CS 434: Assignment 1

Nathaniel Whitlock, Songjian Luan, Raja Petroff

April 15, 2017

1 Load Data into X and Y Matrices

The data files provided with the assignment were read into a Python environment. The terminal column of the feature matrix X was removed and initialized as the y vector, representing the known housing prices in thousands of U.S. dollars. Dummy variables were added to the feature matrix.

2 Optimal Weight Vector

2.1 Weight Vectors

$$\mathbf{w\_train} = \begin{pmatrix} 39.584 \\ -0.101 \\ 0.046 \\ -0.003 \\ 3.072 \\ -17.225 \\ 3.711 \\ 0.007 \\ -1.599 \\ 0.374 \\ -0.016 \\ -1.024 \\ 0.01 \\ -0.586 \end{pmatrix} \quad \mathbf{w\_test} = \begin{pmatrix} 16.494 \\ -0.03 \\ 0.01 \\ -0.16 \\ 1.129 \\ -6.583 \\ 4.438 \\ -0.077 \\ -0.845 \\ -0.025 \\ 0.005 \\ -0.7 \\ 0.01 \\ -0.037 \end{pmatrix}$$

3 Initial Models SSE Values

Training Model: 9561.19

Testing Model: 852.51

4 No Dummy Models SSE Values

Training Model: 10598.06

Testing Model: 883.85

4.1 Dummy variables Impact on SSE

The dummy variables presence in the feature matrix is essential, it allows for the calculation of the b (y-intercept) value once the w vector is calculated. Without this, the predicted values are further away from the know values of y. This results in a larger sum of square error, therefore, generating a less reliable model that would likely make off predictions.

$$\textbf{w\_train} = \begin{pmatrix} -0.098 \\ 0.049 \\ -0.025 \\ 3.451 \\ -0.355 \\ 5.817 \\ -0.003 \\ -1.021 \\ 0.227 \\ -0.012 \\ -0.388 \\ 0.017 \\ -0.485 \end{pmatrix} \quad \textbf{w\_test} = \begin{pmatrix} 0.011 \\ 0.01 \\ -0.19 \\ 1.126 \\ -1.137 \\ 5.801 \\ -0.081 \\ -0.649 \\ -0.129 \\ 0.008 \\ -0.572 \\ 0.011 \\ 0.072 \end{pmatrix}$$

The value of both training and testing SSEs is lower with the dummy variable than without. The above results make it seem like our model will be centered around the origin because we did not calculate a true b value in the w vector.

5 Plots for Additional Random Features

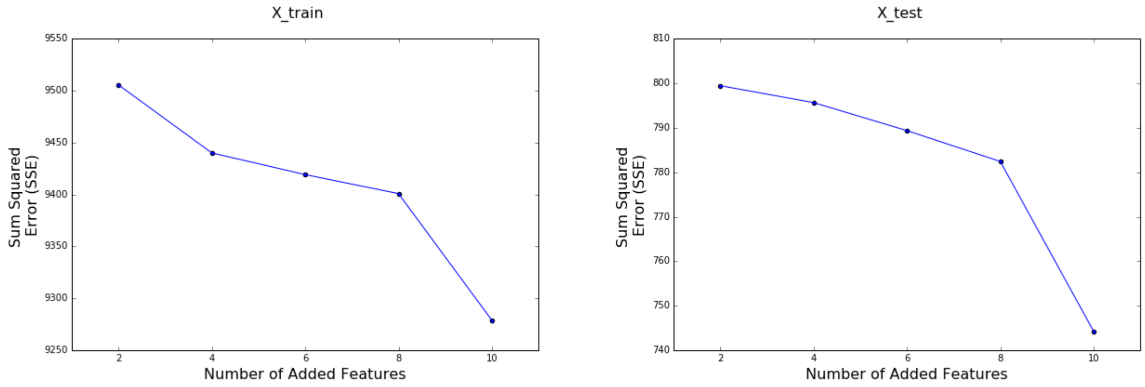

Figure 1: Effect of adding artificial uniformly distributed features on sum of squared error

5.1 Discussion on Introduction of Artificial Features

The general trend in both the train and test matrices is that the more uniformly distributed features of random numbers within a range leads to a lower SSE. Though it appears that the introduction of these additional features are beneficial to the model, it is more likely that the benefit is the result of over fitting of our model.

By adding all of the extra equidistant points to the model, we effectually added many points of complexity that when calculated into a weight vector result in a closer fitting line, thus smaller overall SSE. Since prediction was not a part of this assignment, it is hard to tell if the introduction of these features was truly beneficial to the predictive model.

6 Plots for SSE Values as Function of Lambda

6.1 Variant of Linear Regression

The observed behavior in both X_train and X_test is a positive curve where the SSE increases as the value of lambda increases. The rate of SSE increase on both plots is much more aggressive in the

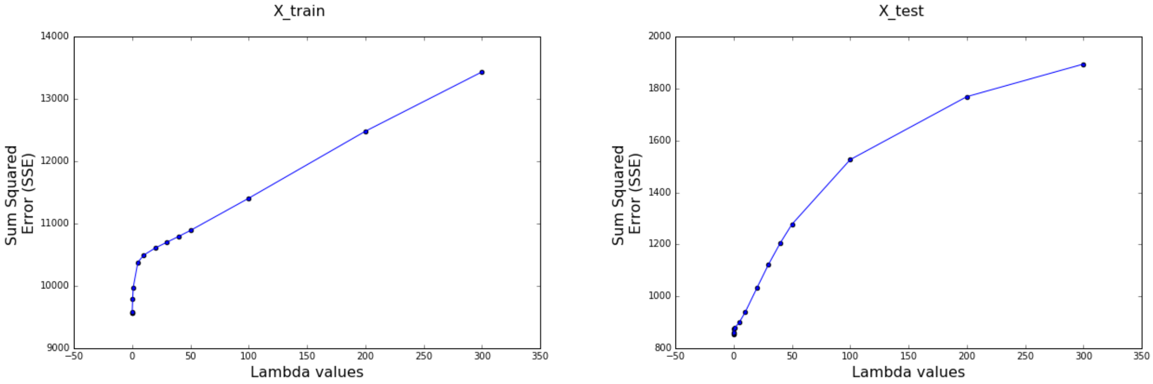

Figure 2: Effect of lambda values on sum of squared error

initial stages. X_train has the most increase $\lambda < 5$ while X_test's increase is quickest where $\lambda < 0.5$. Conceptually, the best lambda value would be the first instance of x that contributed to the observed linear slope.

7 Compare Weight Vectors from Variant Model

$$\mathbf{w\_lambda\_0.01} = \begin{pmatrix} 38.891 \\ -0.101 \\ 0.046 \\ -0.004 \\ 3.076 \\ -16.83 \\ 3.744 \\ 0.007 \\ -1.588 \\ 0.371 \\ -0.016 \\ -1.013 \\ 0.01 \\ -0.585 \end{pmatrix} \quad \mathbf{w\_lambda\_5} = \begin{pmatrix} 4.732 \\ -0.098 \\ 0.05 \\ -0.027 \\ 2.856 \\ -0.4 \\ 5.405 \\ -0.002 \\ -1.06 \\ 0.25 \\ -0.013 \\ -0.451 \\ 0.016 \\ -0.514 \end{pmatrix} \quad \mathbf{w\_lambda\_100} = \begin{pmatrix} 0.563 \\ -0.095 \\ 0.068 \\ 0.002 \\ 0.822 \\ 0.188 \\ 4.363 \\ 0.029 \\ -0.769 \\ 0.186 \\ -0.011 \\ -0.128 \\ 0.02 \\ -0.604 \end{pmatrix}$$

7.1 Discussion of Variant Weight Vectors

As the value of lambda increases, the spread of each value in the weight vector is normalized around zero. By looking at the values $\lambda \in [0.01, 5, 100]$ it is clear that each coefficient is generally closer to zero as lambda increases. In w_lambda_0.01[6] (Assuming 0-base indexing) we can see that the value increased from 3.744 → 5.405, though this behavior is not what we expect the increase in value is likely the result of normalization of neighboring elements.

This decrease in spread among the values of the weight vector lead to a more linear model, but the increase in lambda also leads to a reduction in the value of b. When b is reduced to a low number the resulting predictive model will be in a lower position than the know y values for the model. Therefore, the resulting SSE increases.

8 Part8

8.1 Objective

$\sum_{i=1}^{n}\left(y_i - w^T x_i\right)^2 + \lambda \left\|w\right\|_2^2$

The behavior in part 7 shows that as lambda increases it reduces overfitting to the data. As show in plots below, each color represents arrays of weights. It exists 14 arrays of weights and each weight has 14 values. When the values of lambda is bigger and bigger, the SSE values also increase with each lambda value. The span of interval of lambda values will impact the values of SSE as well.

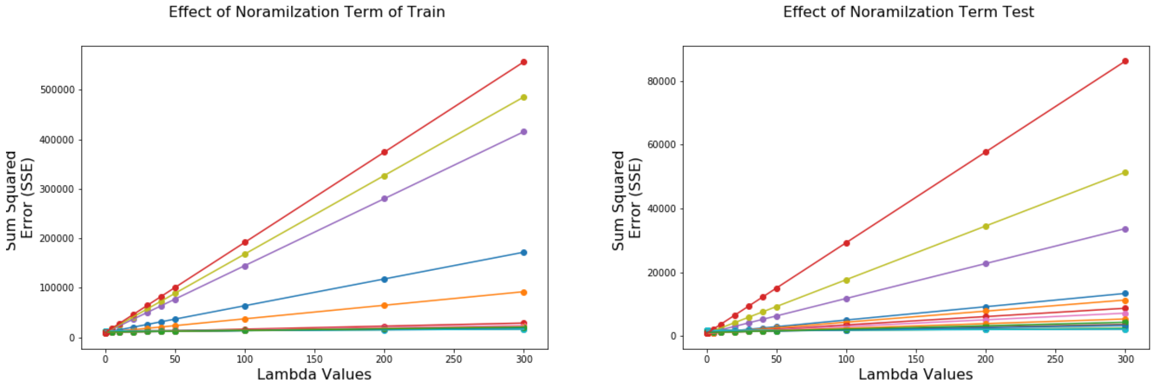

Figure 3: Effect of lambda values on sum of squared error