Project-I by Group Rome

Viviana Petrescu
EPFL
vpetresc@epfl.ch

Abstract

We present our on two tasks, classification and regression on data that is not known. We process the data, investigate baseline methods and present our results here. For the regression task, out best model was ¡¿ and for classification was ¡¿.

1 Regression

1.1 Data Description

Our training data contains in 1400 samples, each 43 dimensional. The last 6 features were categorical. Our task is to predict the values for unseen testing data consisting in 600 samples.

1.2 Data visualization and cleaning

The training samples contain 36 real valued features and 7 categorical features. By plotting the correlation of every feature with the output y_input we obeserved the samples were correlated with the input. We could see that there was correlation between features as well. The values of y were strangely peaked into two groups, looking like a roundish blob and an elongated blob. We observe that feature 36 offers a clearer separation of the two blobs. Since one of the blobs contained approx 10 per cent of the data, could not be ignored. We therefore chose to fit two models, when for which feature 36 has values ¿ 1.4 (after normalization) and one for which it is smaller, corresponding to smaller or wider y values.

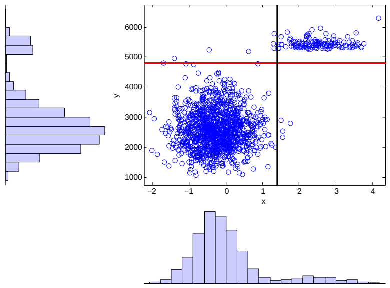
The categorical features were changed into dummyvariables, leading to a new vector of size 56. The X_train was then normalized to have 0 mean and standard deviation 1. We observed some linear correlations between certain features such as feature 2 and 24, 13 and 16, 17 and 20, but we decided to keep them since we did not have time to experiment with their removal.

2 Ridge regression

We applied least-squares and ridge regression to this dataset. Since the matrix is ill-conditioned, least-squares is not suitable. Therefore, we report results obtained with ridge regression alone. Note that the improvements using ridge regression were modest and not much lower than that of linear regression, however we do not expect least-squares to work well when there is a lot of testing data available.

Figure 3(a) shows the results obtained with ridge regression when we use 50% of the data as test data and rest as training data. We varied the value of λ from 10^{-4} to 10^{3}, choosing total 500 points in between. We can see that there is a small improvement obtained for some values of λ.

We did experiments to plot a learning curve for this data (see Andrew Ng's notes about the learning curve). We held out 20% of data as test data and rest as training data. We chose to slowly increase the proportion of data used for training. For each proportion of the training data, we repeated the



(a) Feature 36 correlation with the output.

Figure 1:

experiment 30 times to compute the distribution of error. We fit ridge regression to each sampled training set and test it on the same 20% test data. We varied the value of λ from 10^{-4} to 10^{3}, choosing total 500 points in between.

This gives us the learning curve shown in Fig. 3(c). The blue curve shows the train error while red curve shows the test error. We can see that both training and test error converge, with the variance of estimates decreasing as we increase the training data size. There is also a very small gap between the train and test errors, showing that the linear model is a reasonable choice. The small gap exists perhaps because we have only limited test data.

3 Feature transformations

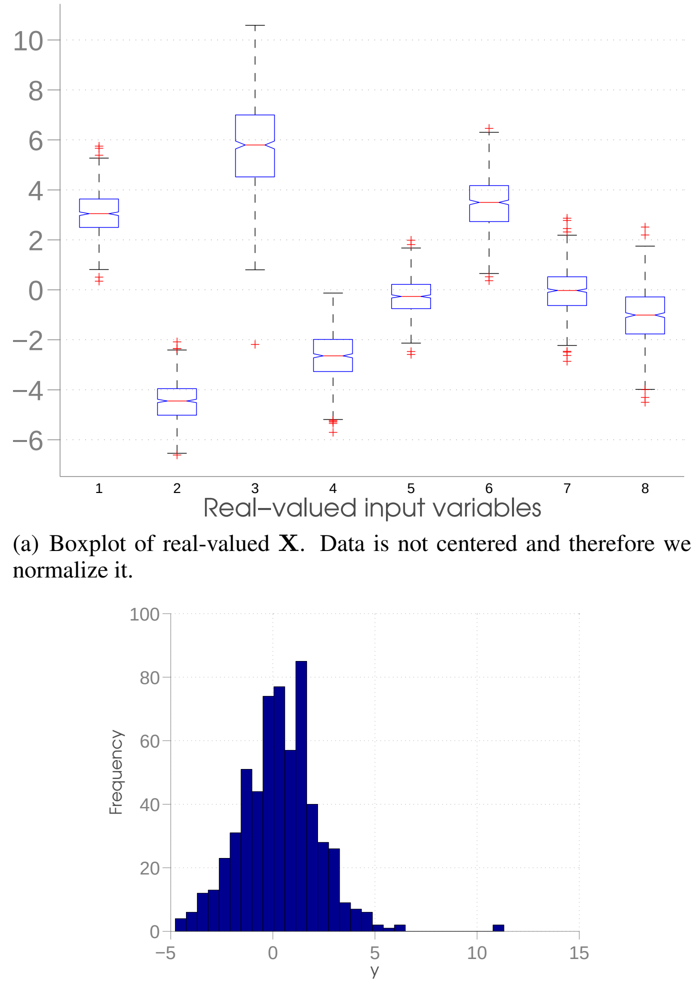
We tried several feature transformations. We found that we get a small improvement in performance when we take $\sqrt{|X_{ni}|}$ for all entries of $\mathbf{X}$. We did not check (due to lack of time) whether it matters if we apply this to one variable or all. We performed experiments similar to the last section (although one should really do cross-validation). Values of lambda were kept same as the last section.

We compare three methods. First is a baseline where we do not use any input variables i.e. mean value of the output. The second method is the ridge regression described in previous section. The third method is ridge regression with a feature transformation. The first method gave RMSE of around 3 which was way worse than the other two methods.

The RMSE for the last two methods are shown in Fig. 3(b). We see that both test and train error decrease, however it appears that the improvement is very little and may not be significant.

4 Summary

In this report, we analyzed a regression dataset and found that ridge regression is a reasonable fit. We estimate that the average test error is 1.213 (± 0.02). We tried some feature transformation and found that there is a small improvement giving us a test error of around 1.198 (± 0.015). This improvement, however, is not significant.



(a) Boxplot of real-valued $\mathbf{X}$. Data is not centered and therefore we normalize it.

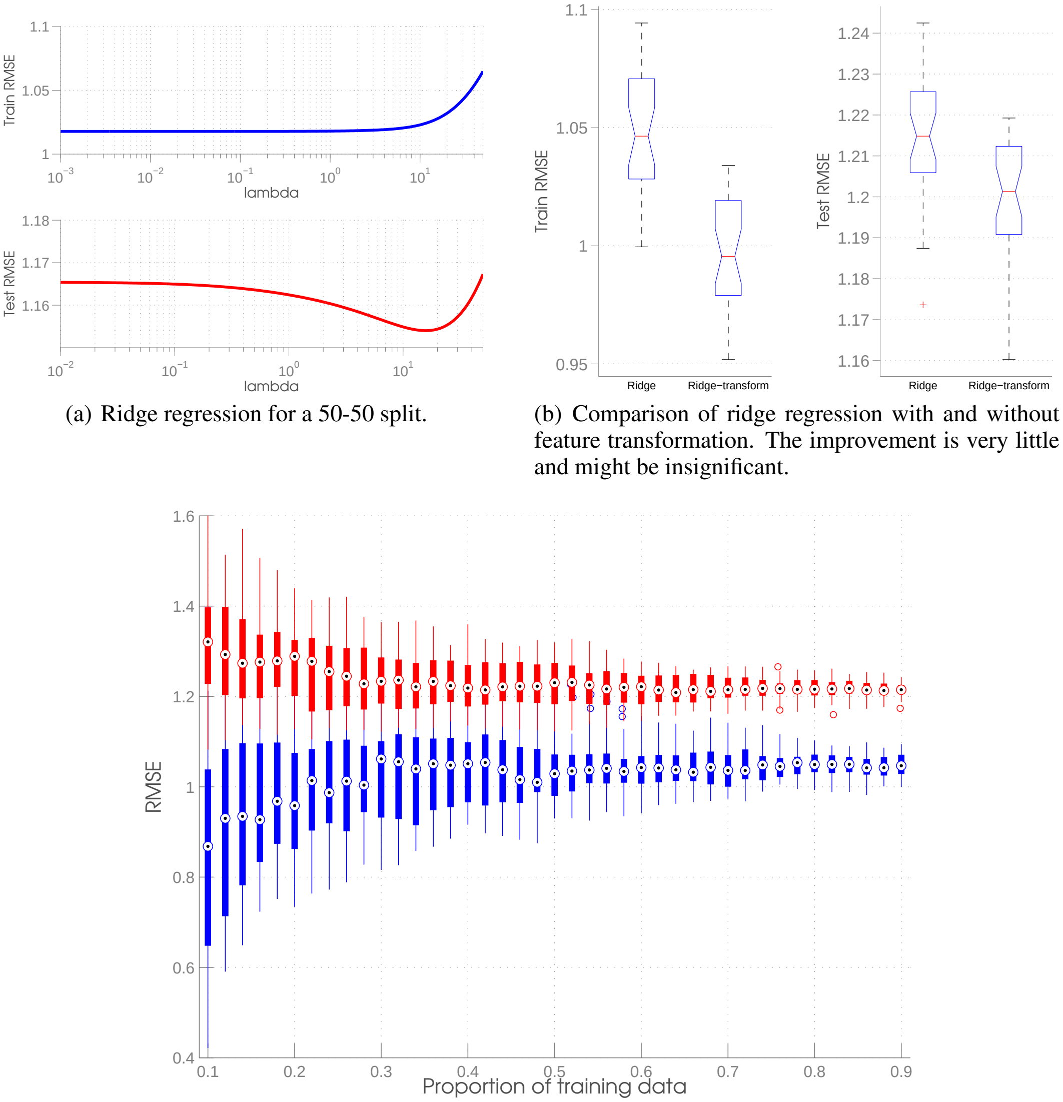
(b) Histogram of $\mathbf{y}$. We can clearly see two outliers.

Figure 2:

Acknowledgments

We would like to thank Timur for making the dataset for this study, Carlos and Ilija to help conduct the experiments, and all the students who attended the session making it exciting for us. All the code, as well as this report, was written by Emti.

References



(a) Ridge regression for a 50-50 split.

(b) Comparison of ridge regression with and without feature transformation. The improvement is very little and might be insignificant.

(c) Learning curve. Blue is training data and red is test data.

Figure 3: