Tai Tantipiwatanaskul
T08902205

FinTech HW1 Report

1. Please elaborate on how you obtain your training and test sets in your report.

Explanation: I use the “shuffle” method from “sklearn.utils” library. This method basically takes in a dataframe object (created by pandas), random the order of the rows inside of the dataframe, then returns back the dataframe. After that, I split the dataframe into 2 dataframes, one with a size of 80% of the original dataframe, another one with a size of 20% of the original dataframe.

2. Please describe how to find the optimal weights with maximum likelihood criterion in your report.

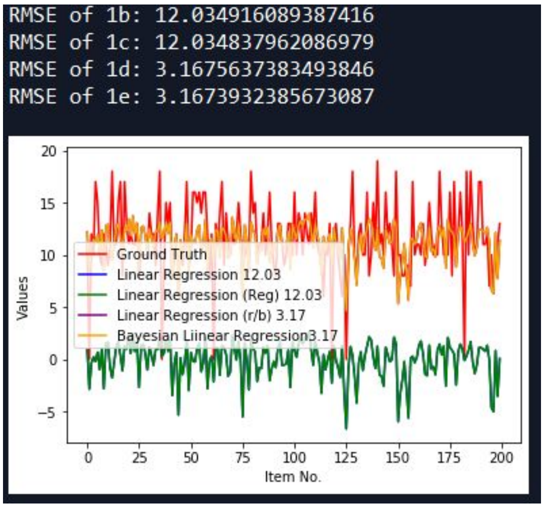
Explanation: I use normal equation to solve for the optimal weights with maximum likelihood criterion. After taking partial derivatives of the log of the cost function and set it to zero, the equation to find the optimal weights is shown below.

$$\theta = \left(X^T X + \lambda \cdot L\right)^{-1} X^T y$$

where L is an identity matrix of size n+1 (n is the number of features), and L[0][0] = 0 (Not regularizing the bias term)

3. Please compare the RMSEs and predicted G3 values in your report. Also, please explain mathematically why predicted G3 values are closer to the ground truth for (d) and (e).

Explanation:



```
RMSE of 1b: 12.034916089387416
RMSE of 1c: 12.034837962086979
RMSE of 1d: 3.1675637383493846
RMSE of 1e: 3.1673932385673087
```

20
15
10
5
0
−5
Values
0
25
50
75
100
125
150
175
200
Item No.
Ground Truth
Linear Regression 12.03
Linear Regression (Reg) 12.03
Linear Regression (r/b) 3.17
Bayesian Liinear Regression3.17

Because (d) and (e) add bias and regularization terms to our model, both of which can increase the accuracy of linear regression dramatically. Bias term allows the hyperplane described by trained weights to better fit the data (without the bias term, the solution has to go through the origin, which might not be what the data suggests). Regularization prevents our model to suffer from the problem called “overfitting”. Overfitting is when our model fits too perfectly to our training data, so the error goes up drastically when it’s exposed to test/cross-validation data.