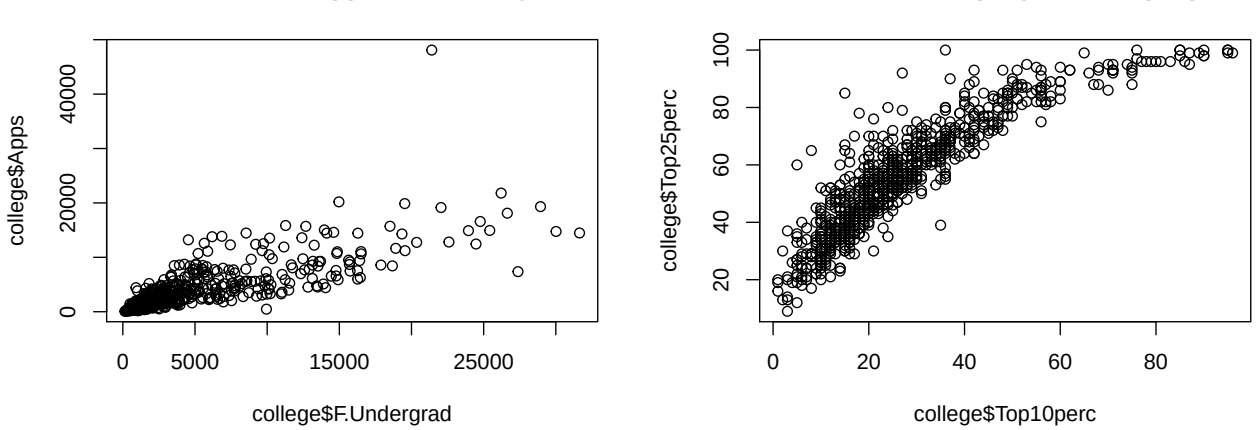
# Analysis 3

*Seline Yang, Alex Osier, Nick Clinch, Terry Lee*

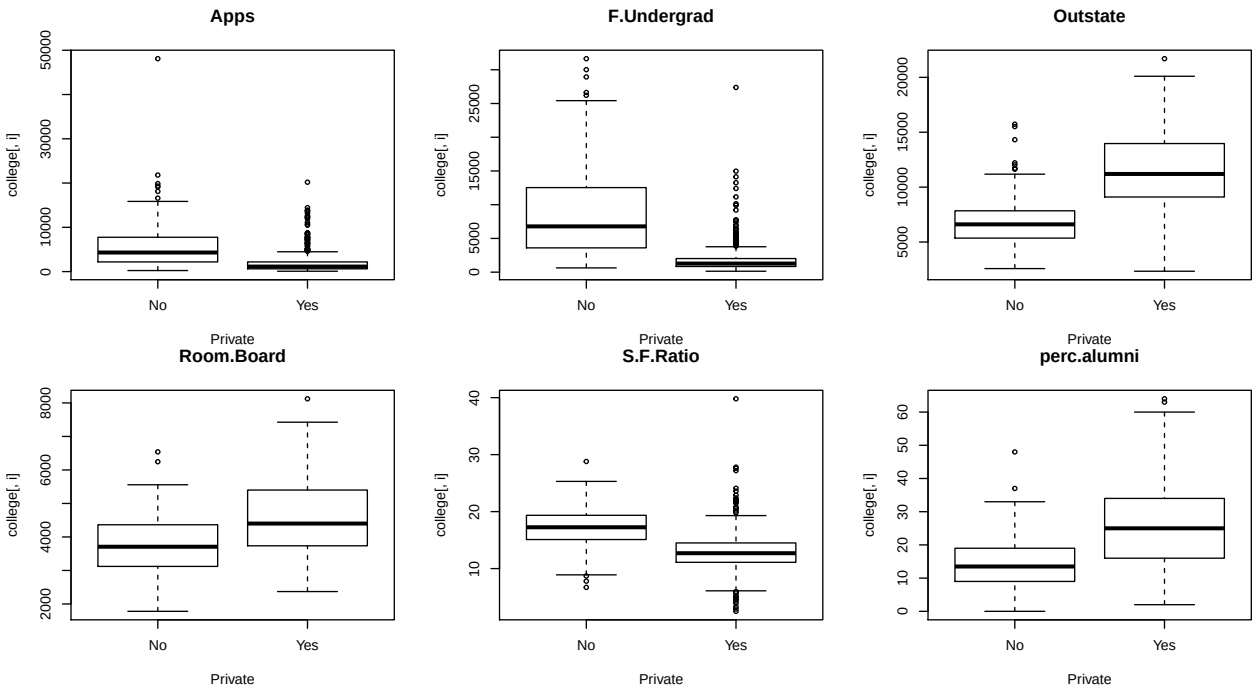
*2/20/2018*

## Section 1: Data Processing and EDA

There are 19 variables in the dataset, two of which are categorical and the rest of which are quantitative. The correlation matrix of quantitative variables indicated that F.Undergrad is highly correlated to Apps(.81), and Top10perc to Top25perc(.89). Here are the scatter plots.

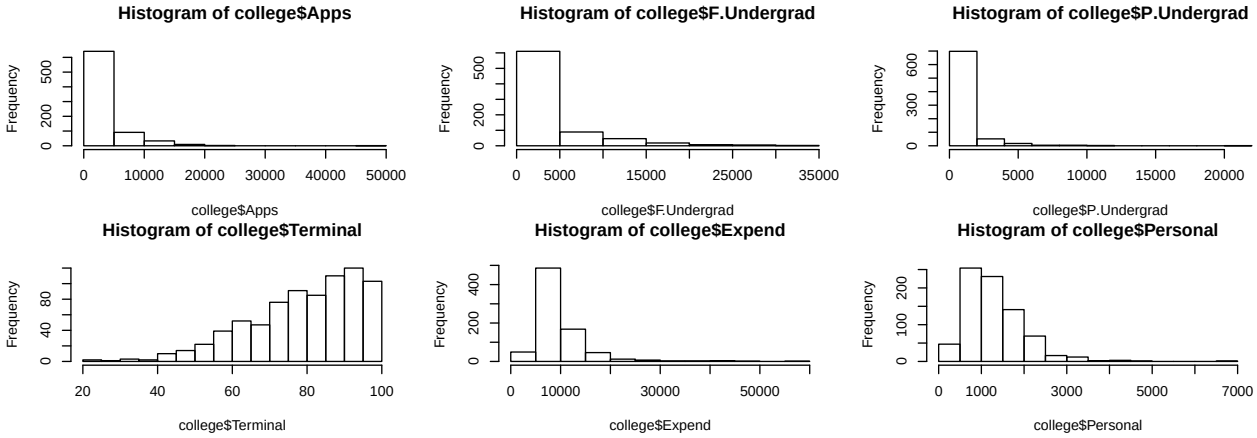



Then, we generated 17 boxplots for Private versus all other quantitaive variables. Private and public schools showed noticeable differences in Apps, F.Undergrad, Outstate, Room.Board, S.F.Ratio, and perc.almuni.




By looking at the data distribution from each variable, we found that Apps, F.Undergrad, P.Undergrad, Terminal, Expend, and Personal are highly skewed. So we might need to do some transformations for these

data for fitting the models.




# Section 2: The Full Model

```
##
## Call:
## lm(formula = Apps ~ ., data = college18)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8096   -673   -126    435  31852
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.437e+03  8.953e+02   3.840 0.000134 ***
## PrivateYes  -3.665e+02  2.372e+02  -1.545 0.122721
## Accept      -2.024e+03  6.298e+02  -3.213 0.001369 **
## Enroll      -6.167e+03  7.526e+02  -8.194 1.07e-15 ***
## Top10perc    1.958e+01  9.585e+00   2.043 0.041446 *
## Top25perc   -7.884e+00  7.575e+00  -1.041 0.298299
## F.Undergrad  6.519e-01  2.044e-02  31.898  < 2e-16 ***
## P.Undergrad -1.196e-01  5.454e-02  -2.193 0.028591 *
## Outstate     1.376e-02  3.294e-02   0.418 0.676212
## Room.Board   1.151e-01  8.373e-02   1.375 0.169554
## Books       -2.999e-02  4.074e-01  -0.074 0.941338
## Personal    -9.217e-02  1.069e-01  -0.862 0.388809
## PhD         -4.683e+00  7.876e+00  -0.595 0.552292
## Terminal    -1.064e+01  8.639e+00  -1.231 0.218596
## S.F.Ratio    9.134e+00  2.219e+01   0.412 0.680684
## perc.alumni -1.809e+01  6.889e+00  -2.626 0.008812 **
## Expend       7.577e-02  2.133e-02   3.552 0.000405 ***
## Grad.Rate    1.641e+01  5.016e+00   3.271 0.001122 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1767 on 759 degrees of freedom
## Multiple R-squared:  0.7962, Adjusted R-squared:  0.7916
## F-statistic: 174.4 on 17 and 759 DF,  p-value: < 2.2e-16
```

```
## [1] 3313172
```

$R^2$ of the Full model is 0.7962, which means 79.62% of variability in Apps was explained by our full model. We used 10-fold CV to estimate the test MSE because of its bias-variance balancing property and computational efficiency. The estimated test MSE is 3313172, which is extremly huge. Since some of the variables are highly skewed, so we might need to do some transformation in order to reduce the MSE. So we suggested to use log transformation for skewed and wide distributions.

```
##
## Call:
## lm(formula = Apps ~ Private + Accept + Enroll + Top10perc + Top25perc +
##     log(F.Undergrad) + log(P.Undergrad) + Outstate + Room.Board +
##     Books + Personal + PhD + Terminal + S.F.Ratio + perc.alumni +
##     Expend + Grad.Rate, data = college[, cbind(-1, -2)])
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4302  -1120   -300    736  37030
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.391e+04  1.305e+03 -10.660  < 2e-16 ***
## PrivateYes       -3.900e+02  3.210e+02  -1.215   0.2247
## Accept           -5.592e+02  7.939e+02  -0.704   0.4814
## Enroll           -6.596e+03  9.594e+02  -6.875 1.29e-11 ***
## Top10perc         3.138e+01  1.224e+01   2.564   0.0105 *
## Top25perc        -4.955e+00  9.667e+00  -0.513   0.6084
## log(F.Undergrad)  2.636e+03  1.487e+02  17.727  < 2e-16 ***
## log(P.Undergrad)  7.841e+01  7.555e+01   1.038   0.2997
## Outstate         -6.116e-02  4.250e-02  -1.439   0.1506
## Room.Board        9.330e-02  1.109e-01   0.841   0.4005
## Books            -2.883e-01  5.227e-01  -0.552   0.5814
## Personal         -7.736e-03  1.362e-01  -0.057   0.9547
## PhD              -1.272e+01  1.006e+01  -1.265   0.2062
## Terminal         -5.996e+00  1.100e+01  -0.545   0.5859
## S.F.Ratio        -4.713e+01  2.914e+01  -1.617   0.1062
## perc.alumni      -3.171e+00  9.044e+00  -0.351   0.7259
## Expend            6.962e-02  2.734e-02   2.547   0.0111 *
## Grad.Rate         5.475e+00  6.405e+00   0.855   0.3929
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2251 on 759 degrees of freedom
## Multiple R-squared:  0.6692, Adjusted R-squared:  0.6618
## F-statistic: 90.33 on 17 and 759 DF,  p-value: < 2.2e-16
```

```
## [1] 4948258
```

Thought it would be nicer to make log transformations of those varibles who are skewed. But once we changed, the $R^2$ is not improved, and so isn't MSE. But if we do a log transformation of the response variable, we noticed that $R^2$ has improved. Also, the training MSE is much more smaller than before.

```
confint(MFull2)
```

```
##                         2.5 %        97.5 %
## (Intercept)      -1.647041e+04 -1.134744e+04
## PrivateYes       -1.020243e+03  2.401582e+02
```

```
## Accept           -2.117731e+03  9.993210e+02
## Enroll           -8.479322e+03 -4.712540e+03
## Top10perc         7.351954e+00  5.540995e+01
## Top25perc        -2.393277e+01  1.402286e+01
## log(F.Undergrad)  2.344056e+03  2.927860e+03
## log(P.Undergrad) -6.989655e+01  2.267141e+02
## Outstate         -1.445882e-01  2.227785e-02
## Room.Board       -1.244315e-01  3.110376e-01
## Books            -1.314380e+00  7.378324e-01
## Personal         -2.751127e-01  2.596402e-01
## PhD              -3.246329e+01  7.018168e+00
## Terminal         -2.759548e+01  1.560337e+01
## S.F.Ratio        -1.043324e+02  1.007434e+01
## perc.alumni      -2.092540e+01  1.458246e+01
## Expend            1.596358e-02  1.232850e-01
## Grad.Rate        -7.098100e+00  1.804741e+01
```

Looking at the confidence interval for our model, we see that the range isn't great. While some variables have small confidence intervals, others do not, which makes us believe that we should perform varaible selection.

## Section 3: Subset Selection

### 3.

Two possible approximations to BSS are forward selection and backwards elimination. Forward selection starts with only the intercept and keeps adding a variable that results in the highest $R^2$ each step until $R^2$ cannot be improved by adding another variable. On the other hand, backwards elimination starts with the full model and keeps removing a variable likewise until $R^2$ cannot be improved by removing another variable.

### 4.

Data set with a large number of predictors would motivate us to use an approximation technique since BSS fits all possible $2^p$ models given p possible explanatory variables. With such a data set, BSS will be computationally expensive compared to approximation techniques.

### 5.

Since BSS would require that we fit $2^{17}$ models, it's essentially required that we use a different form of subset selection. Instead, we will use backwards selection to pick the best model.

```
##
## Call:
## lm(formula = Apps ~ Private + Accept + Enroll + Top10perc + F.Undergrad +
##     P.Undergrad + Room.Board + Terminal + perc.alumni + Expend +
##     Grad.Rate, data = college[, cbind(-1, -2)])
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8103   -654   -120    454  31767
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.233e+03  6.831e+02   4.734 2.63e-06 ***
## PrivateYes  -3.373e+02  2.198e+02  -1.535 0.125266
## Accept      -2.006e+03  6.052e+02  -3.314 0.000962 ***
## Enroll      -6.202e+03  7.215e+02  -8.595  < 2e-16 ***
## Top10perc    1.110e+01  5.793e+00   1.916 0.055723 .
## F.Undergrad  6.482e-01  1.991e-02  32.559  < 2e-16 ***
## P.Undergrad -1.283e-01  5.391e-02  -2.379 0.017587 *
## Room.Board   1.318e-01  7.866e-02   1.676 0.094166 .
## Terminal    -1.494e+01  5.684e+00  -2.629 0.008747 **
## perc.alumni -1.737e+01  6.671e+00  -2.603 0.009407 **
## Expend       7.835e-02  1.793e-02   4.370 1.41e-05 ***
## Grad.Rate    1.666e+01  4.884e+00   3.410 0.000683 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1763 on 765 degrees of freedom
## Multiple R-squared:  0.7955, Adjusted R-squared:  0.7926
## F-statistic: 270.5 on 11 and 765 DF,  p-value: < 2.2e-16
```

```
## [1] 3059196
```

The best model selected by backwards selection with the AIC criterion is of the form Apps ~ Private + Accept + Enroll + Top10perc + F.Undergrad + P.Undergrad + Room.Board + Terminal + perc.alumni + Expend + Grad.Rate.

### 6.

It isn't particularly surprising that Books, Personal, and Top25perc were removed by backwards elimination. The cost of books is minor compared to the total cost of attending a college and doesn't intuitively seem to be an important predictor. Similarly, both Top10perc and Top25perc should serve as measures of prestige for a school, so it isn't very surprising that one of the two would be eliminated during backward selection. However, the removal of S.F.Ratio is surprising, since one would imagine that a high ratio would usually indicate that the college is quite large and therefore would recieve a large number of applications.

### 7.

I would prefer the model $M_{LS}$ over the full model, since the smaller model is easier to interpret and is less likely to be overfit than the full model.

## Section 4: Ridge

### 8.

Ridge regression is a technique for analyzing multiple regression data that suffer from multicollinearity. When multicollinearity occurs, least squares estimates are unbiased, but the variance are large so they may be far from the true value.

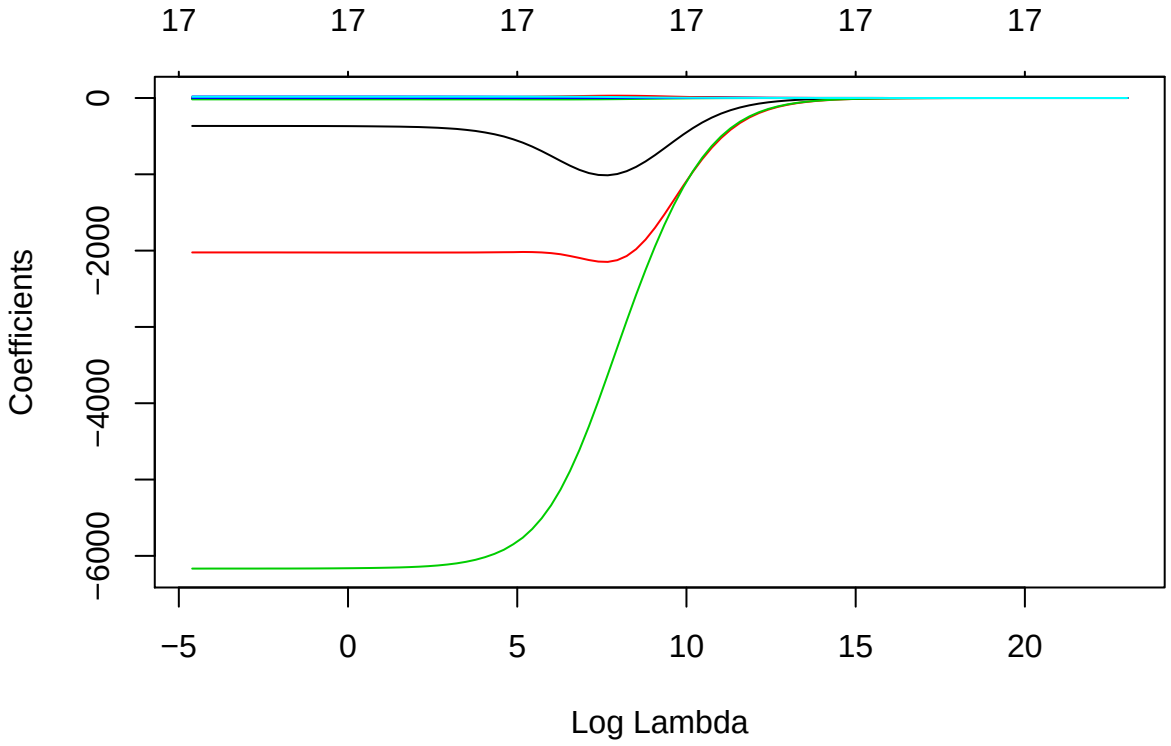
## 9.

To select the best value of $\lambda$, we will iterate, roughly, from $log(\lambda) = 6$ to 15 and compute the MSE for each value of $log(\lambda)$. Then we select the value with the lowest associated MSE as our $\lambda$ for Ridge Regression. We find that the most appropriate $\lambda$ would be 379.0722.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     315    3153   31536  354573  315281 3150214
```

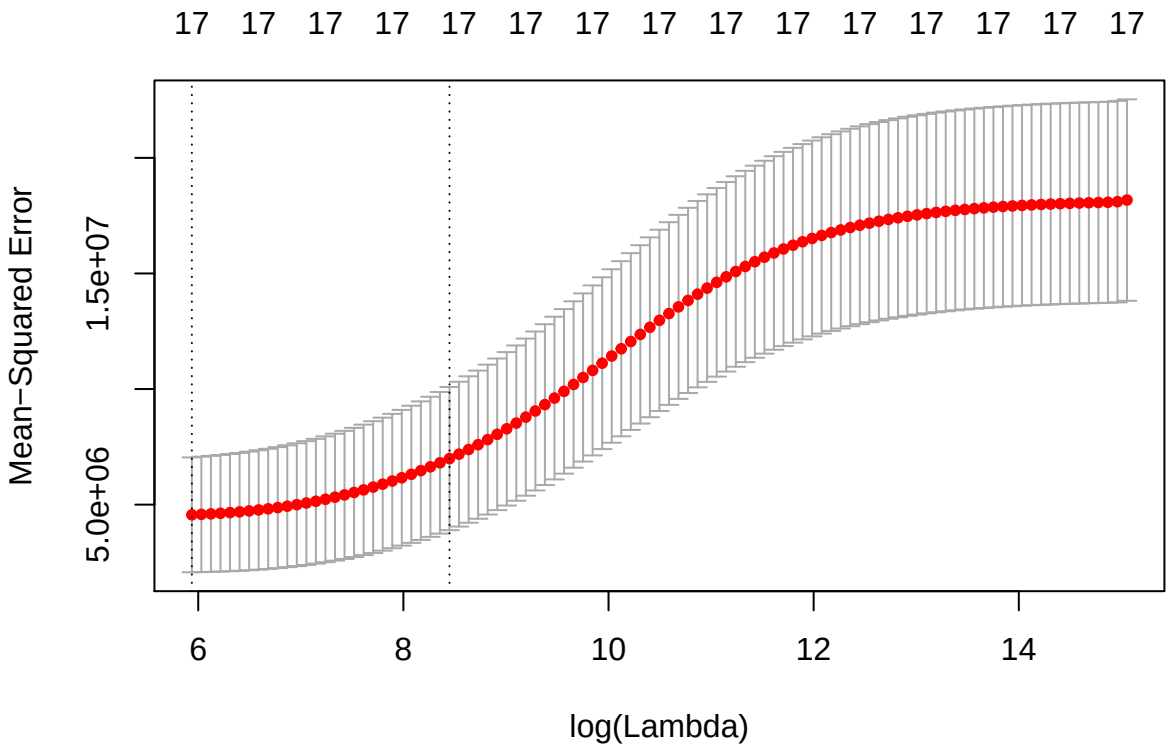
```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.000e+00 8.000e+00 8.697e+03 4.066e+08 9.326e+06 1.000e+10
```

```
## [1]  18 101
```




```
## [1] 1649958
```

```
## [1] 1650300
```




```
## [1] 379.0722
```

```
## [1] 379.0722
```

### 10.

The coefficient estimates with our choice of $\lambda$ are:

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                          1
## (Intercept)  2.444406e+03
## PrivateYes  -7.485253e+02
## Accept      -2.022950e+03
## Enroll      -5.376078e+03
## Top10perc    1.581031e+01
## Top25perc    1.361411e+00
## F.Undergrad  5.362143e-01
## P.Undergrad  1.033053e-02
## Outstate     1.487538e-02
## Room.Board   1.311480e-01
## Books        9.055307e-02
## Personal    -1.315212e-02
## PhD         -2.112615e+00
## Terminal    -6.222891e+00
## S.F.Ratio    1.860574e+01
## perc.alumni -1.868160e+01
## Expend       7.302781e-02
## Grad.Rate    1.709537e+01
```

### 11.

Compare the coeficients:

```
ridgeCoeff <- coef(ridge.final)
CoefMat <- cbind(coef(ridge.mod)[,101],coef(ridge.final))
colnames(CoefMat) <- c("LS", "Ridge")
CoefMat
```

```
## 18 x 2 sparse Matrix of class "dgCMatrix"
##                          LS         Ridge
## (Intercept)  3.437469e+03  2.444406e+03
## PrivateYes  -3.664805e+02 -7.485253e+02
## Accept      -2.023593e+03 -2.022950e+03
## Enroll      -6.166594e+03 -5.376078e+03
## Top10perc    1.957529e+01  1.581031e+01
## Top25perc   -7.883347e+00  1.361411e+00
## F.Undergrad  6.519264e-01  5.362143e-01
## P.Undergrad -1.196211e-01  1.033053e-02
## Outstate     1.376149e-02  1.487538e-02
## Room.Board   1.151304e-01  1.311480e-01
## Books       -2.999210e-02  9.055307e-02
## Personal    -9.216657e-02 -1.315212e-02
## PhD         -4.683057e+00 -2.112615e+00
## Terminal    -1.063670e+01 -6.222891e+00
## S.F.Ratio    9.134350e+00  1.860574e+01
## perc.alumni -1.808988e+01 -1.868160e+01
## Expend       7.577497e-02  7.302781e-02
## Grad.Rate    1.640584e+01  1.709537e+01
```

```
## [1] 1661867
```

Looking at the MSE of Ridge, we see that it is lower than the MSE of the full model by almost a third. However, the coefficients don't really tell a story. Since the process is automated by R, I believe that it is putting more "weight" on coefficients that are important compared to those that are not. This leads me to believe that we need some variable selection as well. This leads us to Lasso.

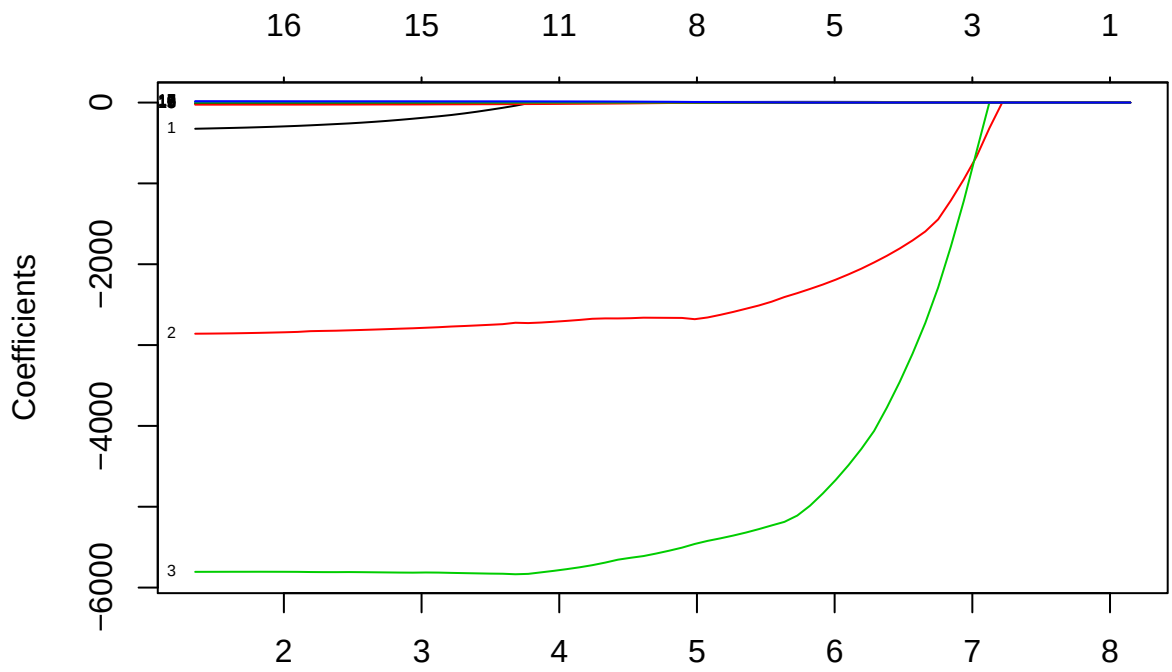
## Section 5: Lasso

### 12.

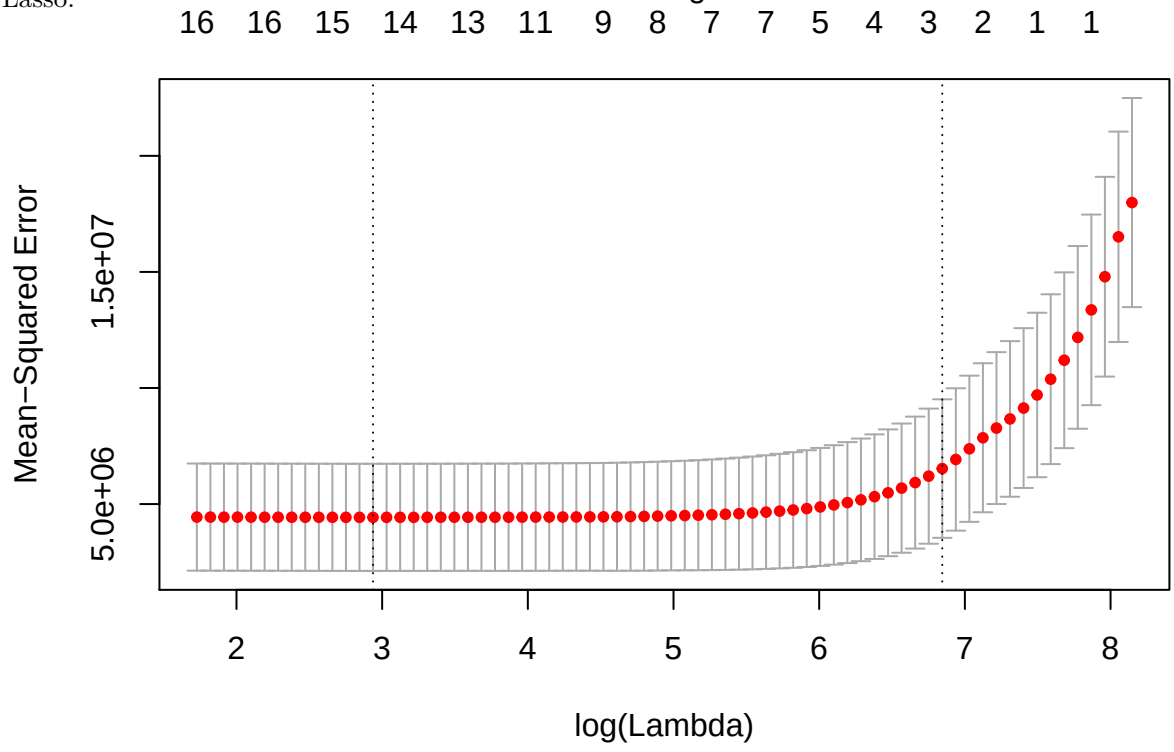
Lasso combines the variable selection abilities of BSS with the shrinkage properties of ridge regression to find the best fit of $\hat{\beta}$. Lasso can yield either a more accurate or more interpretable model than ridge regression.

### 13.

To select the best value of $\lambda$, we will iterate from $log(\lambda) = 1$ to 8 and compute the MSE for each value of $log(\lambda)$. Then we select the value with the lowest associated MSE as our $\lambda$ for



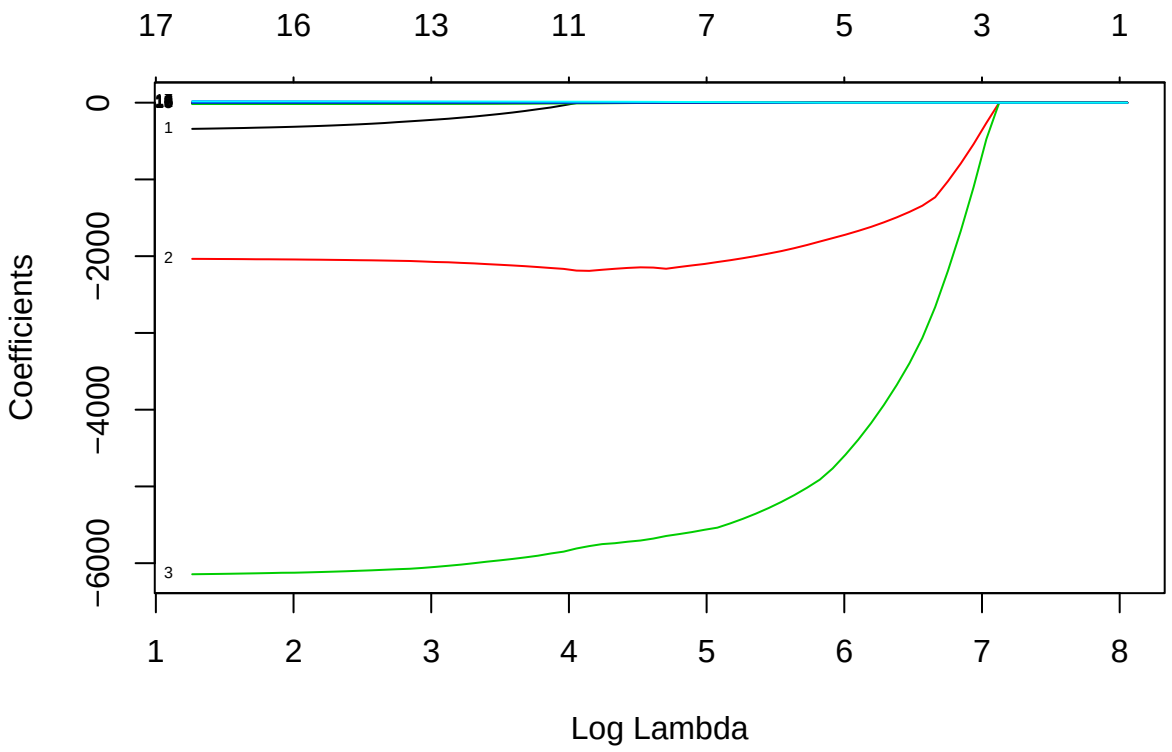

Lasso.




```
## [1] 18.86651
```

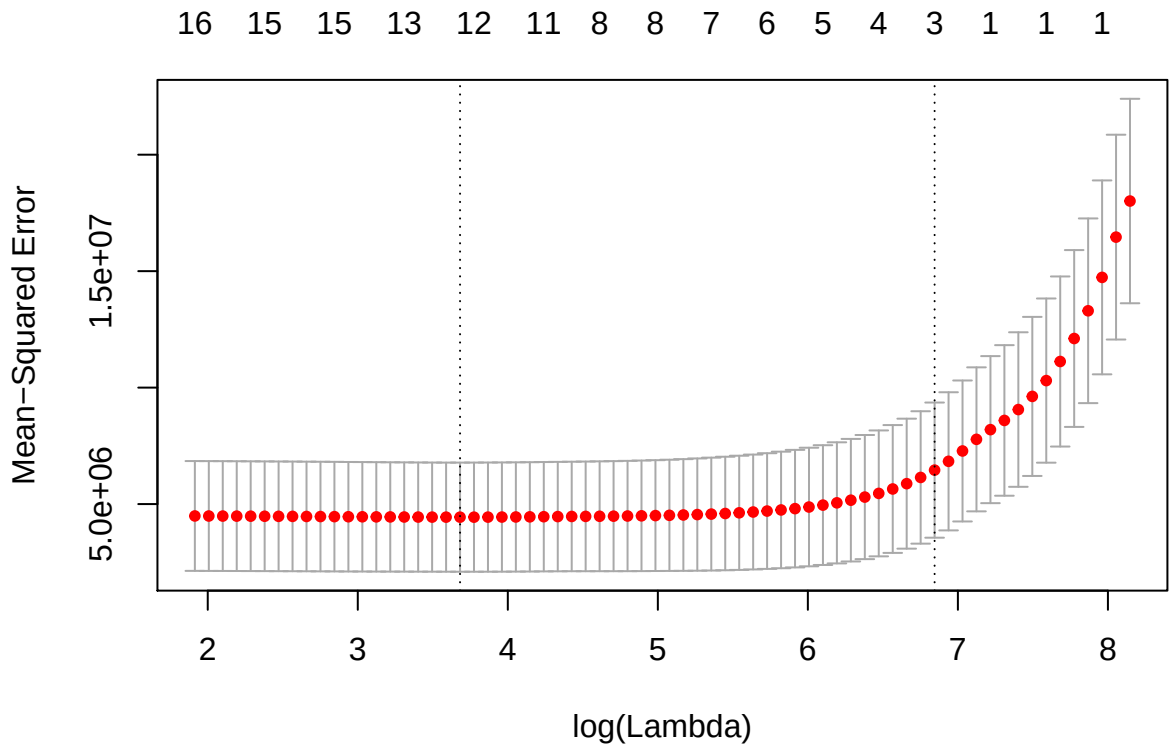
Break

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    3.539   19.345  105.697  478.705  577.183 3150.213
```




```
## [1] 1646463
```

```
## [1] 1646569
```




```
## [1] 39.71221
```

## 14.

Once again, the coeffiecient estimates that we get from Lasso for our choice of $\lambda$ are:

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                          1
## (Intercept)  3.170034e+03
## PrivateYes  -2.348724e+02
## Accept      -2.065756e+03
## Enroll      -6.056054e+03
## Top10perc    1.019169e+01
## Top25perc    .
## F.Undergrad  6.443733e-01
## P.Undergrad -9.634336e-02
## Outstate     .
## Room.Board   1.010010e-01
## Books        .
## Personal    -6.260583e-02
## PhD         -1.575451e+00
## Terminal    -9.545107e+00
## S.F.Ratio    .
## perc.alumni -1.540052e+01
## Expend       7.397968e-02
## Grad.Rate    1.513819e+01
```

ALTERNATIVE LASSO

```
outLasso<-glmnet(xinfo, college$Apps, alpha=1, lambda=cv.out$lambda.min)
lasso.coef2<-predict(outLasso, type = "coefficients", s=cv.out$lambda.min)
lasso.coef2
```

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                          1
## (Intercept)  2.883949e+03
## PrivateYes  -1.113663e+02
## Accept      -2.130733e+03
## Enroll      -5.919179e+03
## Top10perc    8.491677e+00
## Top25perc    .
## F.Undergrad  6.374004e-01
## P.Undergrad -6.937379e-02
## Outstate     .
## Room.Board   7.410085e-02
## Books        .
## Personal    -2.168738e-02
## PhD          .
## Terminal    -6.153935e+00
## S.F.Ratio    .
## perc.alumni -1.234306e+01
## Expend       6.799275e-02
## Grad.Rate    1.388476e+01
```

**15.**

```
CoefMat <- cbind(ridgeCoeff,lasso.coef)
colnames(CoefMat) <- c("Ridge", "Lasso")
CoefMat
```

```
## 18 x 2 sparse Matrix of class "dgCMatrix"
##                      Ridge         Lasso
## (Intercept)  2.444406e+03  3.170034e+03
## PrivateYes  -7.485253e+02 -2.348724e+02
## Accept      -2.022950e+03 -2.065756e+03
## Enroll      -5.376078e+03 -6.056054e+03
## Top10perc    1.581031e+01  1.019169e+01
## Top25perc    1.361411e+00  .
## F.Undergrad  5.362143e-01  6.443733e-01
## P.Undergrad  1.033053e-02 -9.634336e-02
## Outstate     1.487538e-02  .
## Room.Board   1.311480e-01  1.010010e-01
## Books        9.055307e-02  .
## Personal    -1.315212e-02 -6.260583e-02
## PhD         -2.112615e+00 -1.575451e+00
## Terminal    -6.222891e+00 -9.545107e+00
## S.F.Ratio    1.860574e+01  .
## perc.alumni -1.868160e+01 -1.540052e+01
## Expend       7.302781e-02  7.397968e-02
## Grad.Rate    1.709537e+01  1.513819e+01
```

Lasso recommends removing Top25Perc, Outstate, Books, and S.F.Ratio from the model. Both coefficient estimates for Outstate and Books were already very close to zero in our Ridge model, while the ones for Top25Perc and S.F.Ratio were small but not as close to zero.

```
ridge.pred <- predict(ridge.mod, s=cv.out$lambda.min, newx = xinfo[-CV_train,])
mean((new_college$Apps[-CV_train]-ridge.pred)^2)
```

```
## [1] 1568388
```

```
lasso.pred <- predict(outLasso, s =bestlamLasso, newx = xinfo[-CV_train,])
mean((new_college$Apps[-CV_train]-lasso.pred)^2)
```

```
## [1] 1568058
```

Comparing the MSE ridge to the MSE lasso, we see the lasso has a lower MSE. We would almost expect this since the lasso has variable selection as well as dealing with how variance and giant coefficients.

## Section 6: Choosing a model

The final model that we have chosen to work with is the model selected by Lasso. Unlike the full model and Ridge Regression, the lasso model carries out subset selection. Furthermore, the explanatory variables removed by subset selection agree with the variables eliminated by backwards selection with the AIC criterion, which implies that the subset selection of Lasso was carried out in a smart manner. Lastly, the Lasso model performs shrinkage, which the BSS model fails to do. Thus, the Lasso model provides the best combination of subset selection, RSS minimization, and shrinkage.

The final model is of the form:

Apps ~ PrivateYes + Accept + Enroll + Top10perc + F.Undergrad + P.Undergrad + Room.Board + Personal + PhD + Terminal + perc.alumni + Expend + Grad.Rate

## Section 7: Executive Summary / Abstract

In this analysis, we explore the relation between the number of applications that a college receives and a number of possible explanatory variables. We then compare the effectiveness of several modelling techniques and compare their relative effectiveness for both prediction and interpretability. We first start with fitting the full model and explore the impact of some basic transformations on the MSE. Then, we move on to more advanced methods like psuedo-Best Subset Selection (i.e. Backwards Elimination with the AIC criterion) and Ridge Regression to see the effects of variable selection and shrinkage on our model. Ultimately, we perform and settle upon Lasso as the optimal tradeoff of variable selection, shrinkage, accuracy, and interpretability.