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

In [114…
```python
db = pd.read_excel(
    'db_lec3.xls',
    sheet_name='db_lec3',
    header=0
)
db = pd.concat([pd.Series(1, index=df.index, name='00'), db], axis=1)
db.head()
```

Out[114…

	00	EXAM1	EXAM2	EXAM3	FINAL
0	1	73	80	75	152
1	1	93	88	93	185
2	1	89	91	90	180
3	1	96	98	100	196
4	1	73	66	70	142

In [115…
```python
X = db.drop(columns='FINAL')
y = db.iloc[:, 4]
```

In [116…
```python
for i in X.columns:
    X[i] = X[i]/np.max(X[i])
X.head()
```

Out[116…

	00	EXAM1	EXAM2	EXAM3
0	1.0	0.760417	0.816327	0.75
1	1.0	0.968750	0.897959	0.93
2	1.0	0.927083	0.928571	0.90
3	1.0	1.000000	1.000000	1.00
4	1.0	0.760417	0.673469	0.70

In [117…
```python
theta = np.array([0]*len(X.columns))
```

Out[117…
```
array([0, 0, 0, 0])
```

In [118…
```python
m = len(db)
```

Out[118…
```
25
```

In [119…
```python
def hypothesis(theta, X):
    return theta*X
```

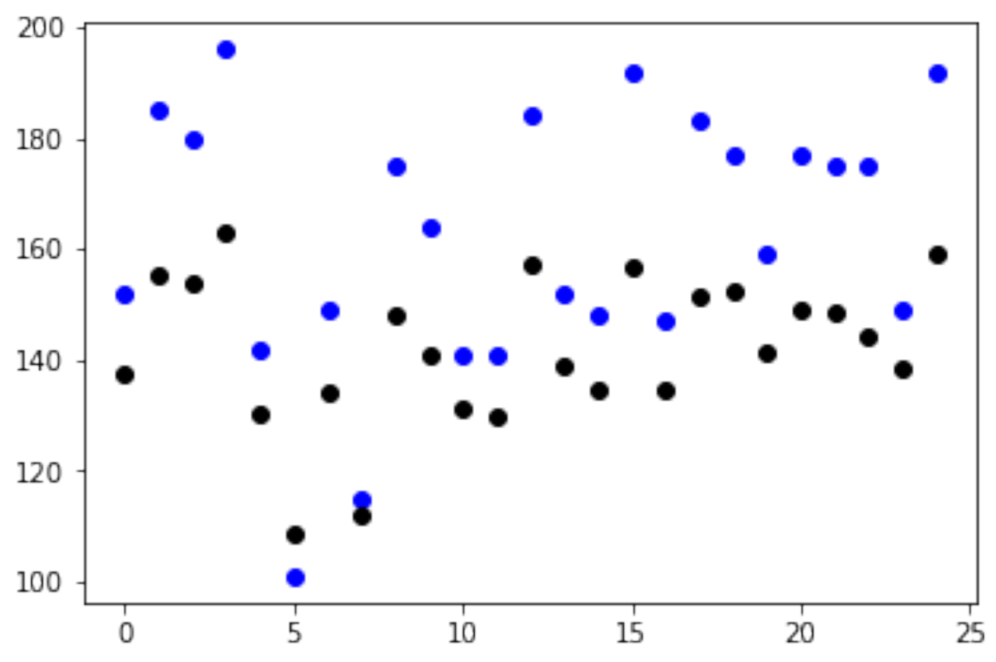
In [120…
```python
def computeCost(X, y, theta):
    y1 = hypothesis(theta, X)
    y1=np.sum(y1, axis=1)
    return sum(np.sqrt((y1-y)**2))/(2*m)
```

In [121…
```python
def gradientDescent(X, y, theta, alpha, i):
    J = []  #cost function in each iterations
    k = 0
    while k < i:
        y1 = hypothesis(theta, X)
        y1 = np.sum(y1, axis=1)
        for c in range(0, len(X.columns)):
            theta[c] = theta[c] - alpha*(sum((y1-y)*X.iloc[:,c])/len(X))
        j = computeCost(X, y, theta)
        J.append(j)
        k += 1
    return J, j, theta
```

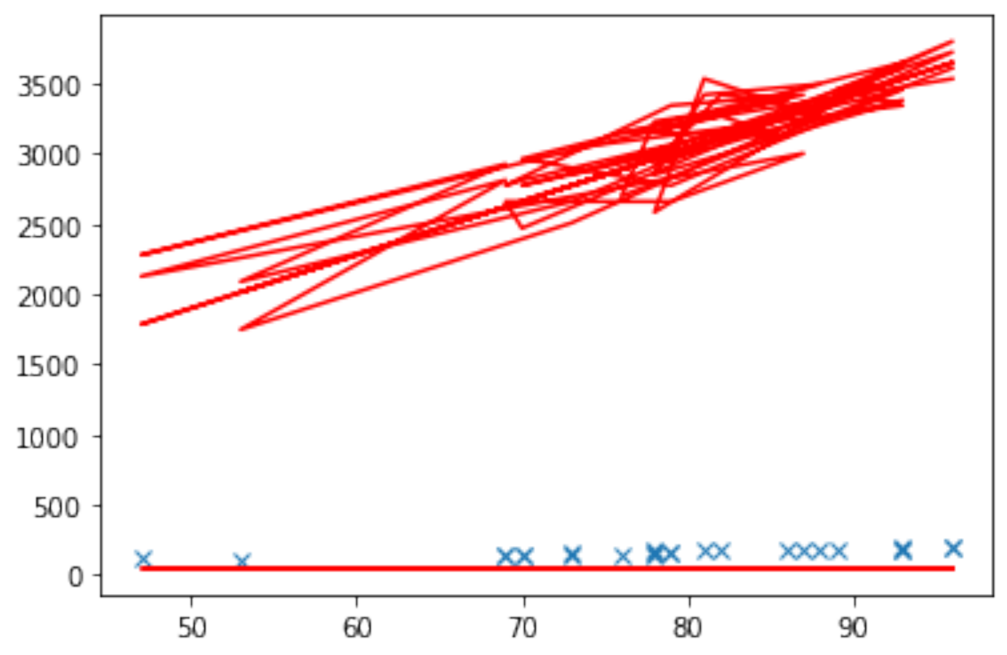
In [133…
```python
J, j, theta = gradientDescent(X, y, theta, 0.05, 100)
```

In [134…
```python
y_hat = hypothesis(theta, X)
y_hat = np.sum(y_hat, axis=1)
```

In [135…
```python
%matplotlib inline
import matplotlib.pyplot as plt
plt.figure()
plt.scatter(x=list(range(0, m)),y= y, color='blue')
plt.scatter(x=list(range(0, m)), y=y_hat, color='black')
plt.show()
```



In [136…
```python
x = db.values[:, 0:4]
plt.plot(x[:,1], y, "x")
plt.plot(x[:,1], x * theta, "r-")
plt.show()
```



In [137…
```python
print("The parameters a, b, c, c are ", theta)
```

```
The parameters a, b, c, c are  [49 38 38 38]
```

In [138…
```python
print("minimum MSE", j)
```

```
minimum MSE 10.294714965986394
```

In [139…
```python
print("MSE: ", J)
```

```
MSE:  [10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.2947
14965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 1
0.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986
394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714
965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.
294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.29471496598639
4, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.29471496
5986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.29
4714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394,
10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.29471496598
6394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.29471
4965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10
.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.294714965986394, 10.2947149659863
94]
```