Project-I by Group Boston

Ivan Baeriswyl
EPFL
ivanbenjamin.baeriswyl@epfl.ch

Simon Noetzlin
EPFL
simon.noetzlin@epfl.ch

Abstract

In this report is exposed the methodology and the results for our first PCML project, which was divided into a regression and a classification part. In the regression part, we found out that it was possible to divide our data into 2 clusters, on which we applied ridge regression (with polynomial transformation on the second cluster). In the classification part, the data-set show us no evident clues of correlation between output variables and input features. We used logistic regression with gradient descent for our model prediction. We got no improvement by penalizing it. Finally, we did some features clearance and polynomial transformation in our samples data to reach the most accurate classification.

1 Regression part

1.1 Data Description

The data set is composed of N = 2800 samples in the training set and 1600 samples in the testing set. The train-data consists of an output vector y and input variables X. Each input vector x_n has D = 66 features. Out of these features, 55 are real-valued, 3 are binary, 4 are categorical with 3 categories and 4 with 4 categories. The test-data does not contain the output y.

1.2 Exploratory Data Analysis and Data Cleaning

We perform basic exploratory data analysis on our data. The histogram in Figure 1(a) and the scatter plot in Figure 1(b) show us that the data seems to contains two distributions.

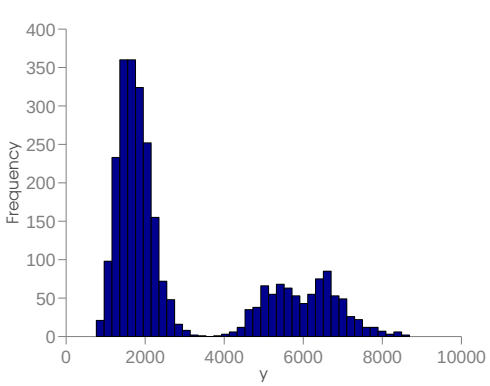
We separate the data into the two clusters and search which features are helping separate the clusters and figure out that the 6^{th} feature shown in Figure 1(c) is clearly separating them. We do not spot any significant outliers while observing the histograms of both clusters.

Standard normalization is applied on each one of our features in our input data X using the formula $\frac{X_k - \mu}{std}$, where X_k is the input vector containing the k^{th} feature, μ its mean and std its standard deviation. Note that both clusters are normalized independently.

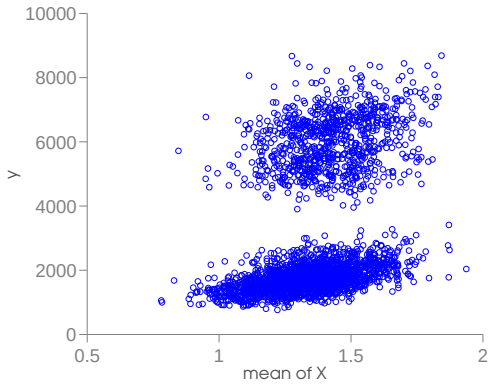
1.3 Ridge Regression

As the input data matrix X is ill-conditioned, least-squares is not suitable. We thus apply ridge-regression. We use k-fold cross-validation to assess our models, with k = 10 (not lower to get a high enough assessment and not higher for computational convenience).

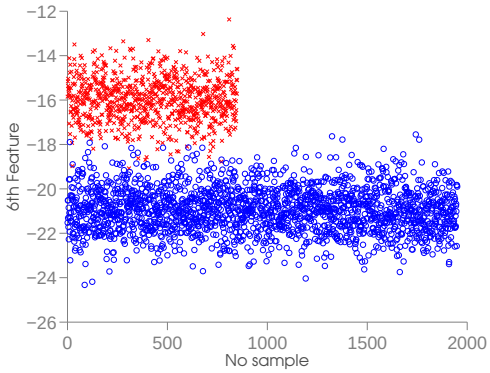
For the first cluster, we vary the λ values from 10^{-2} to 10^{3} and we choose our λ where the RMSE is the lowest, as you can see on the Figure 2(a). For the second cluster, we notice that applying a feature transformation lower the RMSE. The values of λ are taken from 10^{-2} to 10^{2} for the second cluster.



(a) Histogram of the output data y.



(b) Plot of the output y with the mean of X.

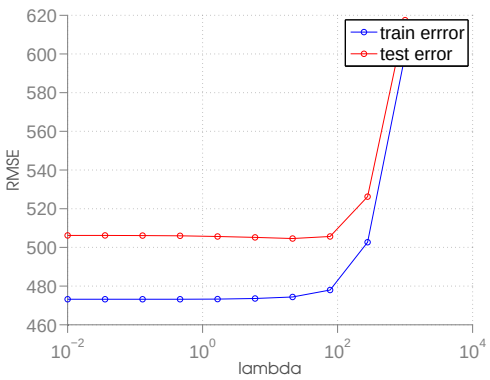


(c) The 6^{th} feature is separating the two clusters.

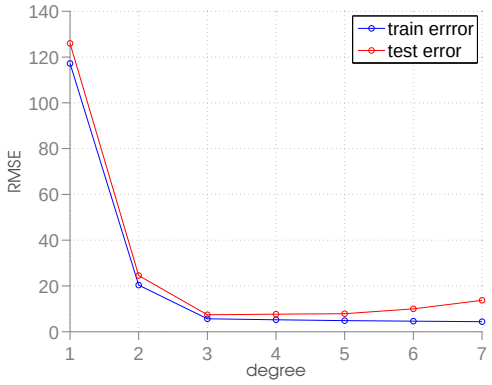
Figure 1:

1.4 Feature transformations

We test ridge regression with and without polynomial transformation. As stated on the Figure 1.4, we get better results for the cluster 2 using polynomial transformations. However it does not improve results for the cluster 1.



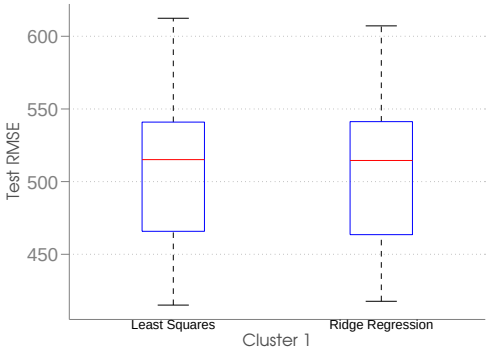
(a) RMSE on cluster 1.



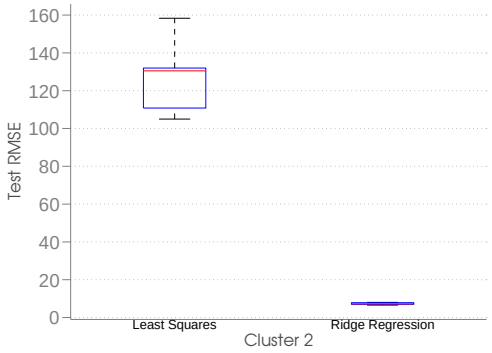
(b) RMSE by degree on cluster 2.

Figure 2:

The results shown in Figure 3(a) and 3(b) put in evidence the better expected performance for ridge regression (with polynomial transformations on the second cluster).



(a) Comparison of least squares and ridge regression on cluster 1.



(b) Comparison of least square and ridge regression with polynomial transformation on cluster 2.

Figure 3:

2 Classification part

2.1 Data Description

The training set is composed of input variables X_train and output variables y_train. There are N = 1500 data examples and each input comes with D = 31 features. Three of them are binary one is categorical and the remaining ones are real valued. The output vector y is binary taking initially the values $\{-1, 1\}$, we decide to change them to $\{0, 1\}$ for computational convenience.The testing-data are input variables X_test with N = 1500 data samples. Each of them have a dimensionality of D = 31 as our training-data matrix.

2.2 Visualization and Data Analysis

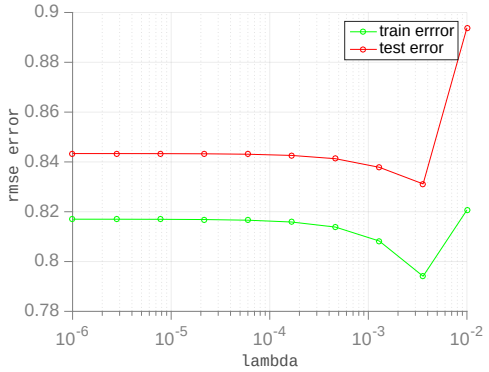
We analyse our variables by visualization before any processing. Plotting each inputs features individually shows no particular trends or correlation between them. Real valued features are mostly Gaussian-like distributed and have mean near 0 and a high standard deviation particularly the 6^{th} feature with std = 1320. Plotting outputs versus inputs give us no interpretable information, thus we compute their correlation coefficient between outputs and input features. With the highest values at ρ = 0.148 we decide to not consider this result in our future computation. We apply standard normalization on the input data for the algorithms to converge better.

2.3 Cross-validation and Logistic Regression

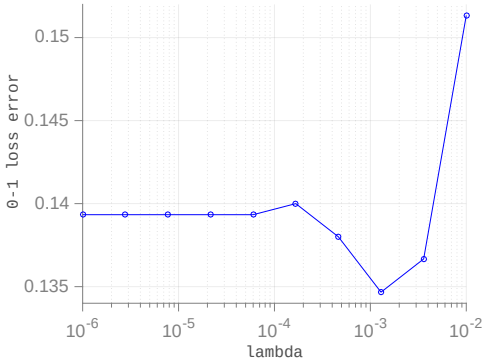
We use 10-fold cross-validation to have more realistic results in our predictions. The choice of our model is about computational time, indeed the second order derivatives of the Newton method is too slow for our test thus we decide to apply gradient descent for our logistic regression. Our first result can be more convincing with approximately 14% for 0-1 loss error and 0.81 for RMSE. Penalization is clearly our next try to get our classifier more accurate nevertheless it does not decrease our initial errors value. The figures 4(a) and 4(b) respectively show 0-1 loss error and RMSE according to λ.

2.4 Features modification

Since penalization does not improve our predictions we try some modifications in our features. We get better results using polynomial transformations. The Figure 5(a) shows that we get reduction of the 0-1 loss error at degree 2 but induce more variance as we can observe in Figure 5(b). Removing categorical and binary features from our input variables improves our test error as well what change the dimensionality of the our inputs data sample from D = 31 to D = 27.

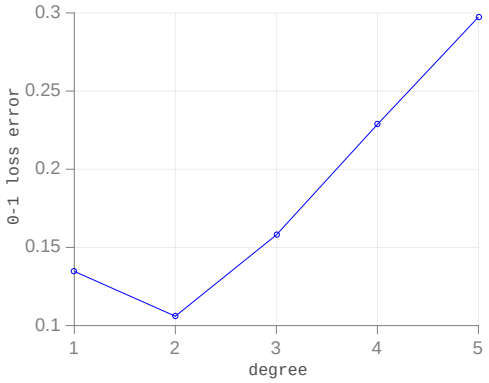


(a) RMSE error with cross-validation and Penalize logistic regression.

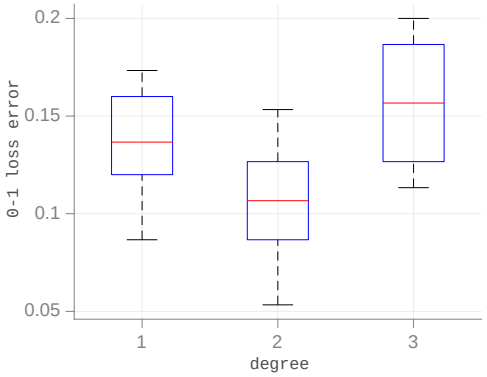


(b) 0-1 loss error with cross-validation and Penalize logistic regression.

Figure 4:



(a) 0-1 loss error after polynomial transformation and features clearance.



(b) Boxplots of 0-1 loss error with polynomial transformation shows some variance at degree 2.

Figure 5:

3 Summary

For the regression data-set, we separate the input variables in two clusters and use ridge regression for both models. We use a polynomial feature transformation for the second cluster. We get the expected RMSE errors of 500 and 10 respectively for model 1 and model 2. For the classification, we use a logistic regression model with gradient descent. Penalization does not improve it thus in consequence we apply some polynomial features transformation and get a better test error. In the end, we reduce our data example dimensionality by removing the categorical and binary features and get a final 0-1 loss error of 10.5%. Due to a lack of time, we do not implement the dummy variables and outliers removal for the classification which would probably enhance our model prediction.

Acknowledgments

We would like to thanks the TAs for their time and advices. The code has been done by both Ivan and Simon. The writing of the report as been shared as such: Ivan for the regression and Simon for the classification.

References