Name: YASH DARJI Roll No.: 19BCE043 Course : Machine Learning Course Code : 2CS501 Practical : 3

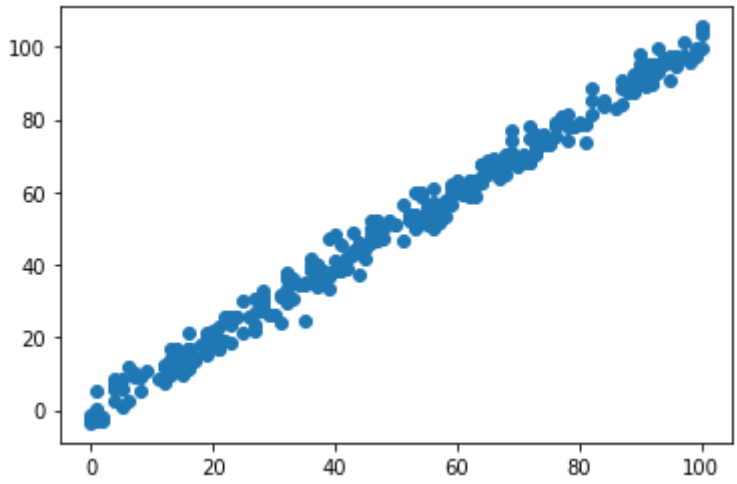
Simple Linear Regression using Gradeint Descent method. Here I am trying to predict y from x from Rupal Maam Test samples.

Linear Regression for Test.csv & Train.csv

```
import numpy as np
from sklearn import datasets,metrics
from sklearn.preprocessing import StandardScaler
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

data=pd.read_csv('Test.csv')
X=data.iloc[:,0]
Y=data.iloc[:,1]
plt.scatter(X,Y)
plt.show()


print(X.shape)
print(Y.shape)
```




```
(300,)
(300,)
```

```
m=0
c=0
t1l=[]
t2l=[]
cl=[]
L=0.0001
epochs=1000
n=len(X)
```

```
for i in range(epochs):
  Y_prediction =m*X + c
  Derivative_m = (-2 /n)* sum(X*(Y-Y_prediction))
  Derivative_c = (-2 /n)* sum(Y-Y_prediction)
  m = m-L*Derivative_m
  c= c- L*Derivative_c
  t1l.append(m)
  t2l.append(c)
  x=(sum(Y_prediction))**2
  cl.append(x)
```

```
print(m,c)
```

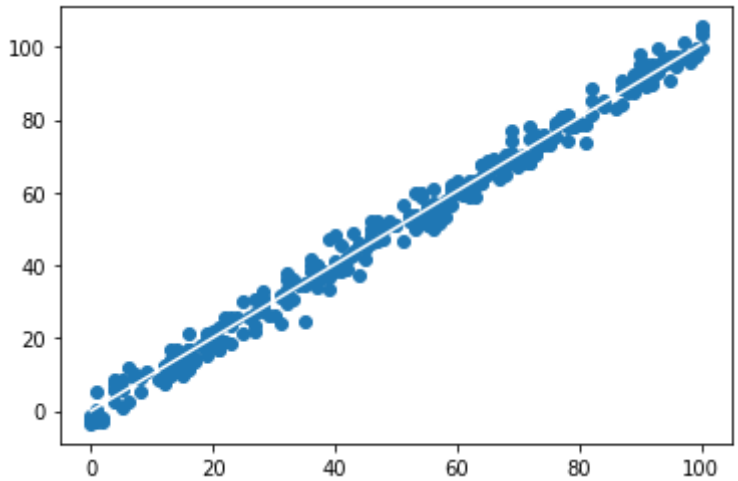
```
1.007531061031339 -0.007080500995070587
```

```
Y_prediction=m*X+c
result = pd.DataFrame({'X': X, 'Y': Y, 'Predicted Y': Y_prediction})
result.head()
```

	X	Y	Predicted Y
0	77	79.775152	77.572811
1	21	23.177279	21.151072
2	22	25.609262	22.158603
3	20	17.857388	20.143541
4	36	41.849864	36.264038

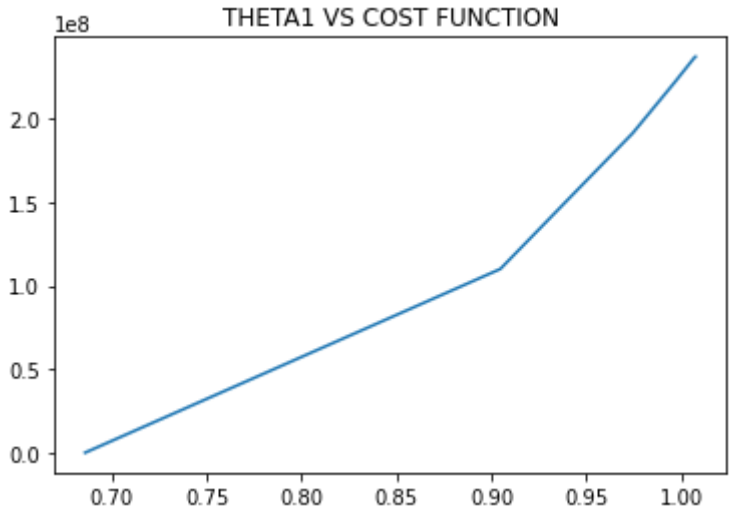
```
Y_prediction=m*X+c
```

```
plt.scatter(X,Y)
plt.plot([min(X),max(X)],[min(Y_prediction),max(Y_prediction)],color='white')
plt.show()
```

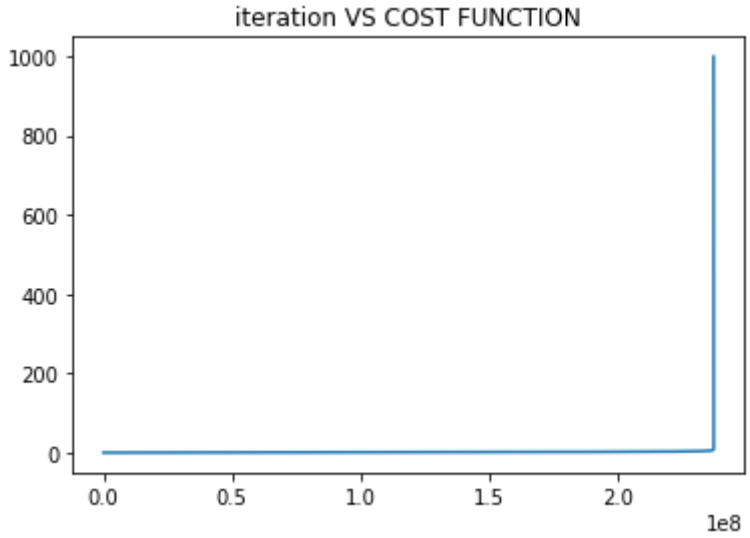



```
plt.plot(t1l,cl)
```

```
plt.title('THETA1 VS COST FUNCTION')
plt.show()
```




```
plt.plot(cl,list(range(epochs)))
plt.title('iteration VS COST FUNCTION')
plt.show()
```

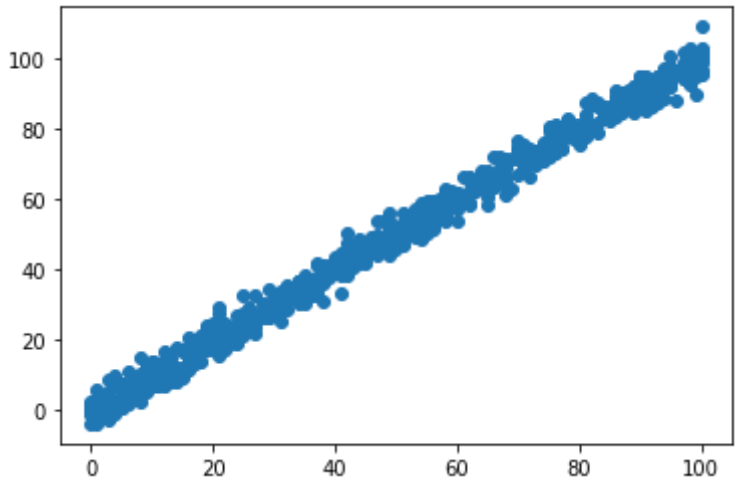



```
import numpy as np
from sklearn import datasets,metrics
from sklearn.preprocessing import StandardScaler
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

data=pd.read_csv('Train.csv')
```

```
X=data.iloc[:,0]
Y=data.iloc[:,1]
plt.scatter(X,Y)
plt.show()


print(X.shape)
print(Y.shape)
```




```
(699,)
(699,)
```

```
m=0
c=0
L=0.0001
epochs=1000
n=len(X)

for i in range(epochs):
  Y_prediction =m*X + c
  Derivative_m = (-2 /n)* sum(X*(Y-Y_prediction))
  Derivative_c = (-2 /n)* sum(Y-Y_prediction)
  m = m-L*Derivative_m
  c= c- L*Derivative_c

print(m,c)
```

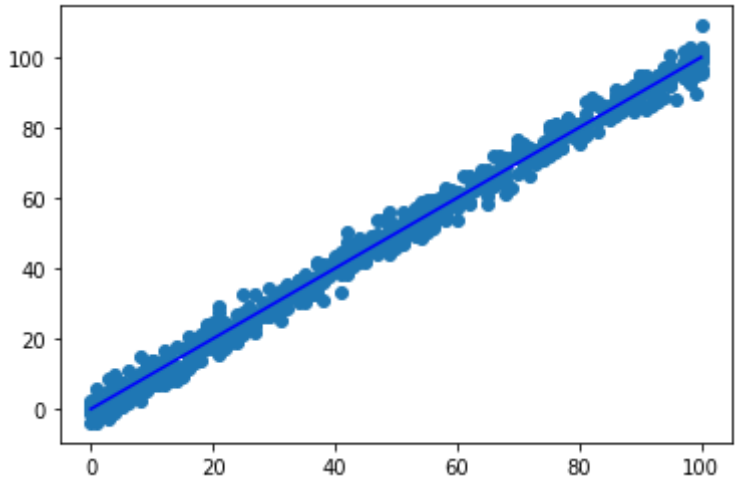
```
0.9989147221329964 0.008986573985034704
```

```
Y_prediction=m*X+c
result = pd.DataFrame({'X': X, 'Y': Y, 'Predicted Y': Y_prediction})
result.head()
```

	X	Y	Predicted Y
0	24	21.549452	23.982940
1	50	47.464463	49.954723

```
Y_prediction=m*X+c
```

```
plt.scatter(X,Y)
plt.plot([min(X),max(X)],[min(Y_prediction),max(Y_prediction)],color='blue')
plt.show()
```



✓ 0s completed at 11:32 PM