HW 3: Predictive modeling for crime rate in a particular neighborhood

By: Christina Valore and Henry Vazquez

RMD file: https://github.com/hvasquez81/DATA621/blob/master/DATA621-Homework3.Rmd

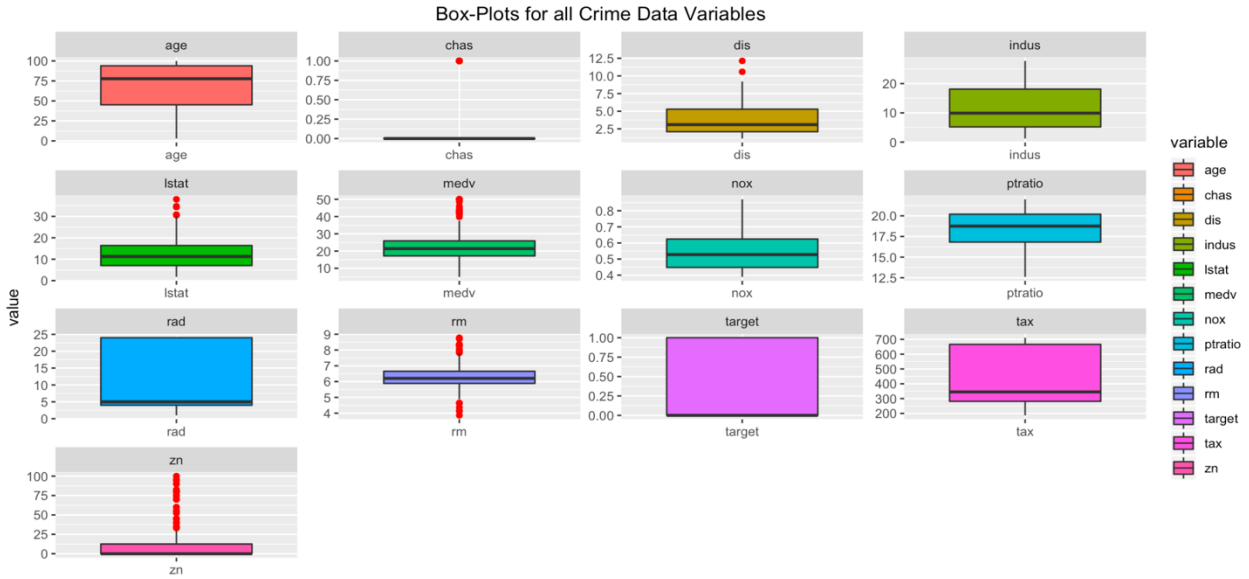
Data exploration

For the training data set named crime-training-data-modified.csv, there are 466 total observations each with 12 different predictor variables and 1 response variable. The evaluation set named crime-evaluation-data-modified.csv, has the same variables minus the response variable and only 40 observations.

Below is the mean, median, min, max and standard deviations for all variables:

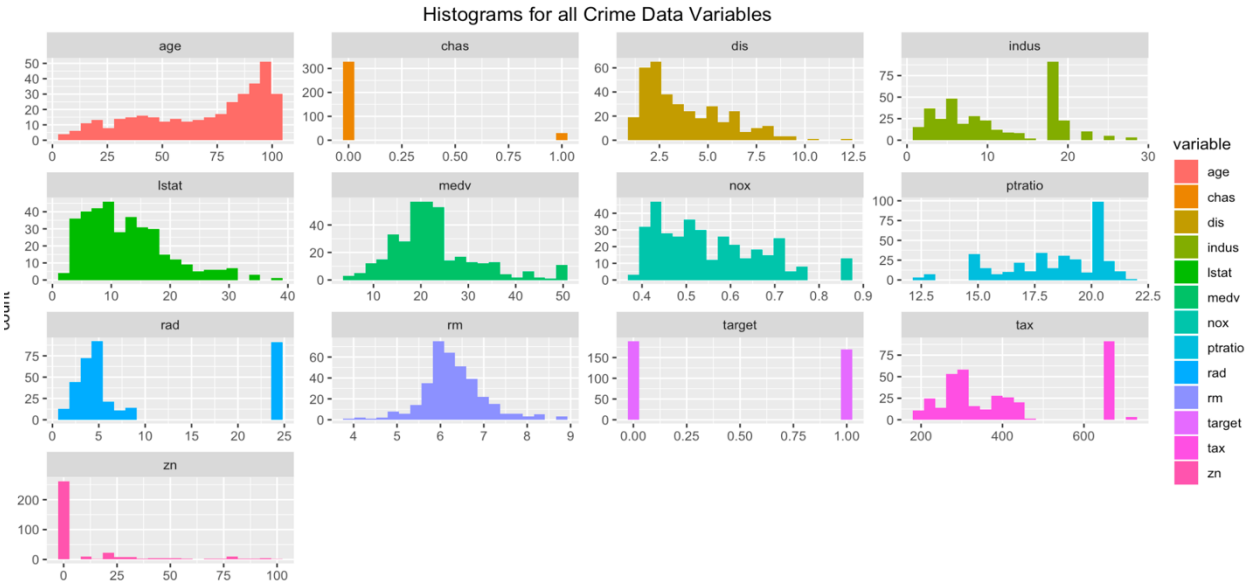
	n	mean	sd	median	min	max
zn	358	11.4372	23.3464	0.0000	0.0000	100.0000
indus	358	11.1906	6.7727	9.9000	1.2200	27.7400
chas	358	0.0838	0.2775	0.0000	0.0000	1.0000
nox	358	0.5537	0.1183	0.5280	0.3890	0.8710
rm	358	6.2880	0.7116	6.2050	3.8630	8.7800
age	358	68.3888	28.0611	77.7500	2.9000	100.0000
dis	358	3.7875	2.0782	3.1073	1.1296	12.1265
rad	358	9.4246	8.6286	5.0000	1.0000	24.0000
tax	358	408.9190	165.2676	345.0000	187.0000	711.0000
ptratio	358	18.3517	2.1724	18.7500	12.6000	22.0000
lstat	358	12.5390	7.0768	11.3000	1.7300	37.9700
medv	358	22.6760	9.0525	21.4000	5.0000	50.0000
target	358	0.4721	0.4999	0.0000	0.0000	1.0000

Next we looked at the boxplots of our variables to spot outliers:



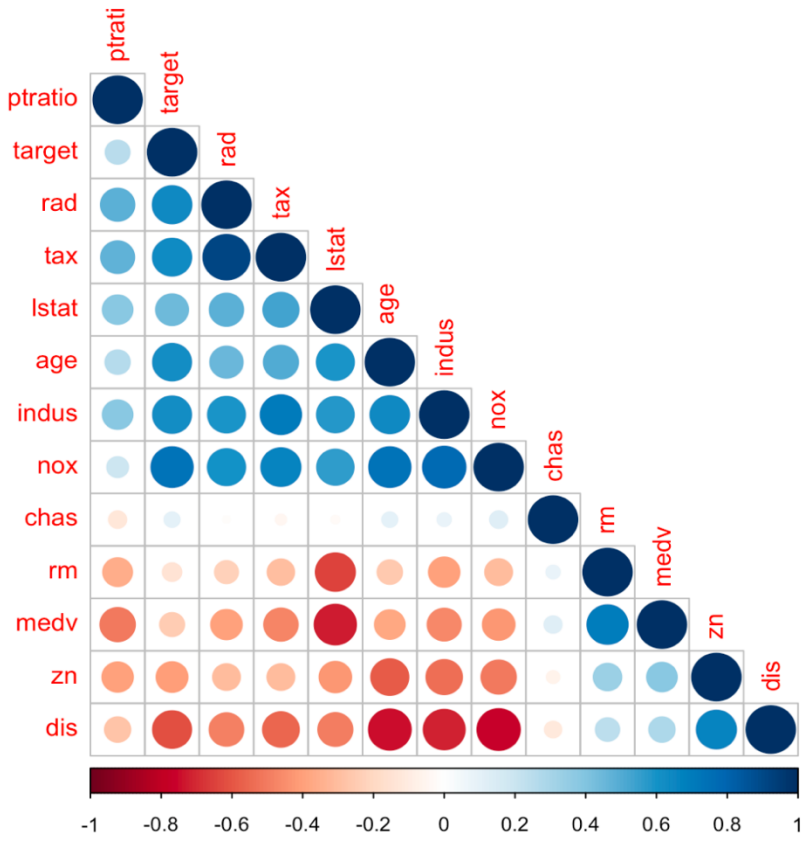
There doesn't seem to be a huge issue among the variables as far as outliers. There are a few exceptions in the data in which outliers are present outside of the 1st and 4th quartiles. For example, the variables zn, rm, dis, lstat and medv have apparent outliers. The variable chas has 1 outlier present, but it's a factor so this can be ignored.

Next we took a look at the histograms of our variables:



Looking at the histograms produced for all variables, some appear to have a normal distribution while others do not. Medv and rm are the 2 variables that appear to be normally distributed while the rest are either bimodal or multimodal, skewed, or just factors.

Finally, we look at the correlation between our variables:



Above is a lower correlation matrix showing the correlation between all variables. Blue being positively correlation and red meaning negatively correlated and the size of the circle implying how intense. The matrix is also ordered by correlation, meaning the positively correlated variables are shown at the top of the triangle and the negatively correlated variables at the bottom. Those variables above the nox row are almost all strongly positively correlated. While those under the rm row and before the nox column are mostly strongly negatively correlated.

To look at the variables that are correlated to the target variable, look at the target column (column 2). The variables between the rad and nox rows are positively correlated to the target variable. The variables mdv, zn and dis are negatively correlated to the target variable. The variables rm, chas, and ptratio have little to no correlation with the target variable.

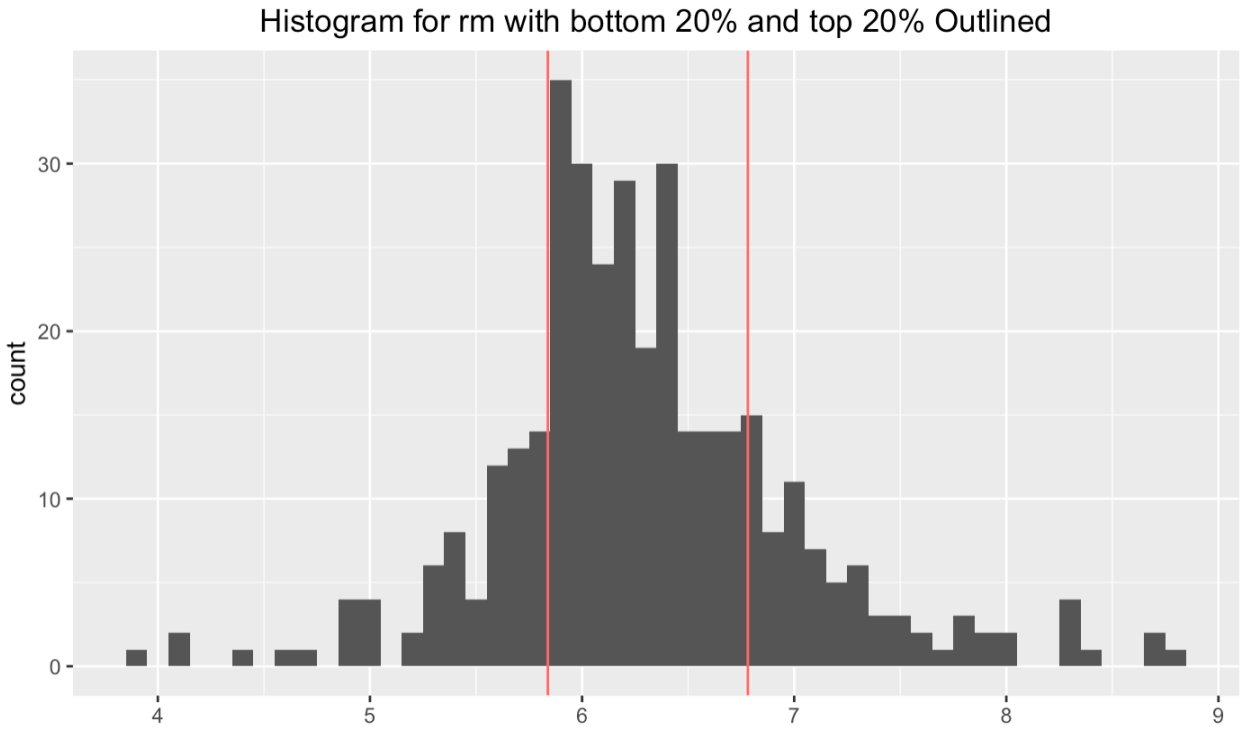
This variables with no correlation to the target variable will want to be avoided when building the logistic model. Also, variables that are strongly correlated with one another should not be included in the model or else this will violate the assumptions of logistic regression.

Data Preparation

Since the data did not have missing values, we do not need to worry about fixing any NA's. There are some variables we can possibly exclude from the analysis since they may not be relevant to the purpose of the project or are not correlated to the data. For example, the

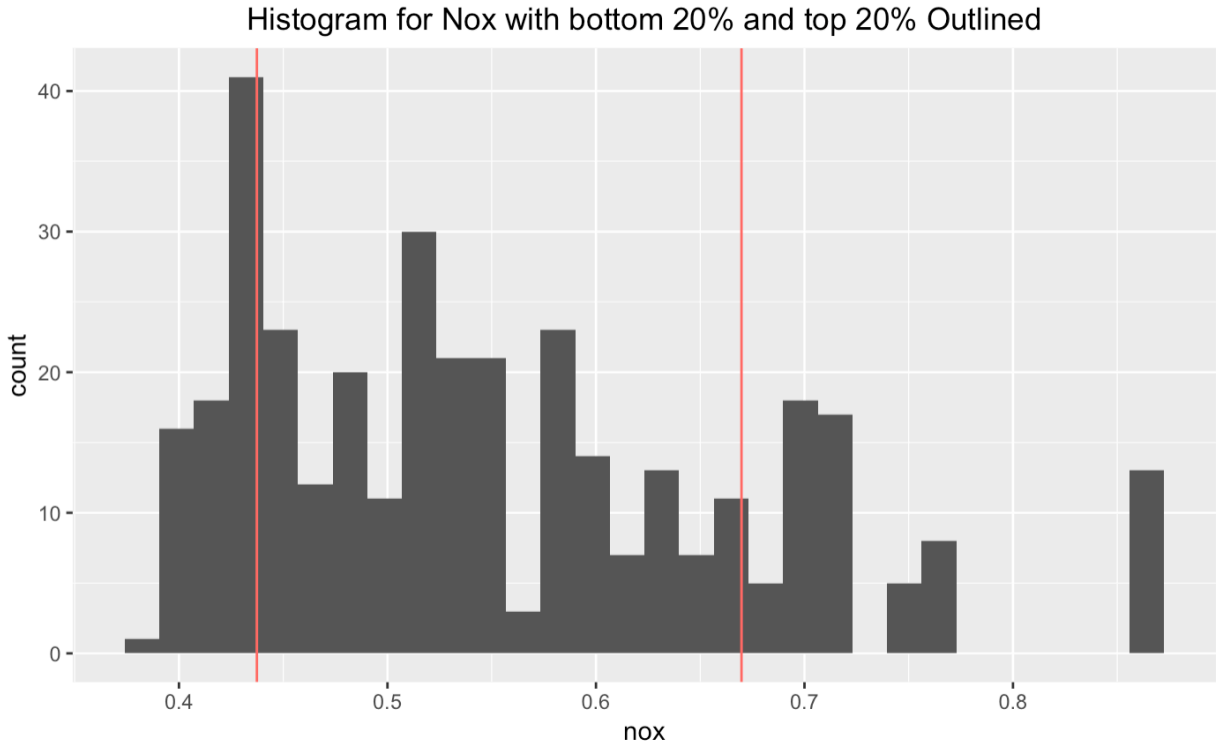
variable chas has over 400 observations with the value 0. Also, based on the correlation matrix the variable is not correlated to the target variable. The dummy variable is defined as 1 if the suburb borders the Charles River and 0 if not. Since the variable is not correlated to crime, it's better off excluding it from the logistic model.

There are also a couple of variables we can put into buckets. For example, the variable rm is the average number of rooms per dwelling. we'd assume that larger dwellings would be associated with higher income communities and therefore less crime.



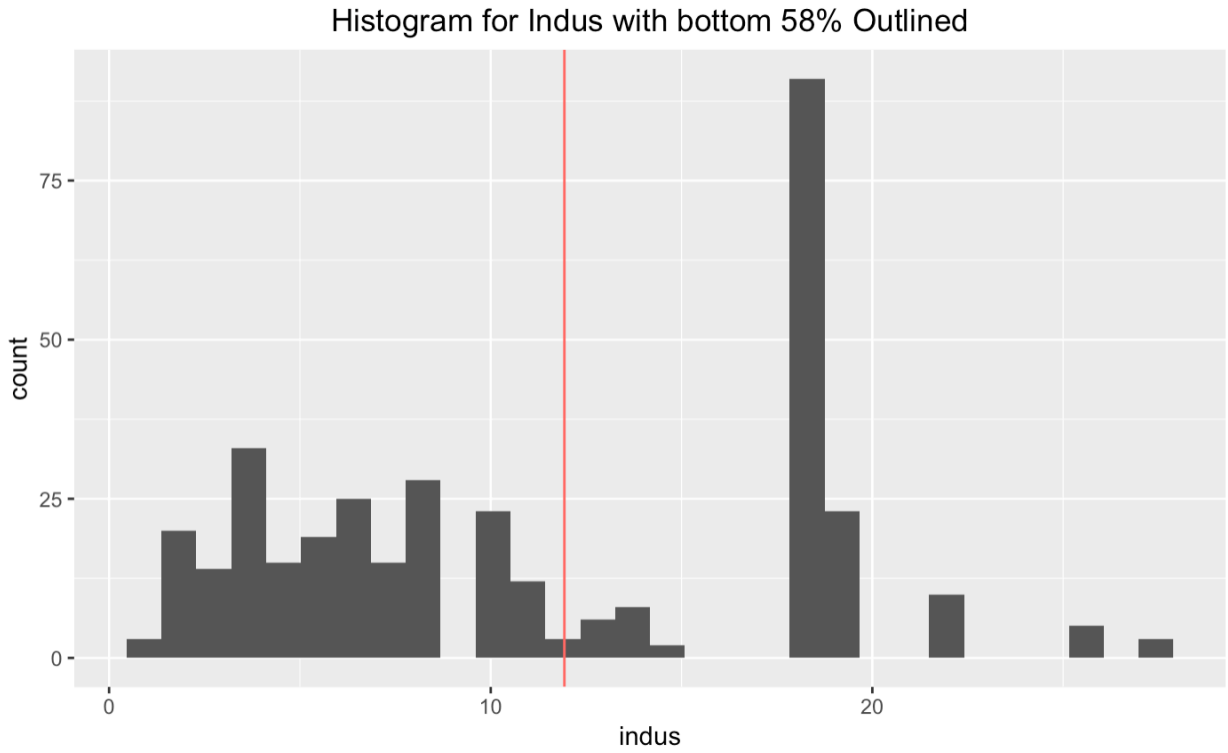
Looking at the histogram above, we see the average rooms for the middle 60% is between `r quantile (train$rm, probs = 0.20)` and `r quantile(train$rm, probs = 0.80)`. We'll go ahead and name the buckets as "low" for those less than or equal to `r quantile(train$rm, probs = 0.20)`, "high" for those greater than or equal to `r quantile(train$rm, probs = 0.80)` and average for those in between.

Another variable that we will transform is nox. The nox variable measures nitrogen oxides concentration (parts per 10 million). The variable itself is positively correlated to the target variable, therefore we assume that areas with high concentrations of nitrogren oxides have higher crime rates.



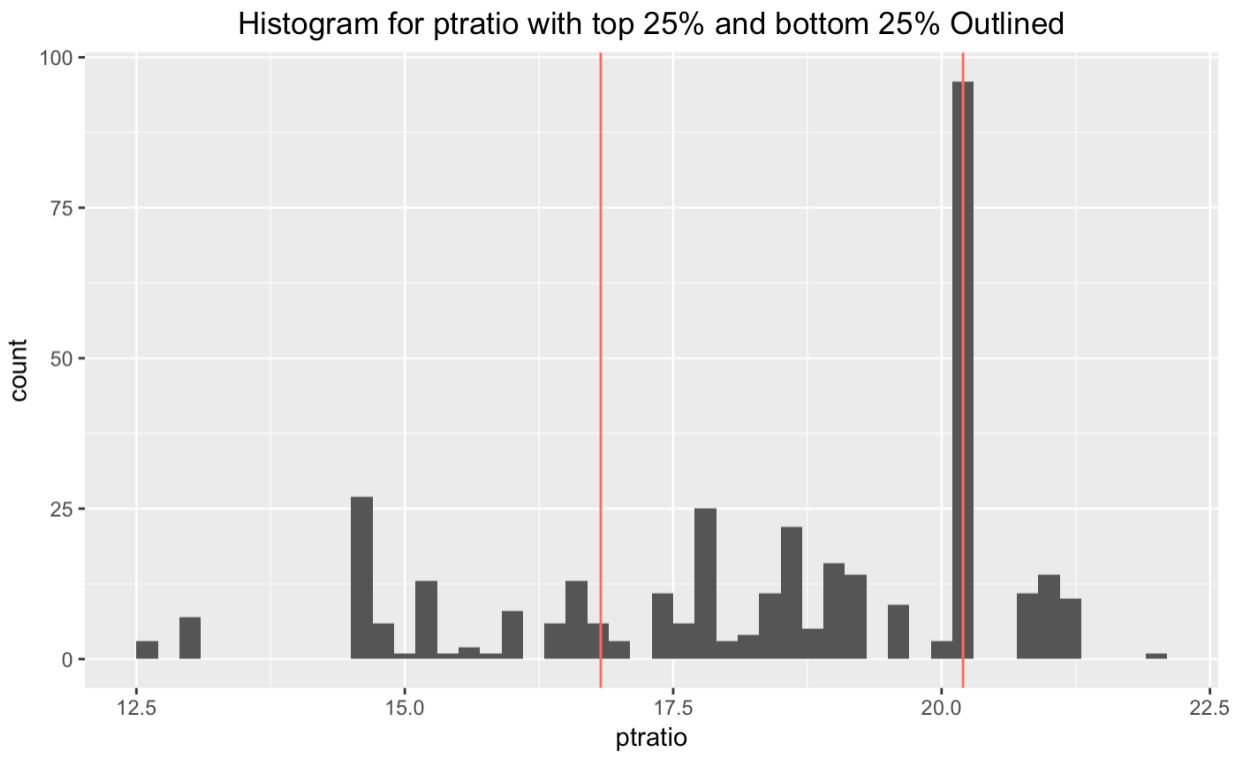
We'll follow the same strategy for the rm variable and use the cutoffs for low nox to be `r quantile(train$nox, probs = 0.20)` and high nox to be `r quantile(train$nox, probs = 0.80)`. Average will be the values between the two.

If we look at the histogram for indus, we see that most of the data is split in two. About half of the observations are under 10% and the other half are above 10%. See the histogram below:



After plotting the histogram, we see that the quantile is split in two by the 58%. For the data set, we'll label those with indus less than or equal to `r quantile(train$indus, probs = 0.58)` as "average"" and those above as "high."

The ptratio measures the pupil-teacher ratio by town. This means that the higher the ratio, the more students to teacher. We see high ratios in under funded schools and districts. We would also assume that crime hates are higher in these areas where school funding is low. See the histogram below:



For this variable ptratios less than or equal to `r quantile(train$ptratio, 0.25)` will be named "low," those greater than or equal to `r quantile(train$ptratio, 0.75)` will be named "high" and anywhere between will be "average."

Build Models

For our model approach we took two strategies involving stepwise elimination. First, we would start with all the original variables (excluding chas) and perform stepwise forward and backwards. We would then do the same for all the original variables and the new bucketed variables. This model would exclude chas and the variables that were used to create the buckets.

For all original variables, we receive the following results:

```
Call:
glm(formula = target ~ zn + indus + nox + rm + age + dis + rad +
    tax + ptratio + lstat + medv, family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7041  -0.2212  -0.0048   0.0056   3.2816

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -36.425739   7.024491  -5.186 2.15e-07 ***
zn           -0.059407   0.039329  -1.510 0.130918
indus        -0.042892   0.050773  -0.845 0.398229
nox          41.465362   8.504695   4.876 1.08e-06 ***
rm           -0.570289   0.779830  -0.731 0.464597
age           0.027713   0.015296   1.812 0.070024 .
dis           0.606509   0.242765   2.498 0.012478 *
rad           0.607785   0.179544   3.385 0.000711 ***
tax          -0.005398   0.003160  -1.709 0.087541 .
ptratio       0.367029   0.137236   2.674 0.007486 **
lstat         0.071314   0.057329   1.244 0.213522
medv          0.208812   0.076363   2.734 0.006248 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 495.18  on 357  degrees of freedom
Residual deviance: 154.40  on 346  degrees of freedom
AIC: 178.4
```

We can see the AIC is at 178.4 and we hope our stepwise elimination will only improve this value. As we perform a forward elimination, our AIC remains at 178.4 and our backward elimination results in an AIC of 175.62.

Next, we perform the same process using the bucketed variables:

```
Call:
glm(formula = target ~ zn + age + dis + rad + tax + lstat + medv +
    rm_group + nox_group + indus_group + ptratio_group, family = "binomial",
    data = train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.94763  -0.41405  -0.04031   0.00181   2.84188

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        -7.872638   2.348322  -3.352 0.000801 ***
zn                 -0.033190   0.024515  -1.354 0.175778
age                 0.031552   0.012621   2.500 0.012420 *
dis                -0.098643   0.189787  -0.520 0.603234
rad                 0.590126   0.166127   3.552 0.000382 ***
tax                -0.002998   0.002958  -1.014 0.310768
lstat               0.029344   0.052290   0.561 0.574672
medv                0.101485   0.049574   2.047 0.040645 *
rm_grouphigh        0.634013   0.825209   0.768 0.442305
rm_grouplow         1.197499   0.569835   2.101 0.035598 *
nox_grouphigh      17.263403 950.108940   0.018 0.985503
nox_grouplow       -0.643138   0.993994  -0.647 0.517617
indus_grouphigh     0.945631   0.589991   1.603 0.108981
ptratio_grouphigh   1.401731   0.491387   2.853 0.004336 **
ptratio_grouplow   -0.483273   0.557946  -0.866 0.386401
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 495.18  on 357  degrees of freedom
Residual deviance: 179.30  on 343  degrees of freedom
AIC: 209.3
```

By using all the variables, our AIC is 209.3 – which is worse than using our original variables only. We continue with forward, resulting in an AIC of 209.3 and backward in 204.77. From these results, we decide to continue only with the models using the original variables and to remove the bucket variables from our models.

Select Models

To decide on selecting between the models we used ANOVA and McFaddens R^2. For ANOVA, we are looking for the widest gap between the null and residual deviance. Here is the ANOVA for the original model with all variables:

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: target

Terms added sequentially (first to last)


        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                    357     495.18
zn       1   92.480       356     402.70 < 2.2e-16 ***
indus    1   74.569       355     328.13 < 2.2e-16 ***
nox      1  105.901       354     222.23 < 2.2e-16 ***
rm       1    6.652       353     215.57  0.009906 **
age      1    0.076       352     215.50  0.783367
dis      1    3.385       351     212.11  0.065803 .
rad      1   41.017       350     171.10 1.509e-10 ***
tax      1    3.976       349     167.12  0.046145 *
ptratio  1    2.728       348     164.39  0.098577 .
lstat    1    1.299       347     163.09  0.254400
medv     1    8.694       346     154.40  0.003193 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

As we continue with the backward model, our gap is 348 - 155.62 and forward is the same as the original. Finally, we calculate the McFadden R^2 for original, backward and forward models, respectively:

```
 McFadden
0.6881949
 McFadden
0.6857294
 McFadden
0.6881949
```

After looking at the AIC, ANOVA, and McFadden R^2, the best results for AIC and R^2 come from the backwards model, so we use this to make our predictions on the test set.

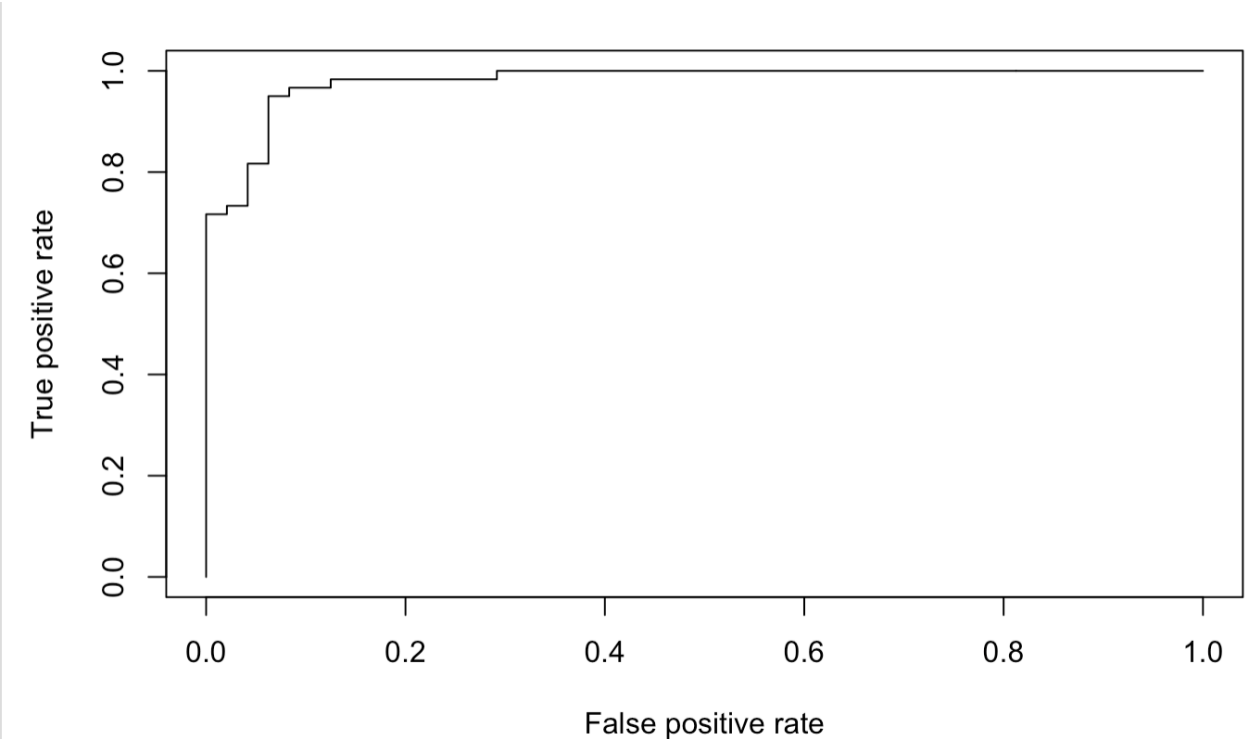
Our predictions on the test set give us an 0.88 accuracy, which is great. If we wanted to further investigate, we could do a k-fold cross validation to be sure this is accurate.

```r
fitted.results <- predict(or_stepb, test, type='response')
fitted.results <- ifelse(fitted.results > 0.5,1,0)

misClasificError <- mean(fitted.results != test$target)
print(paste('Accuracy',1-misClasificError))
```

```
[1] "Accuracy 0.888888888888889"
```

Finally, we build an ROC curve and calculate the AUC, which results in the AUC at 0.9795and the curve as shown:



With the AUC being closer to one, our model has good predictive probability.