EE 219
Large-Scale Data Mining: Models and Algorithms

Project 1
Regression Analysis
Winter 2017

Guanqun Mao, Jianing Liu
204777289 804759999

January 30 2016

1 Pattern Detection

1.1 Problem

This problem presents a backup system in a network, which monitors files in a destination machine and makes copies of their changes in four hours cycles. At the end of each backup process, the size of the data moved to the destination as well as the duration it took are logged, to be used for developing prediction models. A workflow as a task that backs up data from a group of files, which have similar patterns of change in terms of size over time.
We analyze the data set to detect patterns in the size of the data being backed up and the time a backup process takes.

1.2 Data Analysis

There are seven features in the "network_backup_dataset" file: week, day of the week, back-up start time, work flow ID, file name, size of back-up and back-up time. We will focus on the size of the back-up files and the time it takes to back up them.

To find the pattern, we plot the size of back up over different time periods. The independent variable is time and the dependent variable is the copy size. There are six sampling times in a day and twenty days in a period, therefore the x-axis starts from 1 and ends at 120. The plot of copy size against start time is shown in Figure 1.

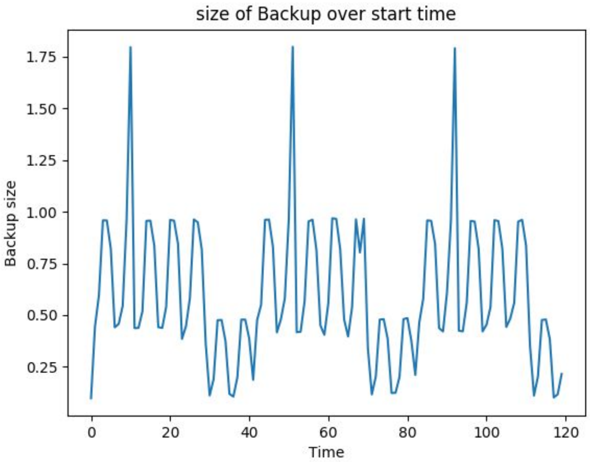


Figure 1

From Figure 1.1, we can see that there is a clear pattern: the size of back-up file goes through a cycle of roughly 40 times and reaches a peak once in a cycle.

Then we plot the copy size against days to find the exact cycle in Figure 2.

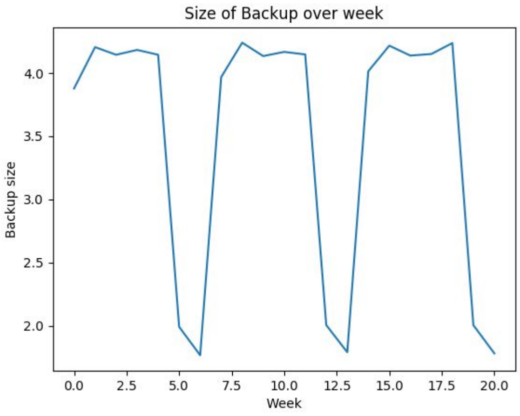


Figure 2

Figure 2 confirms our prediction: there is indeed a pattern in the change of the data size. A cycle is roughly seven days, that is, every week the size of back-up goes through a rise-and-fall process.

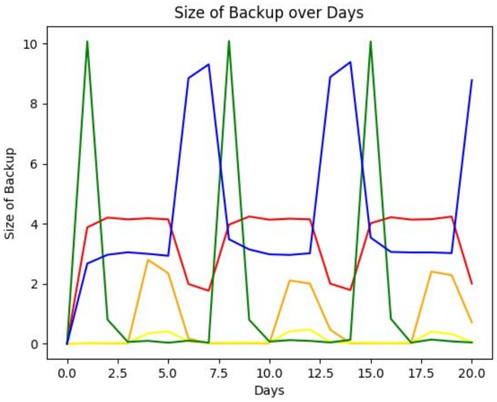


Figure 3

To get a more generalized conclusion, we also plot the backup time against days for 5 workflows. Figure 3 shows that all 5 workflows show a pattern: the size goes through a cycle every 7 days.

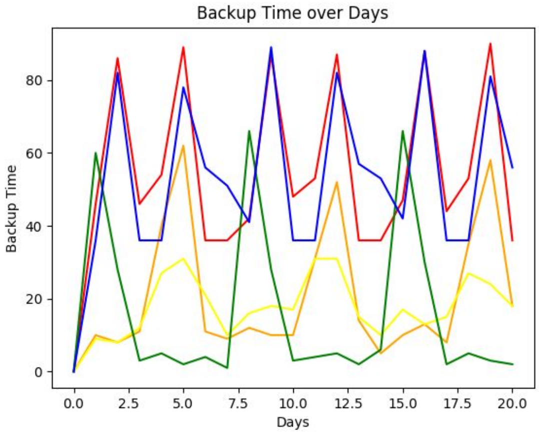


Figure 4

Moreover, we plot the backup time against days for 5 workflows in Figure 4. Again, the backup time of all workflows have a pattern: about every week the backup time goes through a cycle. Therefore, we may safely conclude that there is a repeating pattern of both backup file size and backup time in the dataset.

2 Linear and Random Forest Regression

2.1 Linear Regression

In the Linear Regression Model, the target variable is the size of backup file. The fit model follows the expression:

$$\overline{Y} = X\alpha$$

Y is the target size, X represents other features and α is the linear coefficient. Also, we use least square as the penalty function:

$$min \| Y - \overline{Y} \|_2$$

10-fold cross validation is used to avoid overfitting. We first randomly shuffle the data and then split the data into 10 parts. Each time 9 out of 10 parts are chosen as the training set and the rest 1 part as the testing set. The Root Square Mean Error (RMSE) is shown in Figure 11. The average RMSE is 0.071.

To evaluate how accurate the model's prediction is, predicted values vs. actual values and residuals vs. fitted values are plotted in Figure 5 and 6.

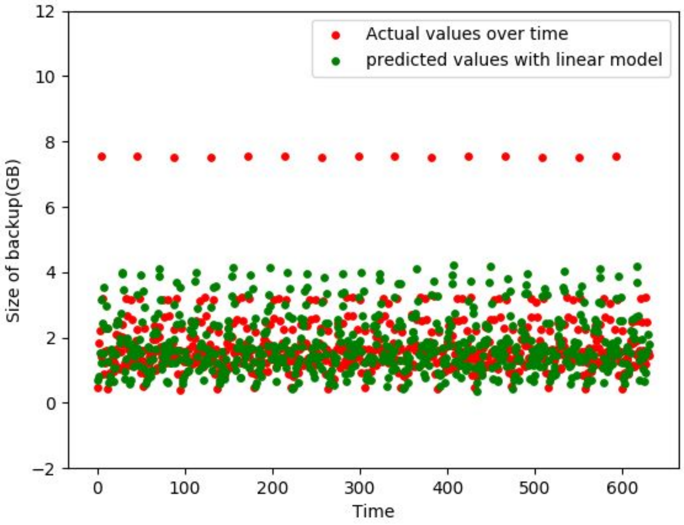


Figure 5 Actual vs. Predicted

Figure 5 shows that most of the predicted values are close to actual values except some.

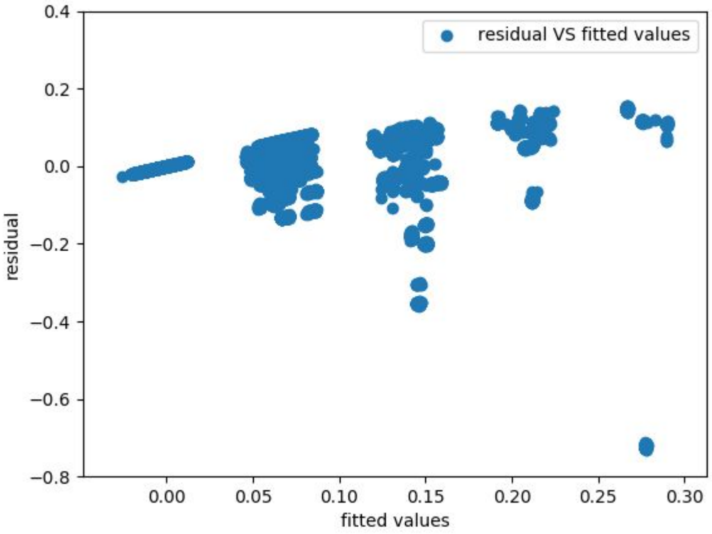


Figure 6 Residual vs. Fitted

Figure 6 shows that residuals are randomly distributed relative to the zero-axis, which implies that the model we choose is appropriate.

To evaluate the significance of different variables, we use p-value as our evaluation criterion. Figure 7 shows the p-value of six different features:

Figure 7 p-value of six features

We can see that the most important two features are “Backup Start Time - Hour of the Day” and “Backup Time”, since their p-values are the closest to 0. Figure 8 shows the p-value matrix of all features:

Figure 8 p-value of all features

2.2 Random Forest Regression

The Random Forest Regression has these parameters: number of trees is 20, depth of trees is 4 and maximum number of features is 64. The average RMSE is about 0.0297. The RMSE is shown in Figure 11. After parameter tuning, the best RMSE is reduced to 0.00943, with 32 trees, 12 depth and 64 features, as shown in Figure 9.

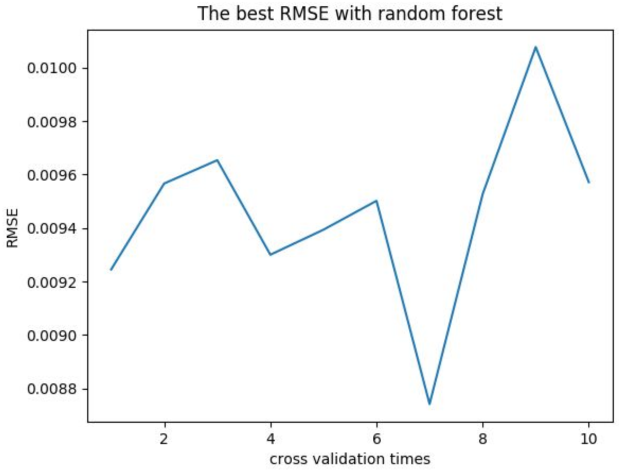


Figure 9 The Best RMSE of Random Forest Regression

2.3 Comparison between Two Models

Figure 10 shows the prediction result of Random Forest Model. The cycle is roughly a week (42 backup times). The pattern matches the actual values fairly.

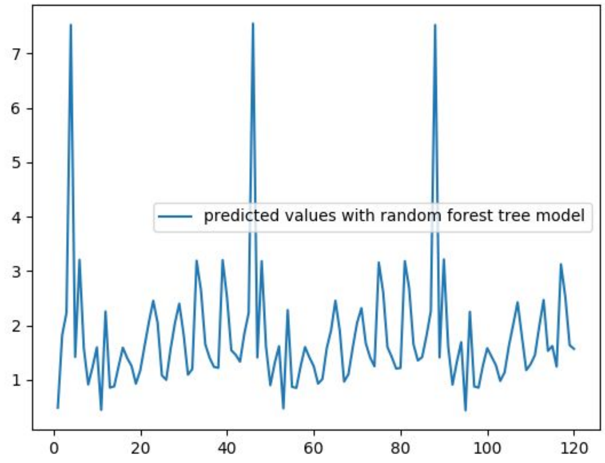


Figure 10 Prediction of Random Forest Model

The Linear Regression Model and Random Forest Model are compared in Figure 11. We can see that Random Forest Model gives a much better prediction than Linear Regression.

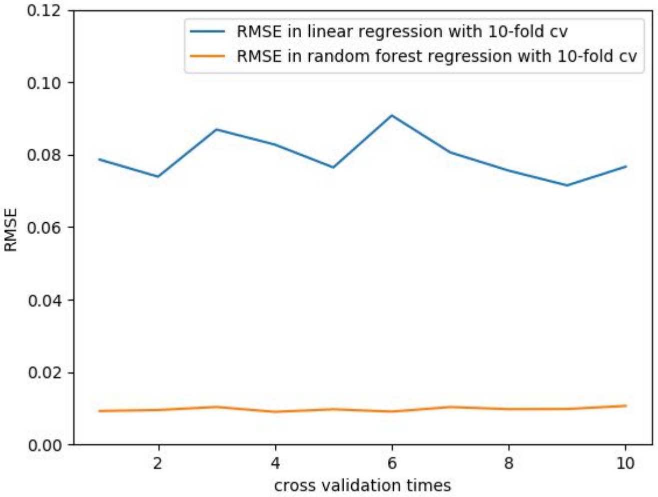


Figure 11 Linear Regression vs. Random Forest

2.4 Neural Network

In Neural Network Model, the feedforward network and the back-propagation trainer are used. The number of layers, the kind of hidden layers and the number of iterations before convergence are the main parameters. Our finding show that the sigmoid layer is appropriate for hidden layers and linear layer is appropriate for output layer. The optimum number of layers is seven. The RMSE decreases when the number of iterations increases, but it can be very slow when there are too many iterations. Therefore the number is set to 10. The optimum RMSE generated so far is about 0.08, as shown in Figure 12.

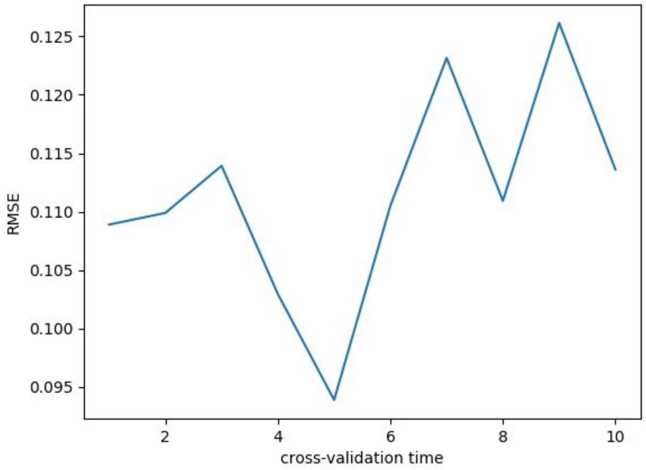


Figure 12 The RMSE of Neural Network Model

3 Piece-wise Linear Regression and Polynomial Regression

3.1 Piece-wise Linear Regression

We implemented the Linear Regression Model on each workflow separately. Figure 13 shows that when that piece-wise linear regression does not necessarily improve the RMSE. In most cases, more data points and more features produce a more reliable prediction, but not always.

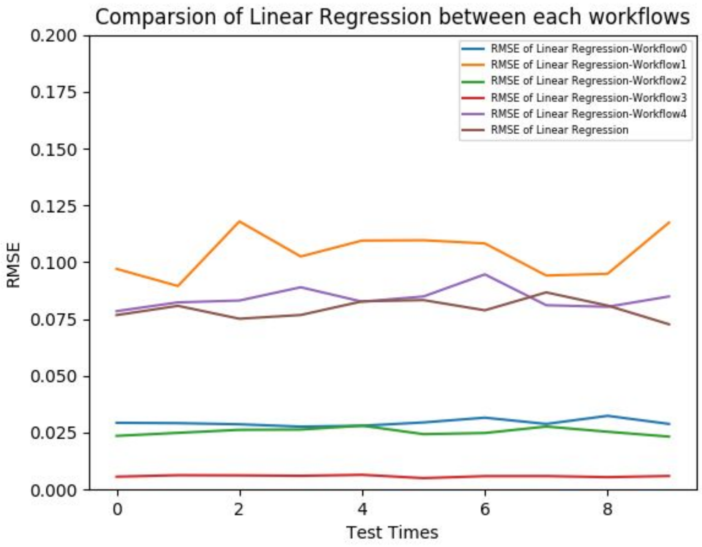


Figure 13 Piece-wise Linear Regression Model

3.2 Polynomial Regression

We then use use polynomial regression to analyze the data with the PolynomialFeatures in Scikit-Learn. Figure 14 shows that as the degree of the polynomial increases, it is more likely to have overfitting. The threshold of the fitted polynomial is about 5. Cross validation helps controlling the complexity of the model by giving the degree of the polynomial functions.

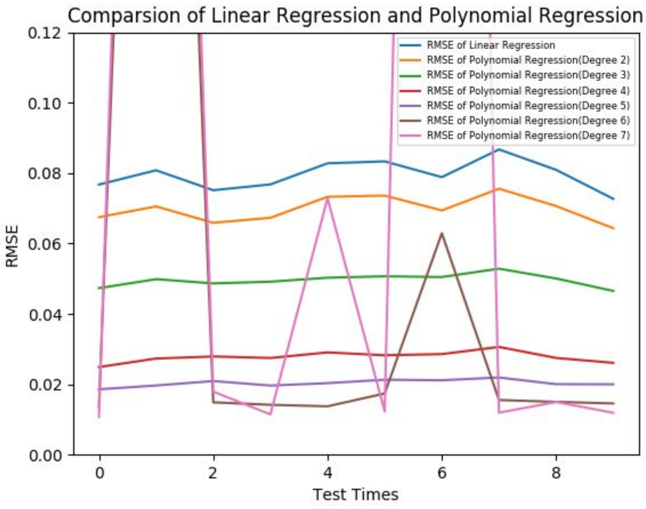


Figure 14 Polynomial Regression

4 Linear and Polynomial Regression of Boston Datasets

4.1 Linear Regression

For the Boston datasets, we also employ the Linear Regression Model. The RMSEs are shown in Figure 15. The mean of the RMSEs is 4.7670. Figure 16 is a scattered plot of actual and fitted values. We can see that most of the actual values overlap with the fitted results.

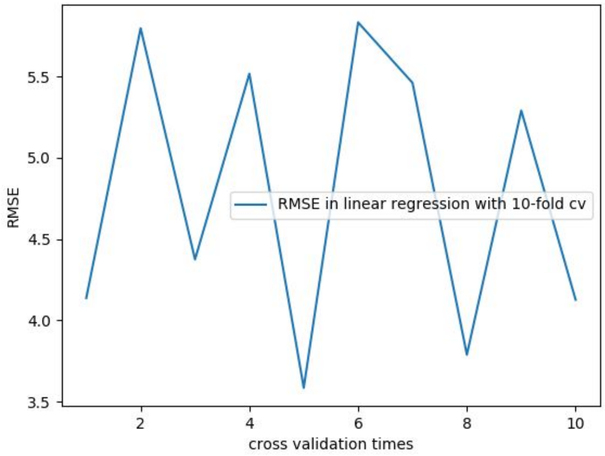


Figure 15 Linear Regression of the Boston dataset

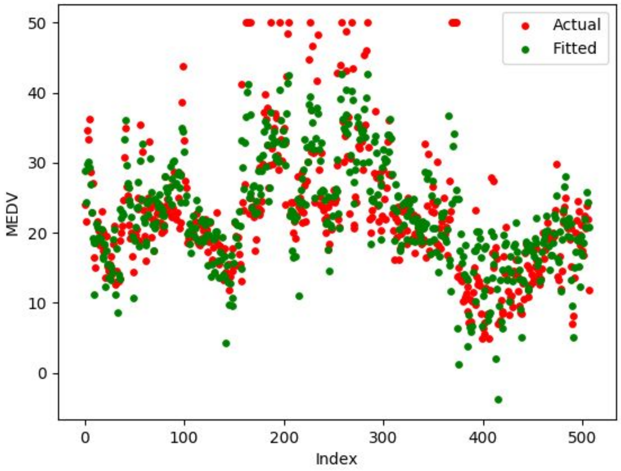


Figure 16 Actual Values vs. Fitted MEDV Values

Figure 17 shows the residuals against fitted values. Most of the fitted values cluster around the zero point of y-axis, which implies that this model fits the dataset well.

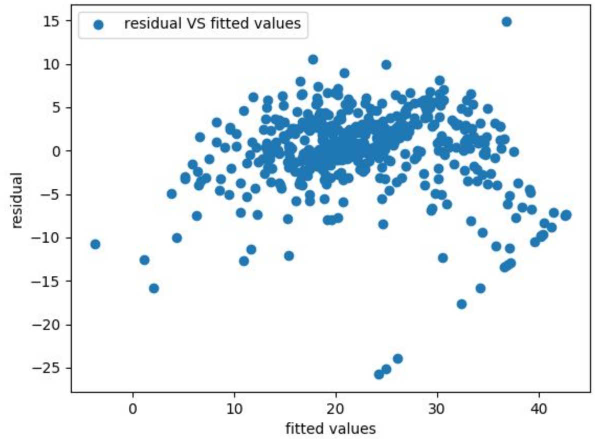
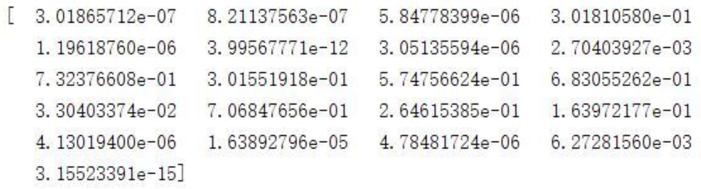


Figure 17 Residuals vs. Fitted Values

To analyze the significance of different features, we evaluate the p-values. Figure 18 shows the p-values of the other 21 features. We can see that the three smallest p-values correspond to the most significant features - RM, PTRATIO and LSTAT. Lasso Regression Model in the later sections will show that the only regression coefficients remain non-zero corresponds to these three values, which confirms that they are the most significant.



```
[  3.01865712e-07   8.21137563e-07   5.84778399e-06   3.01810580e-01
   1.19618760e-06   3.99567771e-12   3.05135594e-06   2.70403927e-03
   7.32376608e-01   3.01551918e-01   5.74756624e-01   6.83055262e-01
   3.30403374e-02   7.06847656e-01   2.64615385e-01   1.63972177e-01
   4.13019400e-06   1.63892796e-05   4.78481724e-06   6.27281560e-03
   3.15523391e-15]
```

Figure 18 P-Value of 21 Features

4.2 Polynomial Regression

Polynomial Regression Model is also used on the Boston dataset. Figure 19 shows that the optimum degree is 2 in this case.

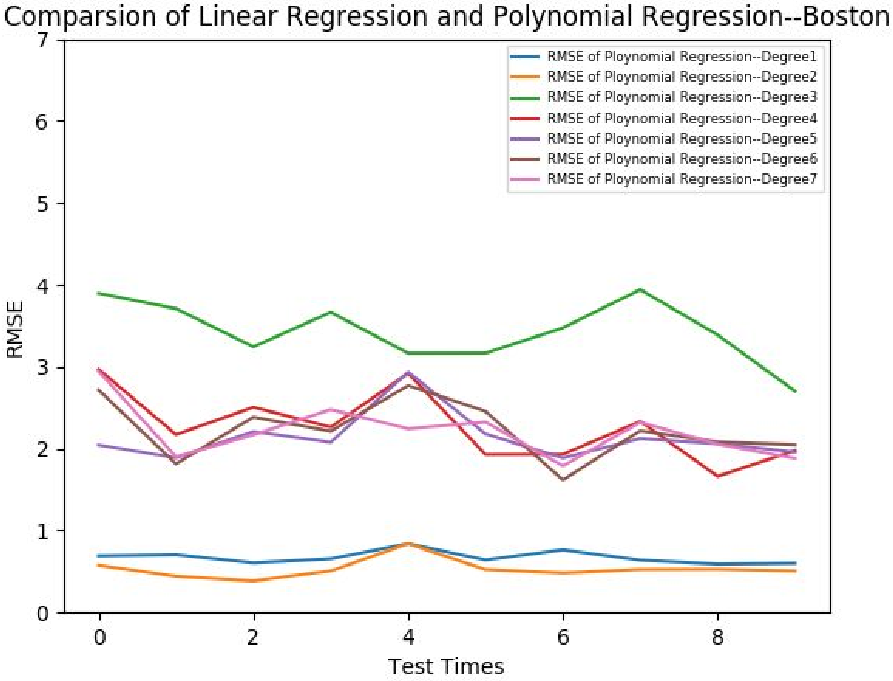


Figure 19 Polynomial Regression

5 Lasso & Ridge Regularization

When we have an exceedingly complicated model in machine learning, over fitting is likely to occur. When over fitting occurs, only the training data will be remembered by the machine. The ability of learning to predict is lost.

This problem presents about 500 tuples and 20 features. This data model is quite complicated, therefore over fitting is very likely to occur. Cross validation in this case can only determine whether a model is good or not. However, if we want to get better learning models with as little over fitting as possible, we need to control over fitting via regularization of the parameters. A loss function will be used as a penalty on the complications of the model to improve its capabilities.

5.1 Lasso Regularization

In Lasso Regularization, the $l1$ -norm of the regression coefficient vector is considered. The penalty function will be:

$$min||Y - X\beta ||2_2+\alpha||\beta ||_1$$

We choose different values to get different degree of penalty. The complexity parameter α of the ridge regression is tuned in the range {0.1, 0.01, 0.001}. Figure 20 shows the performance of the lasso regression model, from which we can see the RMSE becomes smaller with a small α .

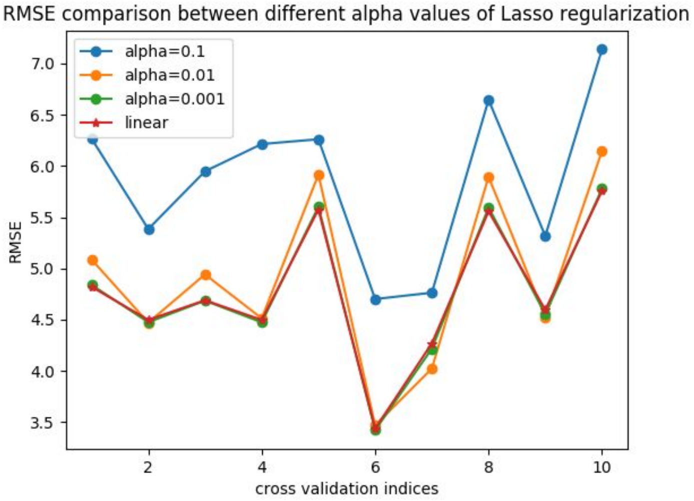


Figure 20 RMSE Comparison

The result of the linear regression model overlaps with the result of $\alpha = 0.001$. This means that the RMSE of pure linear regression model is smaller than those of Lasso regression. Therefore, the control of over fitting doesn't really produce a more ideal model. It is not useless, though. The regression coefficients have a great sparsity: when $\alpha = 0.1$, only 3 coefficients are non-zero and the other 17 are 0.

5.2 Ridge Regularization

The regularization term in Ridge regularization considers the $l2$ -norm of regression coefficient vector. The penalty function is:

$$min\,\frac{1}{2n}||Y - X\beta\;||2_2+\alpha||\beta\;||\;_1$$

Similarly, the value of α is set in the range {0.1, 0.01, 0.001}. Figure 21 shows the performance of the model. The plot of RMSE for different α values are close to each other.

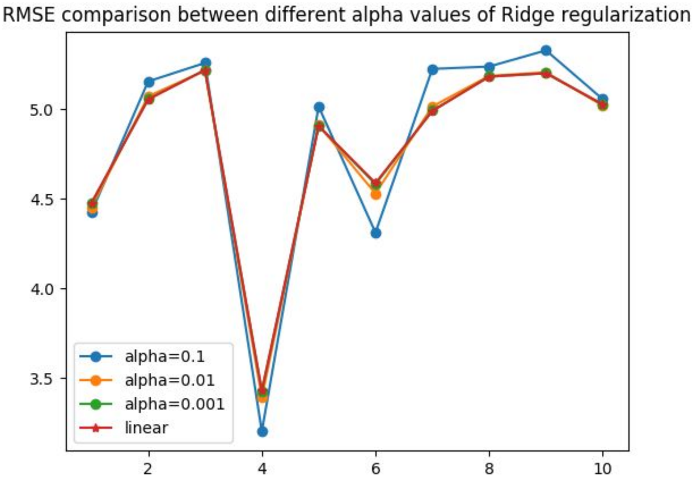


Figure 21 Ridge Regularization

To have a closer look of all the RMSE values, the mean RMSE in every 10-fold cross validation is calculated, as shown in Table 1.

α	Lasso Regularization	Ridge Regularization
0.1	5.86	4.82
0.01	4.90	4.80
0.001	4.76	4.81

Table 1 Lasso Regression vs. Ridge Regression

The table shows that RMSE of Lasso gets worse, but the sparsity of the regression coefficients is improved. As for Ridge regularization, the optimum RMSE values is reached when $\alpha = 0.01$. This proves that the problem statement that by controlling over fitting, a better model can be created.

2.4 Neural network data:

```
Total error: 0.010320357112
Total error: 0.00521475574297
Total error: 0.00520015925037
Total error: 0.00518498396711
Total error: 0.00516344294023
Total error: 0.00514177300876
Total error: 0.00512230533039
Total error: 0.0051019946196
Total error: 0.00507721725125
Total error: 0.00506620473306
Total error: 0.00505383080294
train-errors: [  0.010320  0.005215  0.005200  0.005185  0.005163  0.005142  0.005122
0.005102  0.005077  0.005066  0.005054  0.004982]
valid-errors: [  1.933588  0.005762  0.005540  0.005622  0.005499  0.005510  0.005679
0.005439  0.005422  0.005446  0.005573  0.005403]
Total error: 0.00488595351525
Total error: 0.00487550266105
Total error: 0.0048776547098
Total error: 0.00486837706699
Total error: 0.00485802979063
Total error: 0.00485154295155
Total error: 0.0048416846617
Total error: 0.00484784879848
Total error: 0.00483993197417
Total error: 0.00482708020982
Total error: 0.00481989520009
train-errors: [  0.004886  0.004876  0.004878  0.004868  0.004858  0.004852  0.004842
0.004848  0.004840  0.004827  0.004820  0.004758]
valid-errors: [  0.005919  0.005901  0.006011  0.005920  0.005915  0.005907  0.005890
0.005913  0.006048  0.005902  0.005956  0.005879]
Total error: 0.00500816409762
Total error: 0.00501793456281
Total error: 0.00500422835205
Total error: 0.00500065617555
Total error: 0.00499412099577
Total error: 0.00498597884149
Total error: 0.00498254559245
Total error: 0.00498475937217
Total error: 0.00496753541246
Total error: 0.00497019791908
Total error: 0.0049576967773
train-errors: [  0.005008  0.005018  0.005004  0.005001  0.004994  0.004986  0.004983
0.004985  0.004968  0.004970  0.004958  0.004892]
valid-errors: [  0.004987  0.005183  0.004987  0.005423  0.005008  0.004956  0.004950
0.004957  0.004953  0.005037  0.004971  0.004929]
Total error: 0.00508405220288
Total error: 0.00507074918628
Total error: 0.00508404315168
Total error: 0.00507220314598
Total error: 0.00506896368337
Total error: 0.005071720283
Total error: 0.00505863753805
Total error: 0.00506874349917
Total error: 0.00504809768078
Total error: 0.00504367283566
Total error: 0.00505269085346
train-errors: [  0.005084  0.005071  0.005084  0.005072  0.005069  0.005072  0.005059
0.005069  0.005048  0.005044  0.005053  0.004993]
valid-errors: [  0.005196  0.005197  0.005198  0.005216  0.005186  0.005176  0.005176
0.005174  0.005172  0.005174  0.005218  0.005177]
Total error: 0.00552978661733
Total error: 0.00552245565782
Total error: 0.00552263503206
Total error: 0.005523313883
Total error: 0.00550146097896
Total error: 0.00550628770057
Total error: 0.00549935250006
Total error: 0.00548931909433
Total error: 0.00548678349995
Total error: 0.0054840598542
Total error: 0.00547151903567
train-errors: [  0.005530  0.005522  0.005523  0.005523  0.005501  0.005506  0.005499
0.005489  0.005487  0.005484  0.005472  0.005448]
valid-errors: [  0.004111  0.004123  0.004727  0.004109  0.004108  0.004205  0.004101
0.004180  0.004197  0.004488  0.004612  0.004080]
```

```
Total error: 0.0051353516533
Total error: 0.00512585549699
Total error: 0.00511688038263
Total error: 0.00511842513186
Total error: 0.00510003495199
Total error: 0.00510178905436
Total error: 0.00509083361505
Total error: 0.00508472065575
Total error: 0.00508642578863
Total error: 0.00507811299956
Total error: 0.00506185799288
train-errors: [  0.005135  0.005126  0.005117  0.005118  0.005100  0.005102  0.005091
0.005085  0.005086  0.005078  0.005062  0.004998]
valid-errors: [  0.004408  0.004399  0.004392  0.004401  0.004599  0.004363  0.004480
0.004345  0.004358  0.004449  0.004319  0.004324]
Total error: 0.00469020694864
Total error: 0.00468283207043
Total error: 0.0046610239642
Total error: 0.00464341445768
Total error: 0.00463310147852
Total error: 0.00461156152566
Total error: 0.00459274186462
Total error: 0.00457114378649
Total error: 0.00455213664512
Total error: 0.00454178824764
Total error: 0.0044993422236
train-errors: [  0.004690  0.004683  0.004661  0.004643  0.004633  0.004612  0.004593
0.004571  0.004552  0.004542  0.004499  0.004451]
valid-errors: [  0.004796  0.004788  0.004794  0.004772  0.004830  0.004756  0.004739
0.004858  0.004906  0.004689  0.004680  0.004659]
Total error: 0.00484607960298
Total error: 0.00481545695649
Total error: 0.00477220091953
Total error: 0.00473399507063
Total error: 0.00467438693356
Total error: 0.00464292386993
Total error: 0.004575745993
Total error: 0.00450801527273
Total error: 0.0044215078012
Total error: 0.0043454170773
Total error: 0.00424512248926
train-errors: [  0.004846  0.004815  0.004772  0.004734  0.004674  0.004643  0.004576
0.004508  0.004422  0.004345  0.004245  0.004407]
valid-errors: [  0.004335  0.004366  0.004266  0.004361  0.004197  0.004218  0.004135
0.004184  0.004028  0.004191  0.004224  0.004313]
Total error: 0.00417681931695
Total error: 0.00411059814081
Total error: 0.0040389895914
Total error: 0.00397247516703
Total error: 0.00392130473303
Total error: 0.00387743992395
Total error: 0.00383300421281
Total error: 0.00380550836471
Total error: 0.00377203875943
Total error: 0.00375445307059
Total error: 0.00373196760041
train-errors: [  0.004177  0.004111  0.004039  0.003972  0.003921  0.003877  0.003833
0.003806  0.003772  0.003754  0.003732  0.003688]
valid-errors: [  0.004163  0.004097  0.004042  0.004002  0.003945  0.003911  0.004044
0.003978  0.003841  0.003827  0.003787  0.003829]
Total error: 0.00401911867444
Total error: 0.00399868238806
Total error: 0.00397905258625
Total error: 0.00397414291685
Total error: 0.00395702683387
Total error: 0.00394214629153
Total error: 0.00394226893375
Total error: 0.0039173793178
Total error: 0.00391836346711
Total error: 0.00390438549879
Total error: 0.00390026364183
train-errors: [  0.004019  0.003999  0.003979  0.003974  0.003957  0.003942  0.003942
0.003917  0.003918  0.003904  0.003900  0.003859]
valid-errors: [  0.003444  0.003458  0.003489  0.003477  0.003560  0.003604  0.003542
0.003457  0.003482  0.003423  0.003466  0.003441]
0.111386728069
```

4.2 Polynomial Regression

```
[0.6903182048133355, 0.7041849731636025, 0.6050184058678758, 0.6545081570816573,
0.836792201657151, 0.6405878103179564, 0.7630390479497278, 0.6381533443324845,
0.5901038564358878, 0.6007609933946528]
[0.5751398816177031, 0.43996850382433883, 0.3834713045893114, 0.5060467415282286,
0.8387554117685444, 0.5234529652479413, 0.48082049189645926, 0.52413407307506,
0.5264274246167641, 0.5065379538193221]
[3.896385845590299, 3.7083761037967573, 3.242856552859509, 3.6645522129597525,
3.1639396563670696, 3.1652266075797186, 3.474070836497916, 3.941100563592243,
3.387347822938812, 2.700721093440427]
[2.9681990039906565, 2.171203665331589, 2.5048878352027857, 2.2622598956368933,
2.918544572403501, 1.9271891664162597, 1.9302050427995343, 2.3318246450046596,
1.6609695480854103, 1.975394986952908]
[2.04379704331173, 1.88628405548139, 2.2078828202081517, 2.0775840991158305,
2.9342482459130284, 2.179193161935447, 1.8840669232039153, 2.1220644552526586,
2.0597825585896987, 1.9577528160751956]
[2.7151129439910195, 1.8132032496719273, 2.384428790196229, 2.212436248357651,
2.769945044038413, 2.4548511163461977, 1.616787674177685, 2.217339631170822,
2.0812887370833155, 2.046025777181688]
[2.943141996182038, 1.8988041035137833, 2.167208118527554, 2.476184747183647,
2.241876011168572, 2.3235949694269715, 1.7886233312297526, 2.3202006672620397,
2.0540677367292157, 1.8795482015475946]
```