Predictive Modeling Process

Joseph Simonian & Ellen Hwang

November 4, 2016

Abstract

This report will be examining the applications of ridge regression (RR), lasso regression (LR), principal components regression (PCR), and partial least squares regression (PLSR). We will also be exploring how cross validation methods apply to each of these models and how cross validation strengthens our models.

Our project will be using the Credit dataset, taken from the webpage for the text, *An Introduction to Statistical Learning*, to perform this simple linear regression. We analyze credit card debt based on a number of factors, such as income, gender, and credit limit, and create predictive models for credit card debt through a number of linear regression models. This report will include a description of the data, methods, analysis, and results for the regression models created. # Introduction

In this project, we will be exploring 4 types of regression models that strengthen model interpretability and prediction accuracy.

Two of the regression models, ridge regression and lasso regression, are shrinkage methods which constrains the coefficient estimates to shrink the coefficient estimates towards zero. This helps reduce coefficient estimates reduce their variance. They differ in the application of shrinkage - lasso implements parameter shrinkage and variable selection, whereas ridge merely modifies parameter weights.

The other two regression methods, principal components analysis and partial least squares, are dimension reduction methods. Dimension reduction methods essentially reduce the number of predictors under consideration to strengthen a model. They differ in that PLS works to maximize inter-class variance in its low-dimensional model, whereas PCA maximizes the variance of features themselves.

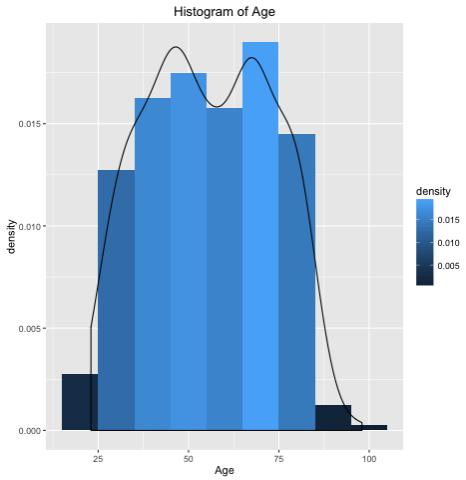
We will be applying these methods to the Credit Dataset to create various models predicting a person's Balance from various independent variables. We will discuss our methods in R, our various results, and conclusions about the relationships and statistics we find.

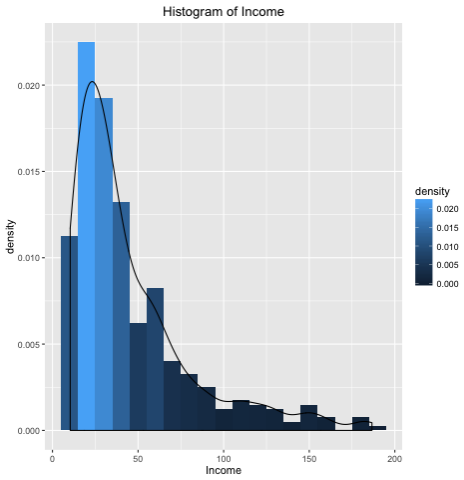
Data

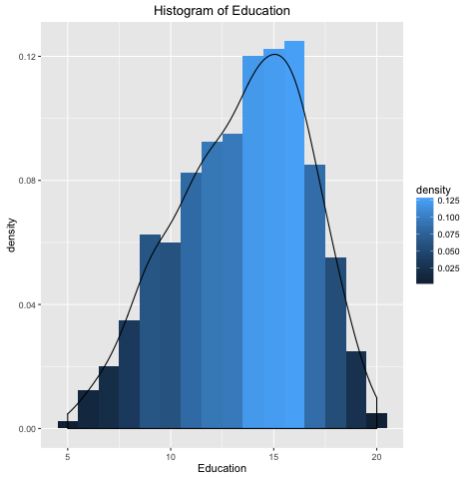
For this model, we use the Credit.csv dataset available from the Introduction to Statistical Learning website (download). This dataset is analyzed in Chapter 6 of the ISL book.

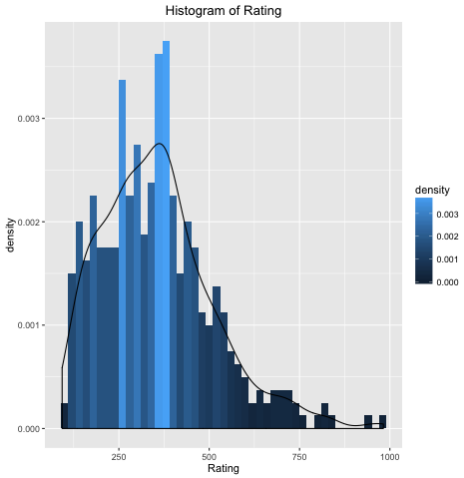
This dataset contains "Balance", representing average credit card debt across a group of people, and 10 explanatory variables, of which 6 are quantitative and 4 categorical. The quantitative variables are: Income, Credit Limit, Credit Rating, No. of Cards, Age and Years of Education. The categorical variables are: Gender, Student Status, Marital Status and Ethnicity.

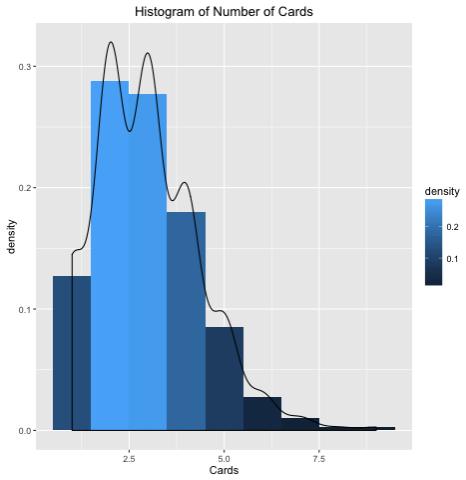
We include basic exploratory data analysis below:

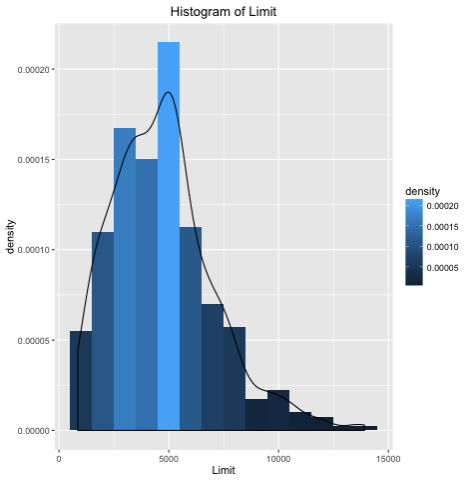

Histogram of Age
density
Age
0.015
0.010
0.005
0.000
25
50
75
100

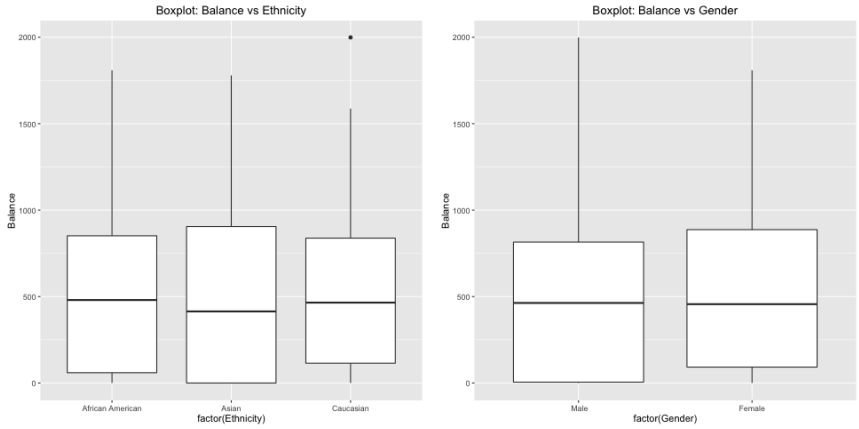

Histogram of Income
density
Income
0.020
0.015
0.010
0.005
0.000
0
50
100
150
200

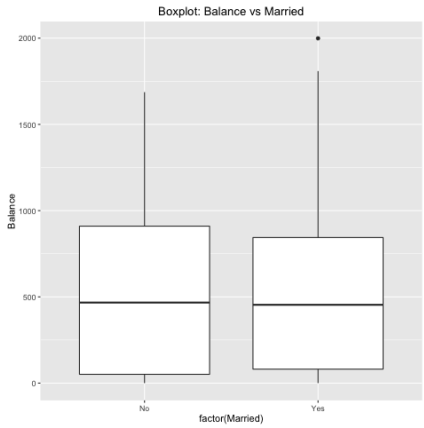

Histogram of Education
density
Education
0.12
0.08
0.04
0.00
5
10
15
20

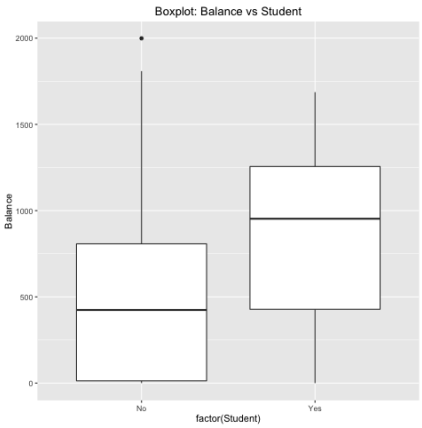

Histogram of Rating
density
Rating
0.003
0.002
0.001
0.000
250
500
750
1000


Histogram of Number of Cards
density
Cards
0.3
0.2
0.1
0.0
2.5
5.0
7.5

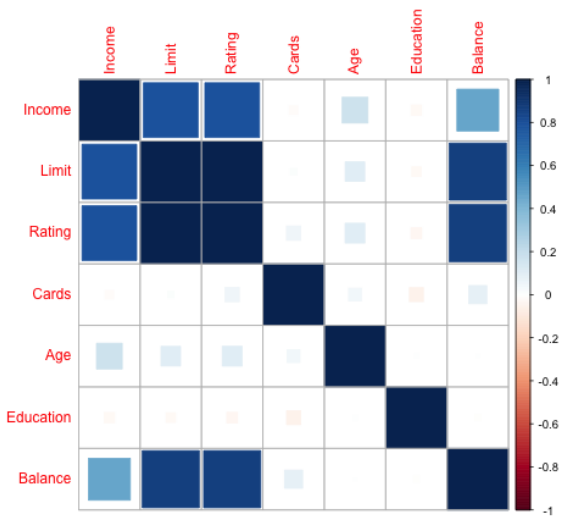

Histogram of Limit
density
Limit
0.00020
0.00015
0.00010
0.00005
0.00000
0
5000
10000
15000

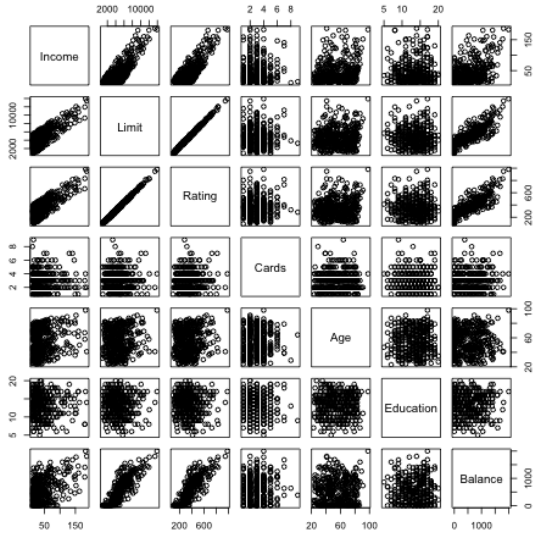






We also include correlation plots and scatterplots between quantitative variables:





Methods

Preprocessing

We use one-hot encoding to dummy out categorical variables, and perform mean-centering and standardization of all variables, setting their means to 0 and standard deviations to 1 to give us comparable scales.

We then split the data into a training set (75% of total data) and a testing set (25% of total data) for the purposes of tuning hyperparameters (such as λ in lasso and ridge regressions).

Lasso Regression

With lasso regression, we first create a preliminary model using `glmnet()` on the training dataset, with $\alpha = 1$ to represent the elasticnet mixing penalty used in lasso regression. We then cross validate our model on our training set using the function `cv.glmnet()`. During cross-validation, we perform a parameter search for lambda in the range $(10^{-10}, 10^{10})$, which was an expansion on the given grid from the project spec. Lastly, we fit a model to the full dataset and predict credit card balances using our best lambda found via CV.

Partial Least Squares Regression

With PLS regression, we use the `plsr()` function modeling *Balance* as a function of all other variables. To find the best number of components, we perform cross-validation and find the minimum of the validation PRESS features. Ultimately, we perform plsr regression on the full dataset using the best number of components found via cross-validation.

Ridge Regression

As with lasso regression, we create an initial model using `glmnet()` on the training dataset, with $\alpha = 0$ to represent the elasticnet mixing penalty used in ridge regression. We then cross validate our model on our training set using the function `cv.glmnet()`. Once again, we perform a parameter search for lambda using the extended grid $(10^{-10}, 10^{10})$. Finally, we fit a model to the full dataset and predict credit card balances using our best lambda found via CV.

Principal Components Regression

Similar to with PLS regression, in PCR we `pcr()` function from the `pls` package to implement our regression. Again, we perform 10-fold cross-validation to determine the optimal number of components, and then make predictions of the full dataset based on that number of components.

Analysis

```
load('../data/lasso-objects.RData')
load('../data/pls-objects.RData')
load('../data/ridge-objects.RData')
load('../data/pcr-objects.RData')
load('../data/ols-regression.RData')
```

Lasso Regression

When we perform cross validation and compute the associated test error we find that the MSE is quite low at 8.7098305×10^{-5}. There are also quite a few coefficients that are zero, as expected for lasso regression.

```
lasso.coef.official
```

```
##      Income       Limit      Rating       Cards         Age  StudentYes
## -0.55166063  0.92504680  0.36787493  0.04499772 -0.01666003  0.26681304
```

So all in all, we find that 6 predictors are relevant in lasso the regression: Income, Credit Limit, Credit Rating, Number of Credit Cards, Age, and Student Status - Education, Gender, Marital Status and Ethnicity all have coefficients of zero.

Ridge Regression

When we perform cross validation and compute the associated test error we find that the MSE fairly low, though still higher than that found with lasso regression, at 4.5388093×10^{-4}. There are also quite a few coefficients that are zero, as expected for lasso regression.

```
ridge.coef
```

```
##        (Intercept)             Income              Limit
##       0.000000e+00      -5.513681e-01       7.813897e-01
##             Rating              Cards                Age
##       5.111911e-01       3.882763e-02      -1.676057e-02
##          Education       GenderFemale         StudentYes
##       0.000000e+00      -1.062175e-06       2.660694e-01
##         MarriedYes     EthnicityAsian EthnicityCaucasian
##       0.000000e+00       0.000000e+00       0.000000e+00
```

Ridge regression generally has a lower probability than Lasso of setting unimportant components to 0, but as we see here, the coefficients Ethnicity, Marital Status and Education have all been set to 0. In addition, Gender has a very small coefficient, reinforcing our belief from Lasso regression that it is not predictive.

Principal Components Regression

Looking at the pls output summary, we find our lowest cross-validation error occurs when we use 9 components are used. The corresponding test set MSE = 0.0845579. We perform PCR using the full data set using M = 9 components (the number found in cross-validation), and obtain a model with the following coefficients:

```
pcr.fit.full$coefficients[,,pcr.best.mod]
```

```
##                   X             Income              Limit
##       5.062543e-05       2.463975e-01       2.875812e-01
##             Rating              Cards                Age
##       2.925997e-01       1.008403e-01      -1.278794e-01
##          Education       GenderFemale         StudentYes
##       8.008883e-03       1.666287e-02       2.443407e-01
##         MarriedYes     EthnicityAsian EthnicityCaucasian
##      -2.800629e-02      -7.539071e-03       9.134116e-03
```

Partial Least Squares Regression

Looking at the pls output summary, we find our lowest cross-validation error occurs when we use 3 components are used. The corresponding test set MSE = 0.0845854, very close to that found via PCR. We perform PLS using the full data set using M = 3 components (the number found in cross-validation). We obtain a model with the following coefficients:

```
pls.fit.full$coefficients[,,pls.best.mod]
```

```
##                  X             Income              Limit
##       0.0001853368      -0.1850805591       0.4891882953
##             Rating              Cards                Age
##       0.4903781918       0.1013777864      -0.1572646807
##          Education       GenderFemale          StudentYes
##       0.0168727387       0.0150581585       0.3794619506
##          MarriedYes     EthnicityAsian EthnicityCaucasian
##      -0.0416224070       0.0166119957       0.0095075827
```

Results

We include a table of regression coefficients from all models below, along with a table of MSE for teach model:

```
## Warning: package 'xtable' was built under R version 3.2.3
```

Table 1: Model Coefficients

	ridge	lasso	pls	pcr
(Intercept)	0.00	0.00	0.00	0.00
Income	-0.55	-0.55	-0.19	0.25
Limit	0.78	0.93	0.49	0.29
Rating	0.51	0.37	0.49	0.29
Cards	0.04	0.04	0.10	0.10
Age	-0.02	-0.02	-0.16	-0.13
Education	0.00	0.00	0.02	0.01
GenderFemale	-0.00	0.00	0.02	0.02
StudentYes	0.27	0.27	0.38	0.24
MarriedYes	0.00	0.00	-0.04	-0.03
EthnicityAsian	0.00	0.00	0.02	-0.01
EthnicityCaucasian	0.00	0.00	0.01	0.01

Table 2: Test MSE by Model

	Model	TestMSE
1	OLS	0.140492573384017
2	Ridge	0.000453880933182744
3	Lasso	8.70983052393698e-05
4	PCR	0.084557919452127
5	PLS	0.0845854044643395

From these results, we can clearly see that our worst model was simple ordinary least squares, followed by PLS and PCR, followed by Ridge and Lasso regressions, with Lasso having the overall lowest MSE.

Conclusions

Throughout this project, we have used a variety of linear models to predict credit card Balance based on owner data. We began by cleaning and preprocessing the data, and then performed a variety of different regression techniques on both the quantitative and (one-hot encoded) qualitative factors.

We found that the best predictors of credit card debt were not demographic but individual - factors such as income, credit rating, and student status. Demographic factors tended to be less predictive, and were indeed ignored by our final model.

From our data, it is clear that shrinkage methods such as Lasso and Ridge are most effective at modeling this dataset, with Lasso in particular performing well.