STA363SecAProj2

Rickey Huang

3/16/2021

Abstract/ Executive Summary

Section 1: Data Cleaning

In this section, the data set is explored and cleaned in order to improve the quality of the data for a good prediction.

Section 1.1: Cleaning the missing data

First of all, every row of data are inspected. However there is no missing data in this college data set. Since the header for the first column which shows the names of colleges are missed, I add a header (*"College"*) to a copy of the original data set I created and use this copy for the analysis below.

Section 1.2: Adjusting variables

From the infromation provided, I learn that the number of student enrolled in the colleges are usually not easily to be collected, the column stored this information is deleted from our data set. Also, since the acceptance rate is a more appropriate varaible than the numebr of acceptance, a new collumn named *"Rate"* is created using the existing variables acceptance *"Accept"* to be divided by the number of applications per academic year *"Apps"*. After adding this new variable to our data, since it is perfectly correlated to the variable *"Accept"*, the old and incomparable variable *"Accept"* is deleted. After arranging the variables, I got the data set for the analysis in this project, which has 777 observations and 18 variables, and among them only the variable *"Private"* is a categorical variable with two levels. Since the goal of this project is to predict the number of applications received during an academic year, the variable *"Apps"* would be the response variable in this project, and all other variables except the names of universities would be the exploratory variables. For the convenience of the analysis, I removed the column storing the college names, and change the variable *"Private"* to *"PrivateYes"*, which is a variable with 1 indicating private school and 0 for not a private school.

Section 2: Selection Only

In order to have a comparatively precise prediction in the end, several models are fitted compared in this project. This section focuses on the selection-only Least Square Linear Regression (LSLR) model, which also implements the Best Subset Selection (BSS) technique to refine the variables we have.

Section 2.1: Best Subset Selection - Stage 1

In the first stage of the BSS, all possible models containing 1 variable, 2 variables, and all the way to the full models (with 16 exploratory variables here) is created. R^2 is used to determine the best models among the models using the same amount of variables.

Section 2.2: Best Subset Selection - Stage 2

Proceeding to the second stage, I compared how well models created in the stage 1 are using the R^2_{adj}.

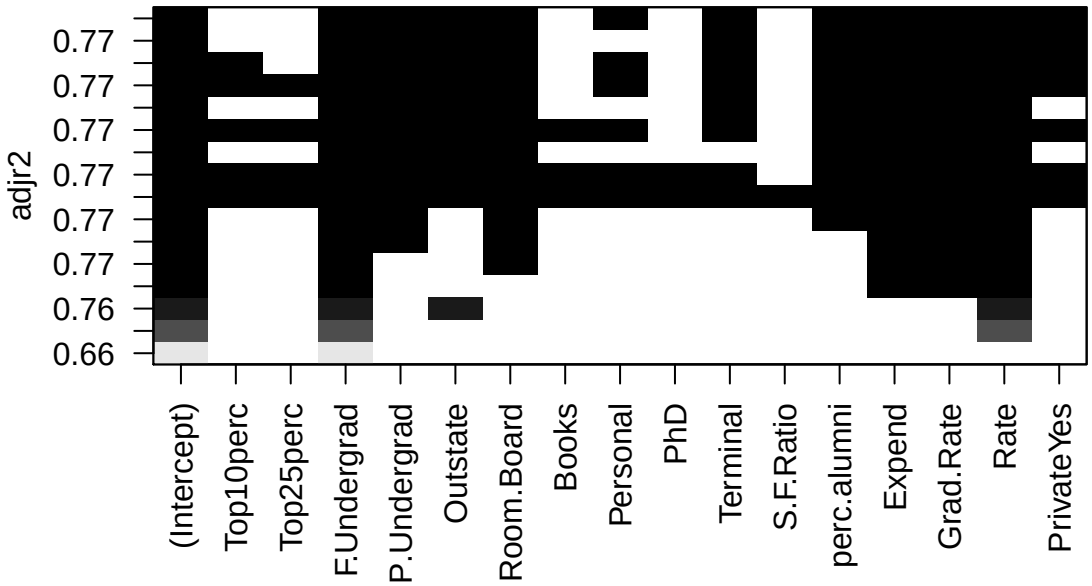

Figure 1: Adjusted R-squareds for models created in the stage 1 of BSS

Gathering the R^2_{adj} computed from the models, the Figure 1 is created to visualize the result. Since we want a model using less variables to explain the pattern in the data as much as possible, we prefer a model with a higher R^2_{adj} and less variables. The model with features *"PrivateYes", "F.undergrad", "P.Undergrad", "Outstate", "Room.Board", "Terminal", "perc.alumni", "Expend", "Grad.Rate", "Rate",* and the intercept with a R^2_{adj} of 0.7738017 is the best fit.

After the features for the LSLR model are chosen, the coefficients for these features can be calculated. In this LSLR model, the estimates are chosen by minimize the residual sum of squares (RSS), which is obtained by Formula 1, where Y is the vector storing all Apps for each row, and X_D is the design matrix.

$$RSS = (Y - X_D\hat{\beta})^T(Y - X_D\hat{\beta}) \tag{1}$$

As a result we get a model with coefficients as shown in the Table 1. Hence the final regression line gotten is $\widehat{Apps} = 1994.50 - 353.54PrivateYes + 0.66F.Undergrad - 0.16P.Undergrad + 0.08Outstate + 0.24Room.Board - 9.89Terminal - 20.15perc.alumni + 0.07Expand + 19.05Grad.Rate - 4812.59Rate$, which has an R^2_{adj} of 0.7738017.

Table 1: The estimates for the LSLR model

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1994.5038628	701.8798770	2.841660	0.0046072
PrivateYes	-353.5375145	240.8623931	-1.467799	0.1425694
F.Undergrad	0.6555742	0.0201696	32.503022	0.0000000
P.Undergrad	-0.1597435	0.0554461	-2.881058	0.0040742
Outstate	0.0838324	0.0328084	2.555209	0.0108046
Room.Board	0.2356826	0.0844043	2.792307	0.0053638
Terminal	-9.8869963	5.8877399	-1.679251	0.0935108
perc.alumni	-20.1480419	7.0000823	-2.878258	0.0041101
Expend	0.0689980	0.0187457	3.680732	0.0002489
Grad.Rate	19.0549834	5.0978967	3.737813	0.0001994
Rate	-4812.5857721	525.3775219	-9.160243	0.0000000

In order to further evaluating the model chosen, the 21-fold cross validation tachnique is use to assess the performance of the LSLR model in prediction by dividing our data into 21 folds of training data and test data. The reason that the k-fold cross validation is chosen is that the data we use is a comparatively large data set, if we use the LOOCV, the cross validation process will be computationally expensive. Hence, the k-fold could not result a low accuracy or a high variance variation, and it also compute the result faster than the LOOCV technique. As a result of the cross validation for the model, we compute the test RMSE for the

The shrinkage technique I use here for the second model is the Ridge Regression. Improved from the LSLR model, the metric RSS plus a penalty term is minimized here to choose better estimates. To be specific, the metric we are minimizing here is expanded in the Formula 2, where the $\lambda \geqslant 0$ is the tuning parameter and the $\lambda\hat{\beta}^T\hat{\beta}$ is the penalty term. By adding this penalty term, we can shrink the estimates and in turn lower the standard error of our model.

$$RSS + \lambda\hat{\beta}^T\hat{\beta} = (Y - X_D\hat{\beta})^T(Y - X_D\hat{\beta}) + \lambda\hat{\beta}^T\hat{\beta} \tag{2}$$

Section 3.2.2 Fitting the Ridge Model

In order to get an appropriate Ridge model, tuning parameters are chosen from 0 to 1000 by 0.5, and the models fitted with these parameters are trained using the 21-fold cross validation method, since this is a comparatively large data. The test MSE's are computed and plotted in the Figure 2. From the Figure 2, we can see that the test MSE keep increasing as the tuning parameter approaching 1000, so the range for the tuning parameter our client suggested is enough to choose a reasonable λ. Since the test MSE explains how far our estimation is away from the real data, we would like to choose λ with the lowest test MSE. The result we get from the cross validation is that the tuning parameter $\lambda = 73.5$ minimizes the test MSE which is 3567052.

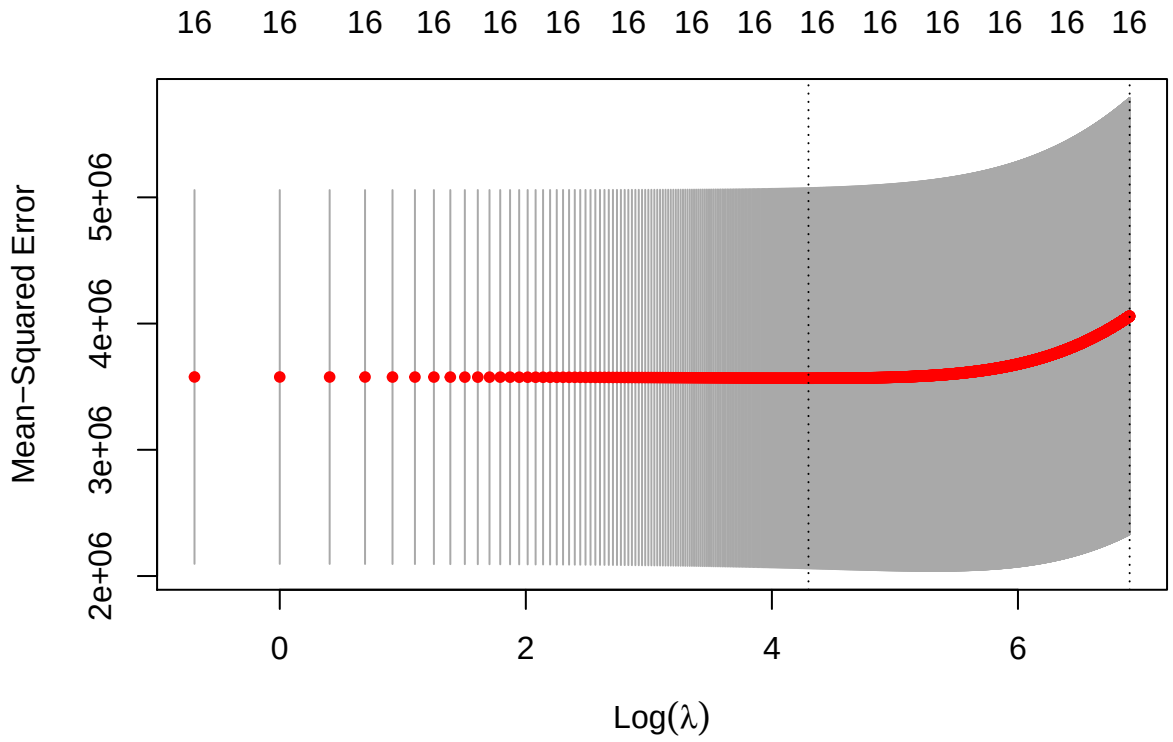

Figure 2: test MSE vs. log(Lamba) for the Ridge model

Section 3.2.3 The Estimates for the Ridge Model

With this parameter, the corresponding penalty term is add to the RSS to create the metric minimized. In this way, the coefficients for the final Ridge model are shown in the Table 2 with the coefficients for the full model without shrinkage placed side by side. Hence, the Ridge model can be written out as $\widehat{Apps} = 2053.32 + 8.48Top10perc - 2.39Top25perc - 449.81PrivateYes + 0.63F.Undergrad - 0.11P.Undergrad + 0.07Outstate + 0.24Room.Board - 0.17Books - 0.11Personal - 0.80PhD - 7.98Terminal + 6.04S.F.Ratio - 21.59perc.alumni + 0.07Expand + 18.63Grad.Rate - 4562.06Rate$.

Table 2: Comparing the Coefficients: Full model vs. Ridge model

	Full.Model	Shrinkage
(Intercept)	2347.9828469	2053.3203010
Top10perc	7.8638193	8.4805942
Top25perc	-4.0746483	-2.3875501
F.Undergrad	0.6573680	0.6285696
P.Undergrad	-0.1459711	-0.1101358
Outstate	0.0784838	0.0729730
Room.Board	0.2395610	0.2425719
Books	-0.2110964	-0.1702510
Personal	-0.1280348	-0.1100649
PhD	-0.9091787	-0.8032072
Terminal	-9.2029383	-7.9782245
S.F.Ratio	2.6902737	6.0486138
perc.alumni	-21.5745533	-21.5946430
Expend	0.0662131	0.0676779
Grad.Rate	18.1436018	18.6297680
Rate	-4740.8089022	-4562.0644359
PrivateYes	-347.5734261	-449.8102435

Section 3.3 Evaluating the Shrinkage Model

Taking the square root of the test MSE, the test RMSE for the Ridge model is compated, which is 1888.664. Comparing to the test RMSE for the full model which is 2014.712, the Ridge model improves 6.26% from the full model. The coefficients for the full model and the Ridge model are shown in the Table 2. From Table 2, the coefficient for *"F.Undergrad"* shrinks from -4.07 to -2.39, and the coefficient for *"Terminal"* also drop from -9.20 to -7.98. However, comparing to the LSLR model, which has a test RMSE of 1881.8, since both the models uses the 21-fold cross validation, the LSLR model performs better on the prediction. Hence, until this point, the LSLR model with only selection does a better job than that the Ridge model with only shrinkage.

Section 4: Selection and Shrinkage

Section 4.1: Reasons for doing both Selection and Shrinkage

Since the test MSE of the previous model drops 6.26% from the full model, the variance of the estimation is smaller. However, the ridge regression technique fits a more biased model to shrinkage the coefficients and variance. Also, since the Ridge Model keeps all variables in it, and a small set of exploratory variables is prefered for prediction, a selection technique could be add to the shrinkage process above to decrease the number of variables in our model. Therefore, the Lasso technique is a good fit for this situation, since it combine the advantage of the both the selection and shrinkage methods.

Section 4.2: The Lasso Model

Section 4.2.1: Details about the Technique

In order to do both the selection and shrinkage, the Lasso technique complete this task also by adding a penalty term. However, improved from the ridge regression, the penalty term is changed. In Lasso, the metric minimized is shown in the Formula 3, where the $\lambda_{Lasso} > 0$ is the tuning parameter for the lasso regression, k is the number of parameters in the model, which is 16 in this project.

$$RSS + \lambda_{Lasso}\|\hat{\beta}\|_1 = (Y - X_D\hat{\beta})^T(Y - X_D\hat{\beta}) + \lambda_{Lasso}\sum_{j=1}^{k}|\hat{\beta}_j| \tag{3}$$

Section 4.2.2: Fitting the Lasso Model

Like the modelling process in the ridge regression, the tuning variable λ_{Lasso} are chosen among 0 to 1000 by 0.5 to generate a Lasso model with the least test MSE. The plot shown the relationship between the test MSE and the log Lambda is created in the Figure 3. From the plot, the test MSE increases as λ_{Lasso} increases, so λ_{Lasso}'s are not neccessary to be tested to determine an appropriate tuning parameter. As shown in the result of the 21-fold cross validation, the least test MSE is from the model with a tuning parameter of 19, which results a test MSE of 3554176.

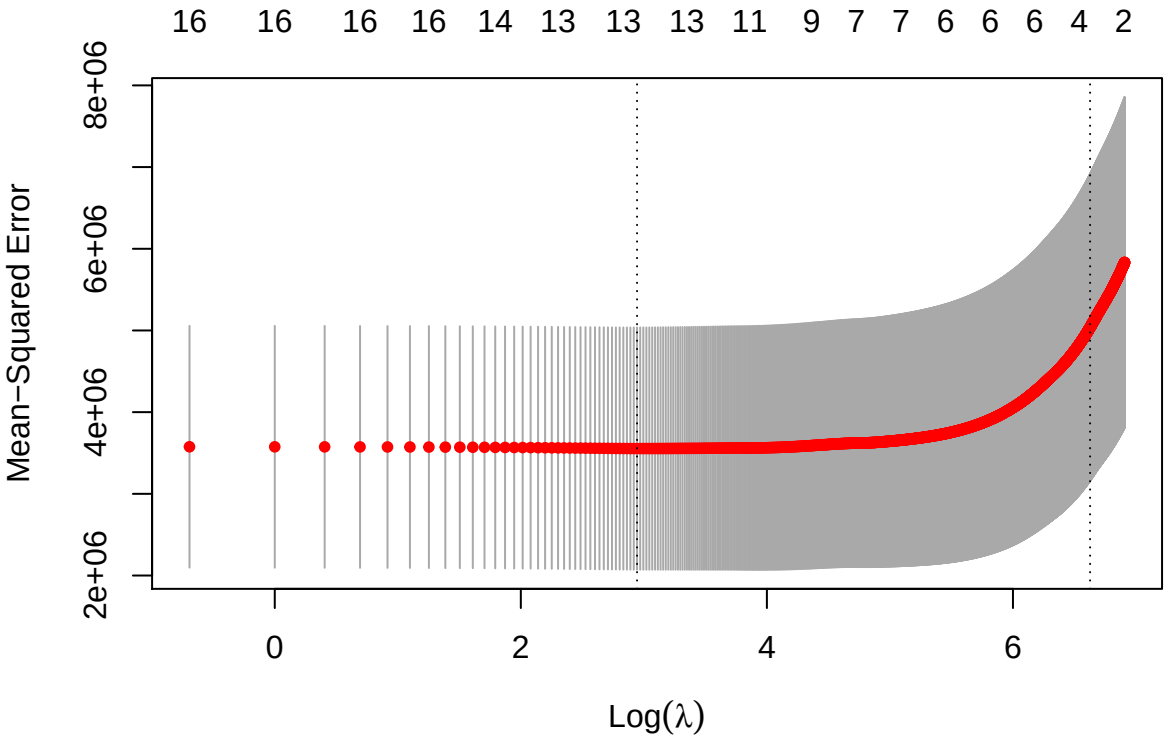

Figure 3: test MSE vs. log(Lamba) for the Lasso model

Section 4.2.2: The Estimates for the Lasso model

As the pernalty term is added to the metric for minimizing, the final model for the Lasso model used for the college data can be created. The coefficients for the lasso model are shown in the Table 3 with the parameters for the Full model and ridge model. Hence, the prediction line for the Lasso model can be written as following: $\widehat{Apps} = 1970.83 + 2.85Top10perc - 213.76PrivateYes + 0.65F.Undergrad - 0.12P.Undergrad + 0.06Outstate + 0.22Room.Board - 0.07Books - 0.10Personal - 5.88Terminal - 18.39perc.alumni + 0.06Expand + 17.18Grad.Rate - 4701.85Rate$.

Table 3: Comparing the Coefficients: Full model vs. Ridge model vs. Lasso model

	Full.Model	Shrinkage	Lasso
(Intercept)	2347.9828469	2053.3203010	1970.8274189
Top10perc	7.8638193	8.4805942	2.8468785
Top25perc	-4.0746483	-2.3875501	0.0000000
F.Undergrad	0.6573680	0.6285696	0.6494415
P.Undergrad	-0.1459711	-0.1101358	-0.1205914
Outstate	0.0784838	0.0729730	0.0633135

	Full.Model	Shrinkage	Lasso
Room.Board	0.2395610	0.2425719	0.2233467
Books	-0.2110964	-0.1702510	-0.0799176
Personal	-0.1280348	-0.1100649	-0.1005813
PhD	-0.9091787	-0.8032072	0.0000000
Terminal	-9.2029383	-7.9782245	-5.8787418
S.F.Ratio	2.6902737	6.0486138	0.0000000
perc.alumni	-21.5745533	-21.5946430	-18.3912000
Expend	0.0662131	0.0676779	0.0648436
Grad.Rate	18.1436018	18.6297680	17.1839692
Rate	-4740.8089022	-4562.0644359	-4701.8525447
PrivateYes	-347.5734261	-449.8102435	-213.7627808

Section 4.3: Evaluating the Lasso Model

As shown the Table 3, the coefficients for the variables are further shrinked from the ridge model, and the coefficients for some variables in the Lasso model drop to 0, which means such varaibles are not selected for prediction in the model. *"Top25perc", "PhD"*, and *"S.F.Ratio"* are dropped off from the full model in the Lasso model. For the same reason as in the Ridge model, the 21-fold cross validation technique is used to show the predictive accuracy of the Lasso model. By taking the square root of the test MSE, we can get the test RMSE for the models. The test RMSE for the Lasso model is 1885.252, which improves 21.95% from the full model which has a test RMSE of 2415.345.

Comparing the three models created so far, the LSLR model has the lowest test RMSE (1881.8), the test RMSE for the Ridge models is the highest (1888.664), and the Lasso model has a test RMSE in the middle (1885.252). Still at this step, the LSLR model is the best model for prediction for the data set given.

Section 5: Elastic Net

Section 5.1: Reasons for using the Elastic Net technique

As the client suggested, the Elastic Net is used to improve the model. As shown in the Table 3, the Lasso model just simply keeps one from strongly related variables and removes the rest of the variables from the full model. Hence, some correlations among variables that contributes to the prediction are simply ignored by the Lasso model. To fix this problem, the elastic net model could be a good choice.

Section 5.2: The Elastic Net Model

Section 5.2.1: Details about the Elastic Net Technique

the Elastic Net takes the advantage of both ridge regression and lasso technique, since it adds both pernalty in the previous two model to the metric for minimization. To be more specific, the metric minimized in the Elastic Net model is shown in the Formula 4, where the λ_{Elnet} is the tuning parameter for the Elastic Net model, and the α is the parameter that decides the model is more similar as the ridge regression model or the lassa model.

$$RSS + \lambda_{Elnet} \sum_{j=1}^{k} ((1-\alpha)\hat{\beta}_j^2 + \alpha|\hat{beta}_j|) \tag{4}$$

Section 5.2.2: Fitting the Elastic Net Model

When the model is fitted, several combinations of the λ_{Elnet} and α are used to fit the model. For the same reason, the 21-fold cross validation method is used to test the models different λ_{Elnet} and α's, and the best model with the lowest RMSE is chosen to fit the final Elastic Net Model.

Section 5.2.3: Estimates for the Elastic Net Model

Table 4: Comparing the Coefficients: Full model vs. Ridge model vs. Lasso model vs. Elatic Net model

	Full.Model	Shrinkage	Lasso	Elastic.Net
(Intercept)	2347.9828469	2053.3203010	1970.8274189	2012.2948856
Top10perc	7.8638193	8.4805942	2.8468785	5.4355978
Top25perc	-4.0746483	-2.3875501	0.0000000	-0.1753023
F.Undergrad	0.6573680	0.6285696	0.6494415	0.6322090
P.Undergrad	-0.1459711	-0.1101358	-0.1205914	-0.1097671
Outstate	0.0784838	0.0729730	0.0633135	0.0681927
Room.Board	0.2395610	0.2425719	0.2233467	0.2376621
Books	-0.2110964	-0.1702510	-0.0799176	-0.1330824
Personal	-0.1280348	-0.1100649	-0.1005813	-0.1060022
PhD	-0.9091787	-0.8032072	0.0000000	0.0000000
Terminal	-9.2029383	-7.9782245	-5.8787418	-7.8027002
S.F.Ratio	2.6902737	6.0486138	0.0000000	2.1517864
perc.alumni	-21.5745533	-21.5946430	-18.3912000	-20.6252852
Expend	0.0662131	0.0676779	0.0648436	0.0670443
Grad.Rate	18.1436018	18.6297680	17.1839692	18.1880286
Rate	-4740.8089022	-4562.0644359	-4701.8525447	-4584.7702978
PrivateYes	-347.5734261	-449.8102435	-213.7627808	-387.0064176

Section 6: Conclusion