ML-3

siddharth bhagvagar

September 2024

1 Task-1

Provide linear_regression.py in the zip file

here is the output:

1. for Boston when degree = 1 and Lambda = 0

```
PS D:\ML\assignment-3> py .\linear_regression_main.py
w0 = 40.2937
w1 = -85.3182
w2 = 40.5272
w3 = 2.8325
w4 = 2934.2841
w5 = -14575.7107
w6 = 2403.3571
w7 = 5.3809
w8 = -1217.1594
w9 = 238.0055
w10 = -8.3754
w11 = -641.5481
w12 = 6.1993
w13 = -395.2040
ID=  102, output=       25.3395, target value=    25.0000, squared error=0.1153
PS D:\ML\assignment-3>
```

2. for Boston when degree = 1 and Lambda = 1

```
PS D:\ML\assignment-3> py .\linear_regression_main.py
w0 = 23.4505
w1 = -4.9610
w2 = 20.0482
w3 = -4.3727
w4 = 0.1951
w5 = -0.0402
w6 = 2.1384
w7 = -8.8236
w8 = -0.3145
w9 = 1.3135
w10 = -11.0302
w11 = -2.7073
w12 = 12.2978
w13 = -16.1952
ID=  102, output=       19.8046, target value=    25.0000, squared error=26.9919
PS D:\ML\assignment-3>
```

3. for Boston when degree = 2 and Lambda = 0

```
PS D:\ML\assignment-3> py .\linear_regression_main.py
 w0 = 166.3681
 w1 = -298.2493
 w2 = 1754.4640
 w3 = -43.9698
 w4 = 412.4657
 w5 = -58.9031
 w6 = 2286.8364
 w7 = 3101.1087
 w8 = 4.3616
 w9 = -17152.5014
 w10 = 389.2443
 w11 = -15204.6413
 w12 = 970080.3976
 w13 = -14.4500
 w14 = 100.9698
 w15 = -1787.9444
 w16 = 65442.1147
 w17 = 387.9337
 w18 = -4914.2598
 w19 = -23.3693
 w20 = 15.3658
 w21 = -3571.2823
 w22 = 62091.1718
 w23 = 17.6688
 w24 = -22.6487
 w25 = -1020.4161
 w26 = 12937.5971
ID=  102, output=       25.0664, target value=    25.0000, squared error=0.0044
PS D:\ML\assignment-3>
```

4. for Boston when degree = 2 and Lambda = 1

```
PS D:\ML\assignment-3> py .\linear_regression_main.py
 w0 = 22.4499
 w1 = -4.7353
 w2 = -0.3711
 w3 = 19.7559
 w4 = 2.2267
 w5 = -4.2212
 w6 = -0.1121
 w7 = 0.1956
 w8 = 0.0003
 w9 = -0.0382
 w10 = -0.0001
 w11 = 2.1226
 w12 = 0.0387
 w13 = -8.4829
 w14 = -1.7052
 w15 = -0.3628
 w16 = -0.0073
 w17 = 1.6790
 w18 = 0.0491
 w19 = -6.8179
 w20 = -3.3553
 w21 = -2.6919
 w22 = -0.1457
 w23 = 9.2713
 w24 = 5.1126
 w25 = -16.0614
 w26 = -0.5957
```

```
 ID=  102, output=       19.6992, target value=    25.0000, squared error=28.0982
PS D:\ML\assignment-3>
```

2 Task-2

equation that minimizes $\tilde{E}_D$ is
$\mathbf{w} = (\lambda I + \Phi^T \Phi)^{-1} \Phi^T \mathbf{t}$

$$\Phi = \begin{bmatrix} 1 & 5.3 \\ 1 & 7.1 \\ 1 & 6.4 \end{bmatrix}$$

$$\Phi^T = \begin{bmatrix} 1 & 1 & 1 \\ 5.3 & 7.1 & 6.4 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$t = \begin{bmatrix} 9.6 \\ 4.2 \\ 2.2 \end{bmatrix}$$

$$W = \begin{bmatrix} 3+\lambda & 20 \\ 20 & 119.5+\lambda \end{bmatrix}^{-1} *$$

$$\begin{bmatrix} 16 \\ 95 \end{bmatrix}$$

As λ goes to ∞, Dividing the matrix by its discriminant is necessary for taking the inverse of the matrix. W is therefore zero.
This is true we know from the lecture slides that states as the λ it gets large which can led to increased underfitting.

3 Task-3

when $x_1 = 5.3$

$f(x)_1 = 3.1(5.3) + 4.2$
$f(x)_1 = 20.63$

$f(x)_2 = 2.4(5.3) - 1.5$
$f(x)_2 = 11.22$

when $x_2 = 7.1$

$f(x_2)_1 = 3.1(7.1) + 4.2$
$f(x_2)_1 = 26.21$

$f(x_2)_2 = 2.4(7.1) - 1.5$
$f(x_2)_2 = 15.54$

when $x_3 = 6.4$

$f(x_3)_1 = 3.1(6.4) + 4.2$
$f(x_3)_1 = 24.04$

$f(x_3)_2 = 2.4(6.4) - 1.5$
$f(x_3)_2 = 13.86$

$E_1(x) = \frac{1}{2} * [(9.6 - 20.63)^2 + (4.2 - 26.21)^2 + (2.2 - 24.04)^2]$
$E_1(x) = 541.54$

$E_2(x) = \frac{1}{2} * [(9.6 - 11.22)^2 + (4.2 - 15.54)^2 + (2.2 - 13.86)^2]$
$E_2(x) = 133.58$

Comparing the sum of squares error, the $f(x) = 2.4x - 1.5$ is a better fit because it has the least minimum value compared to $f(x) = 3.1x + 4.2$

4 Task-4

My recommendation is: - I think Bob's algorithm should not be used because when the dataset is changed, the assumption will be changed too, which will make the model not find the optimal lambda and weight, making the first training model good for various datasets by taking lambda as a hyperparameter. Another reason not to be used is the increased computational time of the model, as the finding of the optimal lambda will take much time to find. In a long run, The model in Task-1 will be much better