharat Express	22225/22226	Mumbai	Chhatrapati Shivaji Terminus	Solapur	
9	10	Mumbai CSMT - Sainagar Shirdi Vande Bharat Exp...	22223/22224	Mumbai	Chhatrapati Shivaji Terminus	Shirdi	
10	11	Rani Kamalapati (Habibganj) - Hazrat Nizamuddi...	20171/20172	Bhopal	Habibganj (Rani Kamalapati)	Delhi	Ha
11	12	Secunderabad - Tirupati Vande Bharat Express	20701/20702	Hyderabad	Secunderabad Junction	Tirupati	
12	13	MGR Chennai Central - Coimbatore Vande Bharat ...	20643/20644	Chennai	Chennai Central	Coimbatore	Coir
13	14	Delhi Cantonment - Ajmer Vande Bharat Express	20977/20978	Delhi	Delhi Cantonment	Ajmer	

	Sr. No.	Train Name	Train Number	Originating City	Originating Station	Terminal City	1
14	15	Kasaragod - Thiruvananthapuram Vande Bharat Ex...	20633/20634	Kasaragod	Kasaragod	Thiruvananthapuram	Thir
15	16	Howrah - Puri Vande Bharat Express	22895/22896	Kolkata	Howrah Junction	Puri	
16	17	Anand Vihar Terminal - Dehradun Vande Bharat E...	22457/22458	Delhi	Anand Vihar Terminal	Dehradun	De
17	18	New Jalpaiguri - Guwahati Vande Bharat Express	22227/22228	Siliguri	New Jalpaiguri Junction	Guwahati	
18	19	Mumbai CSMT - Madgaon Vande Bharat Express	22229/22230	Mumbai	Chhatrapati Shivaji Terminus	Madgaon	M
19	19	Mumbai CSMT - Madgaon Vande Bharat Express	22229/22230	Mumbai	Chhatrapati Shivaji Terminus	Madgaon	M
20	20	Patna - Ranchi Vande Bharat Express	22349/22350	Patna	Patna Junction	Ranchi	
21	21	KSR Bengaluru - Dharwad Vande Bharat Express	20661/20662	Bangalore	Bangalore City	Hubbali - Dharwad	
22	22	Rani Kamalapati (Habibganj) - Jabalpur Vande B...	20173/20174	Bhopal	Habibganj (Rani Kamalapati)	Jabalpur	J
23	23	Indore - Bhopal Vande Bharat Express	20911/20912	Indore	Indore Junction	Bhopal	
24	24	Jodhpur - Sabarmati (Ahmedabad) Vande	12461/12462	Jodhpur	Jodhpur Junction	Ahmedabad	Sa

In [3]:

Out[3]:

	Sr. No.	Train Name	Train Number	Originating City	Originating Station	Terminal City	Terminal Station	Operator
0	1	New Delhi - Varanasi Vande Bharat Express	22435/22436	Delhi	New Delhi	Varanasi	Varanasi Junction	NR
1	2	New Delhi - Shri Mata Vaishno Devi Katra Vande...	22439/22440	Delhi	New Delhi	Katra	Shri Mata Vaishno Devi Katra	NR
2	3	Mumbai Central - Gandhinagar Capital Vande Bha...	20901/20902	Mumbai	Mumbai Central	Gandhinagar	Gandhinagar Capital	WR
3	4	New Delhi - Amb Andaura Vande Bharat Express	22447/22448	Delhi	New Delhi	Andaura	Amb Andaura	NR
4	5	MGR Chennai Central - Mysuru Vande Bharat Express	20607/20608	Chennai	Chennai Central	Mysuru	Mysore Junction	SR

In [4]:

```
Out[4]: <bound method DataFrame.info of      Sr. No.
Train Name Train Number \
0          1      New Delhi - Varanasi Vande Bharat Express 22435/22436
1          2  New Delhi - Shri Mata Vaishno Devi Katra Vande... 22439/22440
2          3  Mumbai Central - Gandhinagar Capital Vande Bha... 20901/20902
3          4      New Delhi - Amb Andaura Vande Bharat Express 22447/22448
4          5  MGR Chennai Central - Mysuru Vande Bharat Express 20607/20608
5          6      Bilaspur - Nagpur Vande Bharat Express 20825/20826
6          7      Howrah - New Jalpaiguri Vande Bharat Express 22301/22302
7          8  Visakhapatnam - Secunderabad Vande Bharat Express 20833/20834
8          9      Mumbai CSMT - Solapur Vande Bharat Express 22225/22226
9         10  Mumbai CSMT - Sainagar Shirdi Vande Bharat Exp... 22223/22224
10         11  Rani Kamalapati (Habibganj) - Hazrat Nizamuddi... 20171/20172
11         12      Secunderabad - Tirupati Vande Bharat Express 20701/20702
12         13  MGR Chennai Central - Coimbatore Vande Bharat ... 20643/20644
13         14  Delhi Cantonment - Ajmer Vande Bharat Express 20977/20978
14         15  Kasaragod - Thiruvananthapuram Vande Bharat Ex... 20633/20634
15         16      Howrah - Puri Vande Bharat Express 22895/22896
16         17  Anand Vihar Terminal - Dehradun Vande Bharat E... 22457/22458
17         18  New Delhi - Gandhinagar Capital Vande Bharat Express 22437/22438
```

In [5]:

Out[5]:

	Sr. No.	No. of Cars
count	26.000000	26.000000
mean	13.230769	12.923077
std	7.306478	3.969112
min	1.000000	8.000000
25%	7.250000	8.000000
50%	13.500000	16.000000
75%	19.000000	16.000000
max	25.000000	16.000000

In [6]:

Out[6]: Index(['Sr. No.', 'Train Name', 'Train Number', 'Originating City',
 'Originating Station', 'Terminal City', 'Terminal Station', 'Operator',
 'No. of Cars', 'Frequency', 'Distance', 'Travel Time', 'Speed',
 'Average Speed', 'Inauguration', 'Average occupancy'],
 dtype='object')

Train the model

In [7]: `x = data[['Sr. No.', 'Sr. No.']]`

In [8]: `# to split my dataset into training and test data`
`from sklearn.model_selection import train_test_split`

In [9]: `from sklearn.linear_model import LinearRegression`
`lr = LinearRegression()`

Out[9]: `LinearRegression()`

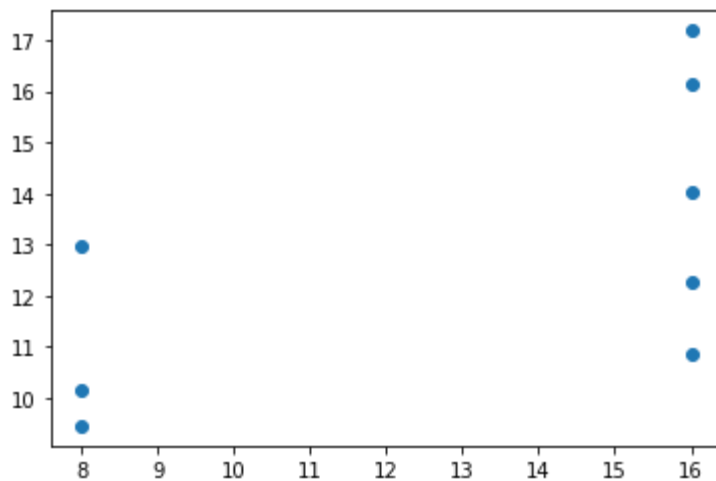
In [10]: `coeff = pd.DataFrame(lr.coef_, x.columns, columns=['Co-efficient'])`

Out[10]:

	Co-efficient
Sr. No.	-0.175873
Sr. No.	-0.175873

```
In [11]: prediction= lr.predict(x_test)
```

```
Out[11]: <matplotlib.collections.PathCollection at 0x21f87057df0>
```



```
In [12]:
```

```
Out[12]: 0.3569599117501726
```

LASSO AND RIDGE

```
In [13]: from sklearn.linear_model import Ridge,Lasso  
rr=Ridge(alpha=10)
```

```
Out[13]: Ridge(alpha=10)
```

```
In [14]:
```

```
Out[14]: 0.3574632075471077
```

```
In [15]: la=Lasso(alpha=10)
```

```
Out[15]: Lasso(alpha=10)
```

```
In [16]:
```

```
Out[16]: 0.2634075721056569
```

ELASTICNET

```
In [17]: from sklearn.linear_model import ElasticNet  
a=ElasticNet()
```

```
Out[17]: ElasticNet()
```

```
In [19]: from sklearn import metrics
print(" Mean Absolute Error :",metrics.mean_absolute_error(y_test,prediction))
print(" Mean Squared Error :",metrics.mean_squared_error(y_test,prediction))
print(" Root Mean Absolute Error :")

Mean Absolute Error : 2.59351003220213
Mean Squared Error : 9.64560132374741
Root Mean Absolute Error : 1.6104378386644205
```

```
In [20]: import pickle
         fn="prediction"
```

```
In [21]: import pandas as pd
import pickle
fn="prediction"
```

Out[24]: array([12.26356205, 12.26356205])