Abstract

We perform a predictive modeling process applied on the data set Credit in order to decipher how factors such as Income, Age, Education, and more influence the amount of credit card debt to a person. Our analysis is based off of Chapter 6 from *An Introduction to Statistical Learning*, "Linear Model Selection and Regularization".

Introduction

In this project, we applied model selection methods on different regression models introduced in Chapter 6: Linear Model Selection and Regularization (from **An Introduction to Statistical Learning** by James et al). The 5 regression models we practiced are *Ordinary Least Squares, Ridge Regression, Lasso Regression, Principle Components Regression and Partial Least Square Regression.* In order to find the best parameter (or coefficients) for those regression models, we splitted our datasets into train and test groups, performed 10-fold cross validation on each model respectively, and in the end chose the best model with the least cross-validation error.

Data

The Credit data set we will be using comes from **An Introduction to Statistical Learning**, contains the following variables: Income, Limit, Rating, Cards, Age, Education, Gender, Student, Married, Ethnicity and Balance. The Credit data set records Balance, the average credit card debt for an individual, according to quantitative predictors such as Income, Limit, etc. and qualitative predictors such as Student, Married, etc. Cards represents the number of credit cards a person has and Income is represented in the thousands of dollars. We investigate 400 individuals in this analysis.

Methods

Ordinary Least Square methods

In statistics, OLS is a method for estimating the coefficients in a linear regression model, with the goal of minimizing the sum of the squares of the differences between the observed responses in the given dataset and those predicted by the linear model. (i.e. $RSS = \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}(\beta_j x_{ij}))^2$, where $\beta_0 ... \beta_p$ are the coefficients estimate)

Ordinary Least Squares (OLS) This is a common liear regression model, and we use it as a benchmark to evaluate the performance of the other 4 regression models.

Shrinkage methods

Shrinkage methods invovles fitting a model with all P predictor. However, the estimated coefficients are shrunken towards 0 relative to the least squares estimates. Shrinkage has the effect of reducing variance. Since after the process of shrinkage, some of the coefficients might be exactly 0, it also performs variable selection.

Ridge Regression (Ridge)

Ridge regression is very similar to least square, except that the coefficients are estimatated by minimizing $RSS + \lambda \sum \beta_j^2$. This equation involves the minimization of two criterias, the first one is the same as OLS, ridge regersssion seeks to fit the data well by minimizing RSS. The second part of this equation, $\lambda \sum \beta_j^2$, is

called the *shrinkage penalty*. And the shrinkage penalty gets smaller as coefficients get closer to 0. λ here is called the tuning parameter, and $\lambda \geq 0$. When $\lambda = 0$, Ridge regression is the same as OLS. As λ grows larger, the impact of shrinkage penalty grows bigger as well. The use of tuning parameter λ is also where Ridge regression outperforms OLS, and the mechnaism's name is called *bias variance tradeoff*. As λ increases, the flexibility of the ridge regression fit decreases, leading to decreased variance but increased bias

Lasso Regression (Lasso)

Lasso regression is an alternative to Ridge and it overcomes oen of Ridge's disadvantage, that even though it uses all P predictors and even though it moves all coefficients close to 0, it won't set any of them to be exactly 0 unless λ is infinity and this disadvantage creates obstacles on model interpretation when P gets really large. Lasso regression minimizes $RSS + \lambda \sum |\beta_j|$. And it is the 2nd term that forces some of the coefficients estimates to be exactly 0.

Dimension Reduction methods

Dimension reduction works by *projecting* the P predictors onto a M-dimensional space, where M < P. This is achieved by computing M different linear combinations, or projections, of the variables. Then these M projections are used as predictors to fit a linear regression model by least squares.

Principal Components Regression (PCR)

The *first principal component* direction of the data is that along which the observation vary the *most*. The PCR approach involves constructing the first M principal components, $Z_1, Z_2, ..., Z_M$, and then using those principle components as predictors in a linear regression model that uses least squares method. This approach reduces dimension because of the fact that usually a samller amount ($M < P$) of principal components are sufficient to explain most of the variabilities in the dataset, as well as the relationship with the response. PCR is an *unsupervised* approach because in the process of finding the first M principal components that best describes the P predictors, the response Y is not used to help determine the principal component direction.

Partial Least Square Regression (PLSR)

PLSR is a *supervised* alternative to PCR. It first identifies a new set of features $Z_1, Z_2, ..., Z_M$ that are linear combinations of the original features, and then fits a linear model using least squares method with those M features. The difference is PLSR identifies $Z_1, Z_2, ...Z_M$ in a *supervised* way, which means it makes use of the response Y while identifying new features, therefore the new features not only approximate the old feature well enough, they are also related the response.

Analysis

Data Processing

In order to conduct our analysis, we first began with some pre-modeling data processing. The two main actions we performed before applying our models were:

1. Dummying out categorical variable
2. Mean centering and standardizing all the variables

For step 1, we took our categorical, or qualitative variables, namely `Student`, `Married`, `Gender`, and `Ethnicity` and *dummied* them out. What this means is that the values for the variables were factored and then given binary indicators. The reason for this is you can not apply regression models, specifically the lasso and ridge regressions that we used from the `glmnet` package, to variables that are non numeric. We used the `model.matrix()` function to perform step 1.

For step 2, we standardized our data so that each variable would have a mean of 0 and standard deviation of 1. The purpose of this is to standardize across different scales, so that coefficients such as $\hat{\beta}_0$ do not vary based on whether they're calculated in pounds or ounces, etc. To do so, we used the `scale()` function.

Training and Testing Data Sets

Another step we took before building our model was to take a set of data for model building and another for testing model performance. The set of data used to build the model, our *training set*, was 300 events randomly selected using the `sample()` function. For our *test set*, we randomly selected 100 events, this time using the `set.seed()` function for reproducibility.

Model Building Process

For each of our regressions we fit the model to our *training set* after performing 10-fold-cross-validation and resampling the data. For the lasso and ridge regressions, we use the function `cv.glmnet()` to apply the model and perform the cross validation whereas with the PCR and PLSR regression models, we simply use the `pcr()` and `plsr()` functions and set the `validation` argument to `"CV"`. After we have fit the functions to our data, we selected the best fit model looking for the minimum values of `$lambda.min` for the lasso and ridge regressions and of `$validation$PRESS` for the PLSR and PCR regressions. Once we selected our best models, `best_comp_num` in our PCR and PLSR regressions and `min_lambda` in our ridge and lasso regressions, we use our *test set* to compute the Mean Square Error to compare all of our best models. Finally to finish off our regressions we used the *full set* of data to get our *actual* coefficients for the models. The full data set was used in combination with our regression functions to compute summary statistics such as R^2, F-Statistic, and more.

Results

Exploratory Data Analysis Results

The results of our exploratory data analysis showed that certain variables in our study were much more important when examining the `Balance` of credit card debt. Two of the variables were highly correlated with the `Balance`, a person's credit `Limit` and credit `Rating`. This is understandable the higher a person's credit rating, generally the higher their credit limit is as well which means they are at liberty to spend more leading to higher balances. Below is the correlation matrix for all of the numeric variables:

	X	Income	Limit	Rating	Cards	Age	Balance
X	1	0.037	0.024	0.022	-0.036	0.059	0.0061
Income		1	0.79	0.79	-0.018	0.18	0.46
Limit			1	1	0.01	0.1	0.86
Rating				1	0.053	0.1	0.86
Cards					1	0.043	0.086
Age						1	0.0018
Balance							1

Table 1: Correlation Matrix

Regression Results

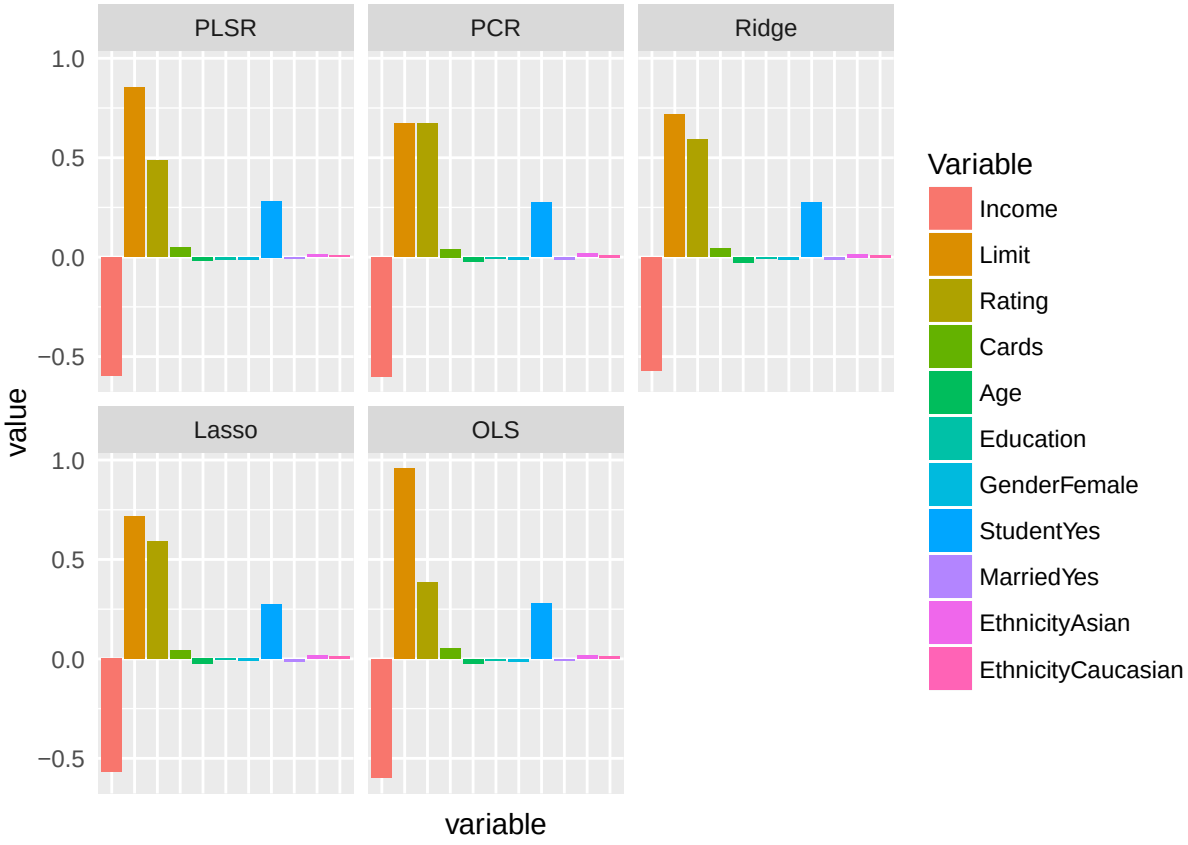
After applying our regressions to our full data sets, we received the following coefficients for our various regression models. While the coefficients obviously vary across the different models, we can notice trends

comparing all together. Across all regression models, the predictor that had the largest affect on Balance was `Limit`. Again across the PLSR, PCR, ridge and lasso regressions we can note that the estimates for $\hat{\beta}_4, \hat{\beta}_5, \hat{\beta}_6$ corresponding to `Age`, `Education`, and `Gender` respectively are all nearly 0. This indicates that these variables do not largely factor in to the `Balance` of a person's credit card debt. As credit card company's cannot discriminate in their providing of credit to people of different genders, ages, or with different amounts of education it makes sense that these estimates are lower.

	PLSR	PCR	Ridge	Lasso	OLS
Income	-0.60	-0.60	-0.57	-0.57	-0.60
Limit	0.85	0.67	0.72	0.72	0.96
Rating	0.49	0.67	0.59	0.59	0.38
Cards	0.05	0.04	0.04	0.04	0.05
Age	-0.02	-0.02	-0.03	-0.03	-0.02
Education	-0.01	-0.01	-0.01	-0.01	-0.01
GenderFemale	-0.01	-0.01	-0.01	-0.01	-0.01
StudentYes	0.28	0.28	0.27	0.27	0.28
MarriedYes	-0.01	-0.01	-0.01	-0.01	-0.01
EthnicityAsian	0.01	0.02	0.02	0.02	0.02
EthnicityCaucasian	0.01	0.01	0.01	0.01	0.01

Table 2: Information about Regression Coefficients

Another enlightening way to look at these coefficients is through a bar chart. The bar chart below shows the regression coefficients for each variable divided up by the regression model used. From this visualization, it's easy to note that `Income` is another strongly correlated variable, except negative. This means that an increase in `Income` is associated with a decrease in `Balance`. Therefore, when a person's income increases their credit card debt decreases, most likely because they can make larger payments towards their balance.



In order to see if the PLSR, PCR, Ridge, Lasso, or OLS regression is the best fit for our data set, we can examine the Mean Square Error to examine the quality of the regressions. While all the models had relatively close MSEs, the two that were the lowest were the PCR and OLS regressions. Therefore, it would appear these were the most suitable for predicting the Credit data set.

	Ridge	Lasso	PCR	PLSR	OLS
1	0.0411	0.0414	0.0397	0.0406	0.0397

Table 3: Mean Square Errors

Conclusions

In conclusion, the regression models that most accurately fit our data set were the OLS and PCR regressions. However, the difference in error between these and the other regressions was not statistically significant. Some of the most valuable insight came from comparing across all of the regression models. For example, we were able to see that variables such as credit limit, credit rating, and income drastically affected balance across all five regression models. Much of this data is aligned with intuitions regarding credit card debt, such as the higher your limit the higher your balance may be.