CUNY Data 621 HW3 Logiistical Regression

Raphael Nash

Introduction

This assignement we will do a logistical regression to predict crime giving an input dataset. A training and evaluation dataset have been provided.

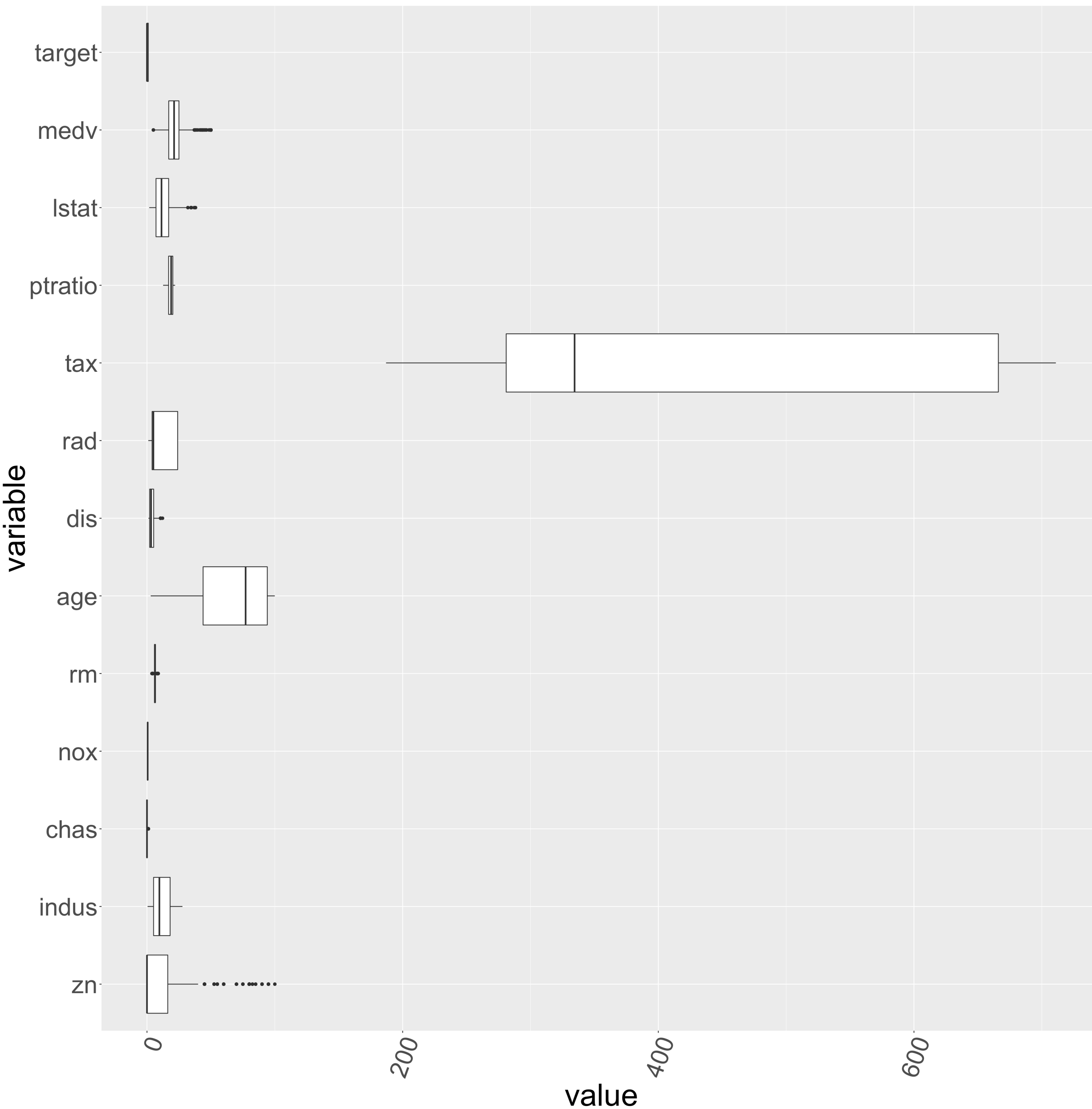
To accomplish this I am going to use the following libraries: 1. ggplot2 2. reshape2 3. coreplot 4. forecast 5. dplyr 6. Deducer

Data Exploration

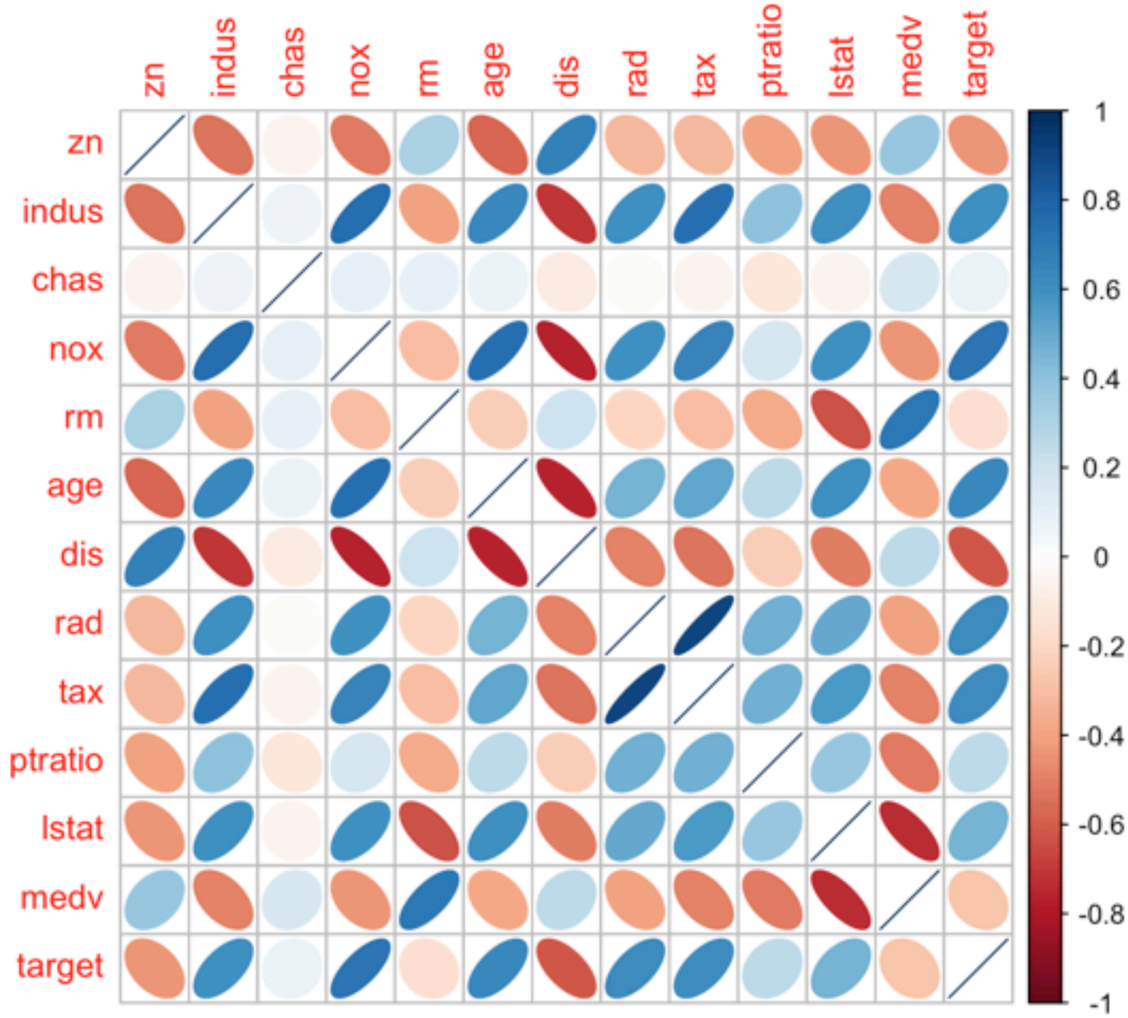
Summary Statistics:

```
##        zn             indus             chas              nox        
##  Min.   :  0.00   Min.   : 0.460   Min.   :0.00000   Min.   :0.3890  
##  1st Qu.:  0.00   1st Qu.: 5.145   1st Qu.:0.00000   1st Qu.:0.4480  
##  Median :  0.00   Median : 9.690   Median :0.00000   Median :0.5380  
##  Mean   : 11.58   Mean   :11.105   Mean   :0.07082   Mean   :0.5543  
##  3rd Qu.: 16.25   3rd Qu.:18.100   3rd Qu.:0.00000   3rd Qu.:0.6240  
##  Max.   :100.00   Max.   :27.740   Max.   :1.00000   Max.   :0.8710  
##        rm             age              dis              rad       
##  Min.   :3.863   Min.   :  2.90   Min.   : 1.130   Min.   : 1.00  
##  1st Qu.:5.887   1st Qu.: 43.88   1st Qu.: 2.101   1st Qu.: 4.00  
##  Median :6.210   Median : 77.15   Median : 3.191   Median : 5.00  
##  Mean   :6.291   Mean   : 68.37   Mean   : 3.796   Mean   : 9.53  
##  3rd Qu.:6.630   3rd Qu.: 94.10   3rd Qu.: 5.215   3rd Qu.:24.00  
##  Max.   :8.780   Max.   :100.00   Max.   :12.127   Max.   :24.00  
##       tax           ptratio         lstat             medv      
##  Min.   :187.0   Min.   :12.6   Min.   : 1.730   Min.   : 5.00  
##  1st Qu.:281.0   1st Qu.:16.9   1st Qu.: 7.043   1st Qu.:17.02  
##  Median :334.5   Median :18.9   Median :11.350   Median :21.20  
##  Mean   :409.5   Mean   :18.4   Mean   :12.631   Mean   :22.59  
##  3rd Qu.:666.0   3rd Qu.:20.2   3rd Qu.:16.930   3rd Qu.:25.00  
##  Max.   :711.0   Max.   :22.0   Max.   :37.970   Max.   :50.00  
##      target      
##  Min.   :0.0000  
##  1st Qu.:0.0000  
##  Median :0.0000  
##  Mean   :0.4914  
##  3rd Qu.:1.0000  
##  Max.   :1.0000  
```

Box Plot of data



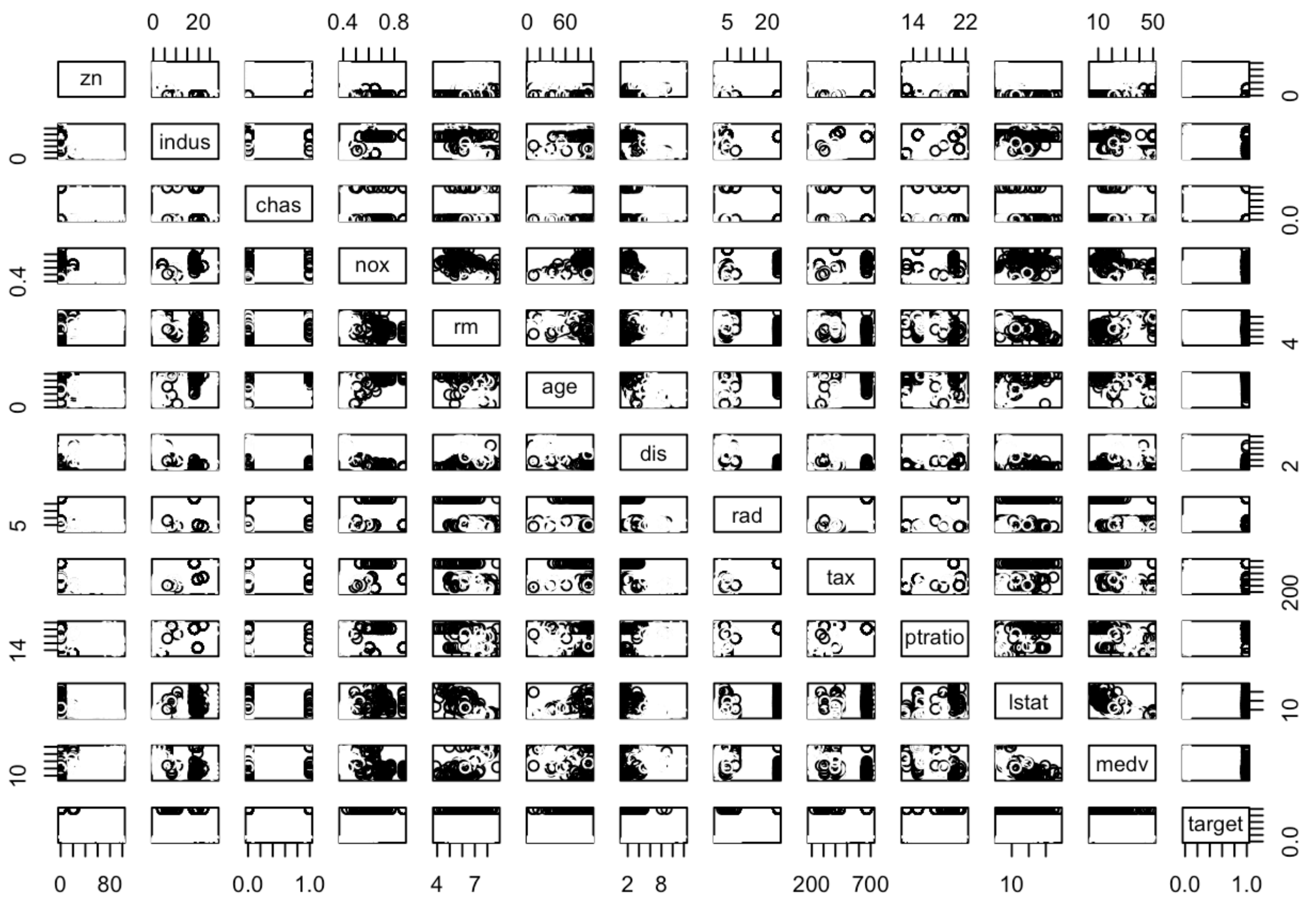
Correlation Plot of Data:



Correlation Matrix:

```
##                  zn       indus        chas         nox          rm
## zn       1.00000000 -0.53826643 -0.04016203 -0.51704518  0.31981410
## indus   -0.53826643  1.00000000  0.06118317  0.75963008 -0.39271181
## chas    -0.04016203  0.06118317  1.00000000  0.09745577  0.09050979
## nox     -0.51704518  0.75963008  0.09745577  1.00000000 -0.29548972
## rm       0.31981410 -0.39271181  0.09050979 -0.29548972  1.00000000
## age     -0.57258054  0.63958182  0.07888366  0.73512782 -0.23281251
## dis      0.66012434 -0.70361886 -0.09657711 -0.76888404  0.19901584
## rad     -0.31548119  0.60062839 -0.01590037  0.59582984 -0.20844570
## tax     -0.31928408  0.73222922 -0.04676476  0.65387804 -0.29693430
## ptratio -0.39103573  0.39468980 -0.12866058  0.17626871 -0.36034706
## lstat   -0.43299252  0.60711023 -0.05142322  0.59624264 -0.63202445
## medv     0.37671713 -0.49617432  0.16156528 -0.43012267  0.70533679
## target  -0.43168176  0.60485074  0.08004187  0.72610622 -0.15255334
##                 age         dis         rad         tax    ptratio
## zn      -0.57258054  0.66012434 -0.31548119 -0.31928408 -0.3910357
## indus    0.63958182 -0.70361886  0.60062839  0.73222922  0.3946898
## chas     0.07888366 -0.09657711 -0.01590037 -0.04676476 -0.1286606
## nox      0.73512782 -0.76888404  0.59582984  0.65387804  0.1762687
## rm      -0.23281251  0.19901584 -0.20844570 -0.29693430 -0.3603471
## age      1.00000000 -0.75089759  0.46031430  0.51212452  0.2554479
## dis     -0.75089759  1.00000000 -0.49499193 -0.53425464 -0.2333394
## rad      0.46031430 -0.49499193  1.00000000  0.90646323  0.4714516
## tax      0.51212452 -0.53425464  0.90646323  1.00000000  0.4744223
## ptratio  0.25544785 -0.23333940  0.47145160  0.47442229  1.0000000
## lstat    0.60562001 -0.50752800  0.50310125  0.56418864  0.3773560
## medv    -0.37815605  0.25669476 -0.39766826 -0.49003287 -0.5159153
## target   0.63010625 -0.61867312  0.62810492  0.61111331  0.2508489
##               lstat       medv      target
## zn      -0.43299252  0.3767171 -0.43168176
## indus    0.60711023 -0.4961743  0.60485074
## chas    -0.05142322  0.1615653  0.08004187
## nox      0.59624264 -0.4301227  0.72610622
## rm      -0.63202445  0.7053368 -0.15255334
## age      0.60562001 -0.3781560  0.63010625
## dis     -0.50752800  0.2566948 -0.61867312
## rad      0.50310125 -0.3976683  0.62810492
## tax      0.56418864 -0.4900329  0.61111331
## ptratio  0.37735605 -0.5159153  0.25084892
## lstat    1.00000000 -0.7358008  0.46912702
## medv    -0.73580078  1.0000000 -0.27055071
## target   0.46912702 -0.2705507  1.00000000
```

Scatterplats for each variable against the target:



Transformations

Since the data does not look normally distributed, I am going to perform a Box-Cox transfomration on each of the input variables. The labdas for each variable are as follows:

lambda <dbl>	variable <fctr>
0.07538486	zn
-0.08779326	indus
0.47220206	chas
-0.99992425	nox
0.28832202	rm
1.99992425	age
-0.61031032	dis
-0.33539473	rad

-0.99992425	tax
1.99992425	ptratio

1-10 of 12 rows Previous **1** 2 Next

Build Models

I will bild a model starting with all the variables are removing the least signifigant variables until the AIC starts increaseing

Summary of model will all varirables (model1):

```
##
## Call:
## glm(formula = target ~ ., family = binomial, data = crime_transformed_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0857  -0.1090  -0.0005   0.1040   3.5539
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.045e+02  1.629e+01   6.413 1.43e-10 ***
## zn          -5.404e-01  5.688e-01  -0.950 0.342083
## indus       -6.930e+00  8.486e+00  -0.817 0.414123
## chas         6.668e-01  7.567e-01   0.881 0.378228
## nox         -7.454e+01  1.166e+01  -6.391 1.65e-10 ***
## rm          -9.119e-01  7.382e-01  -1.235 0.216730
## age          3.586e-04  1.097e-04   3.268 0.001083 **
## dis         -1.443e+01  3.136e+00  -4.601 4.20e-06 ***
## rad         -1.800e+01  4.312e+00  -4.174 2.99e-05 ***
## tax          3.080e+02  4.097e+02   0.752 0.452159
## ptratio      1.327e-02  3.867e-03   3.430 0.000603 ***
## lstat        2.156e-02  5.465e-02   0.395 0.693184
## medv         2.588e-01  7.437e-02   3.480 0.000502 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 187.92  on 453  degrees of freedom
## AIC: 213.92
##
## Number of Fisher Scoring iterations: 8
```

If I drop all non signifigant variables I am left with the following variables:nox, age, dis, pratio, mdev. Therefore I am going to build a model with thoses variables.
Here is the summary for that model (model2)

```
model2 <- glm(target~nox+ age+dis+ rad+ptratio+medv  , data =crime_transformed_df, fa
mily=binomial )
summary(model2)
```

```
##
## Call:
## glm(formula = target ~ nox + age + dis + rad + ptratio + medv,
##     family = binomial, data = crime_transformed_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9517  -0.1563  -0.0018   0.1220   3.3766
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  9.114e+01  1.339e+01   6.806 1.00e-11 ***
## nox         -7.209e+01  1.062e+01  -6.786 1.15e-11 ***
## age          3.151e-04  8.489e-05   3.711 0.000206 ***
## dis         -1.242e+01  2.720e+00  -4.566 4.98e-06 ***
## rad         -1.438e+01  2.840e+00  -5.064 4.11e-07 ***
## ptratio      1.189e-02  2.991e-03   3.977 6.99e-05 ***
## medv         1.609e-01  3.647e-02   4.411 1.03e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 196.11  on 459  degrees of freedom
## AIC: 210.11
##
## Number of Fisher Scoring iterations: 7
```

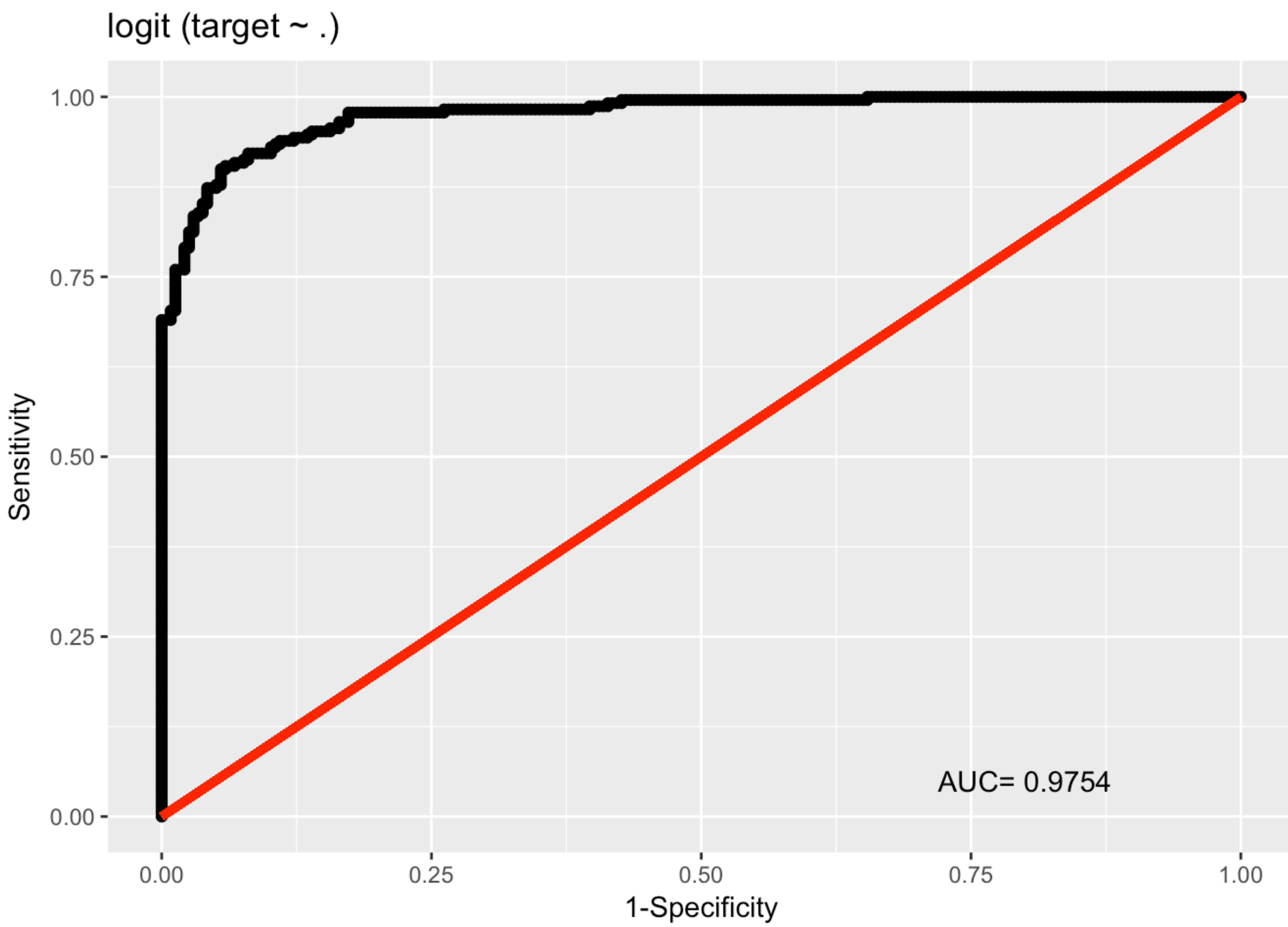
The least signifigant variable of all the variables left is age, so I will drop that variable and create a model with the remaining variables.
Here is the summary of that model:

```
##
## Call:
## glm(formula = target ~ nox + dis + rad + ptratio + medv, family = binomial,
##     data = crime_transformed_df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.28102  -0.20082  -0.00392   0.11758   3.01871
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  93.454841  12.836413   7.280 3.33e-13 ***
## nox         -73.167876  10.047631  -7.282 3.29e-13 ***
## dis          -8.839983   2.407010  -3.673 0.000240 ***
## rad         -13.009411   2.664509  -4.882 1.05e-06 ***
## ptratio       0.009869   0.002781   3.548 0.000388 ***
## medv          0.116046   0.030818   3.765 0.000166 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 211.10  on 460  degrees of freedom
## AIC: 223.1
##
## Number of Fisher Scoring iterations: 7
```

Since AIC started to go up, I am going to stop removing variables.

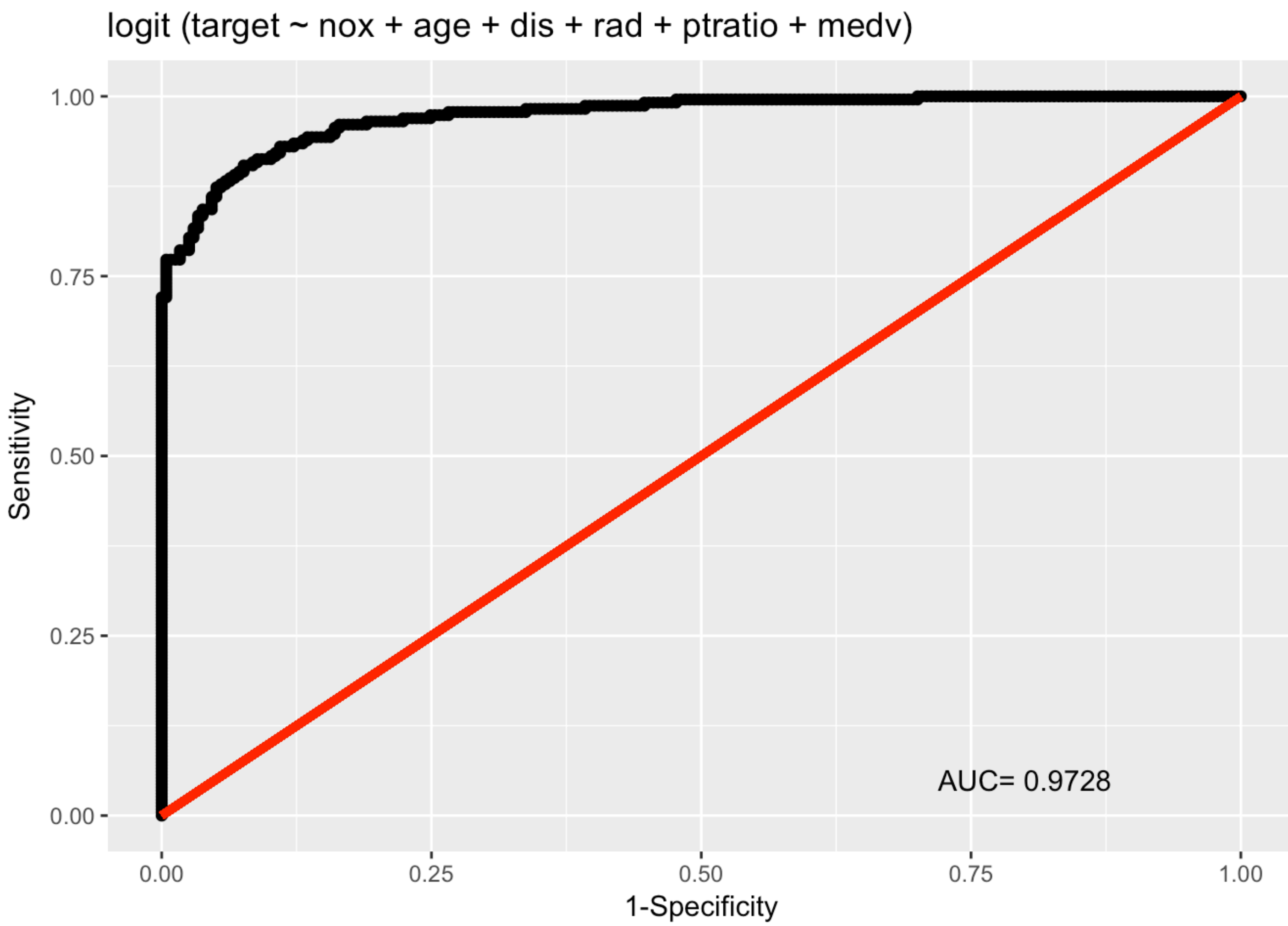
Select Models:

I am going to select the model based on area under the ROC curve (A/K/A AUC) and AIC.



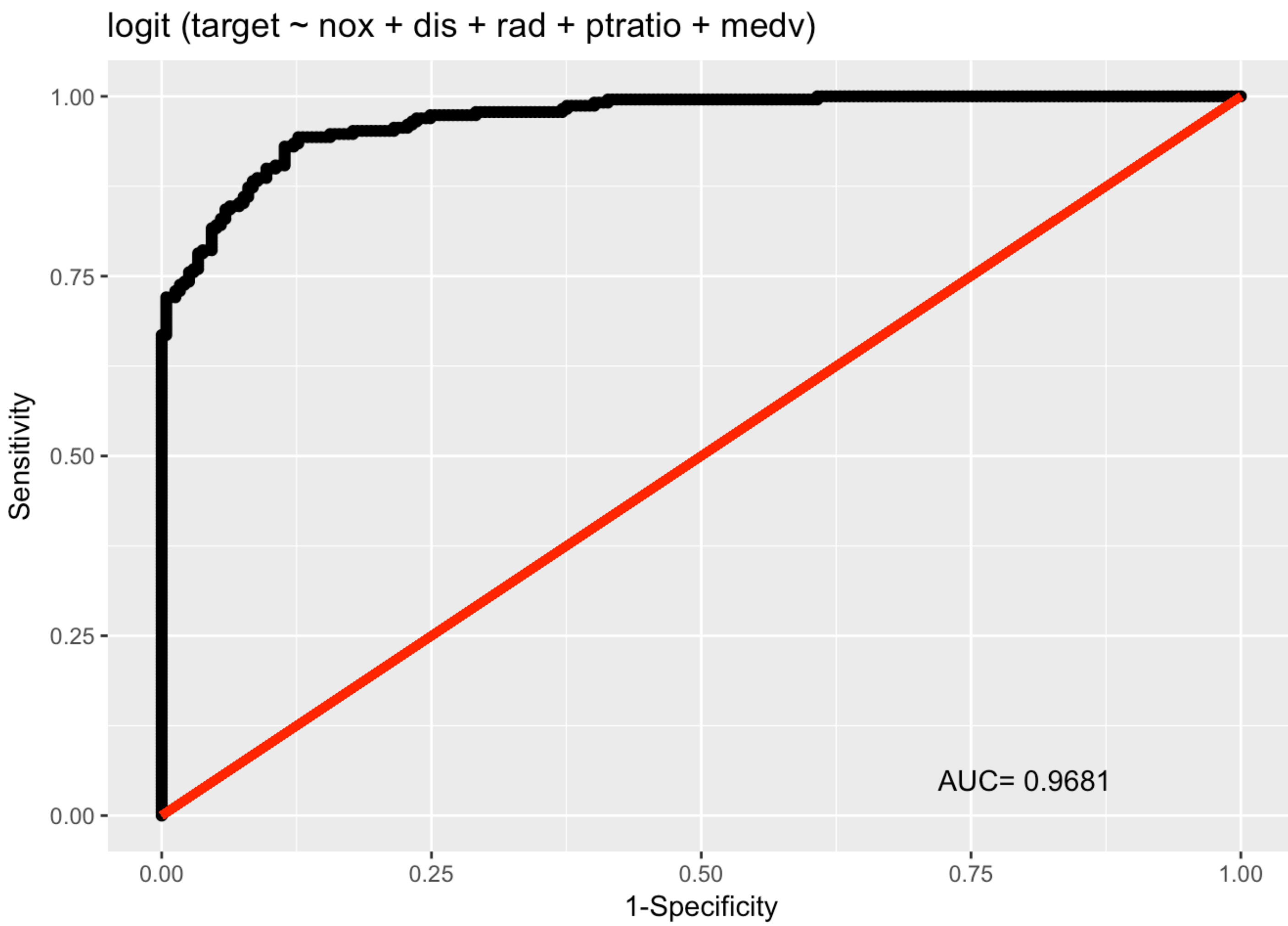
The AIC for model1 is 213.9225599

Model2 Variables in Model 2: nox + age + dis + rad + ptratio + medv



The AIC for model1 is 210.1054573

Model3 Variables: nox + age + dis + rad + ptratio + medv,



The AIC for model3 is 223.1014769

Based the fact that the area under the curve for model 1 and model 2 are virtually identical and the AIC for model 2 is about 1/2 the AIC for model 1 I am going to select model2.

Make Predications

prediction <dbl>	zn <int>	indus <dbl>	chas <int>	nox <dbl>	rm <dbl>	age <dbl>	dis <dbl>	rad <int>	tax <int>
0	0	7.07	0	0.469	7.185	61.1	4.9671	2	242
0	0	8.14	0	0.538	6.096	84.5	4.4619	4	307
0	0	8.14	0	0.538	6.495	94.4	4.4547	4	307
0	0	8.14	0	0.538	5.950	82.0	3.9900	4	307
0	0	5.96	0	0.499	5.850	41.5	3.9342	5	279
0	25	5.13	0	0.453	5.741	66.2	7.2254	8	284
0	25	5.13	0	0.453	5.966	93.4	6.8185	8	284

	0	0	4.49	0	0.449	6.630	56.1	4.4377	3	247
	0	0	4.49	0	0.449	6.121	56.8	3.7476	3	247
	0	0	2.89	0	0.445	6.163	69.6	3.4952	2	276

1-10 of 40 rows | 1-10 of 13 columns Previous **1** 2 3 4 Next

Appendix (R Code)

Setup

library(ggplot2)

library(reshape2)

library(corrplot)

library(forecast)

library(dplyr)

library(Deducer)

crime_df <- read.csv("crime-training-data.csv")

Data Exploration

summary(crime_df)

ggplot(data = melt(crime_df), aes(x=variable, y=value)) + geom_boxplot() + coord_flip() + theme(text = element_text(size=40), axis.text.x = element_text(angle=70, hjust=1))

M <- cor((crime_df)) corrplot(M, method = "ellipse")

pairs(crime_df, col=crime_df$target)

Transformations

calculate_labbdas <- function(df){

df <- df[,1:ncol(df)]

l1 <- numeric(ncol(df))

for (i in 1:ncol(df)){

```
l1[i] <- BoxCox.lambda(df[,i])
```

}

return(data.frame(l1, colnames(df)))

}

box_cox_lambdas <- calculate_labbdas(dplyr::select(crime_df, -target))

colnames(box_cox_lambdas) <- c("lambda", "variable")

box_cox_lambdas

crime_transformed_df <- crime_df

crime_transformed_df$zn < -crime_transformed_df$ zn ^ (filter(box_cox_lambdas, variable=="zn")$lambda)

crime_transformed_df$indus < -crime_transformed_df$ indus ^ (filter(box_cox_lambdas, variable=="indus")$lambda)

crime_transformed_df$chas < -crime_transformed_df$ chas ^ (filter(box_cox_lambdas, variable=="chas")$lambda)

crime_transformed_df$nox < -crime_transformed_df$ nox ^ (filter(box_cox_lambdas, variable=="nox")$lambda)

crime_transformed_df$nox < -crime_transformed_df$ nox ^ (filter(box_cox_lambdas, variable=="rm")$lambda)

crime_transformed_df$age < -crime_transformed_df$ age ^ (filter(box_cox_lambdas, variable=="age")$lambda)

crime_transformed_df$dis < -crime_transformed_df$ dis ^ (filter(box_cox_lambdas, variable=="dis")$lambda)

crime_transformed_df$rad < -crime_transformed_df$ rad ^ (filter(box_cox_lambdas, variable=="rad")$lambda)

crime_transformed_df$tax < -crime_transformed_df$ tax ^ (filter(box_cox_lambdas, variable=="tax")$lambda)

crime_transformed_df$ptratio < -crime_transformed_df$ ptratio ^ (filter(box_cox_lambdas, variable=="ptratio")$lambda)

Build Models

model1 <- glm(target~., data =crime_transformed_df, family=binomial)

summary(model1)

model2 <- glm(target~nox+ age+dis+ rad+ptratio+medv , data =crime_transformed_df, family=binomial)

summary(model2)

model3 <- glm(target~nox+dis+ rad+ptratio+medv , , data =crime_transformed_df, family=binomial)

summary(model3)

Select Models:

rocplot(model1)

rocplot(model2)

rocplot(model3)

Make Predications

crime_eval_df <- read.csv(“crime-evaluation-data.csv”)

crime_eval_transformed_df <- crime_eval_df

crime_eval_transformed_df$zn < -crime_e val_t ransformed_d f$zn ^ (filter(box_cox_lambdas, variable==“zn”)$lambda)

crime_eval_transformed_df$indus < -crime_e val_t ransformed_d f$indus ^ (filter(box_cox_lambdas, variable==“indus”)$lambda)

crime_eval_transformed_df$chas < -crime_e val_t ransformed_d f$chas ^ (filter(box_cox_lambdas, variable==“chas”)$lambda)

crime_eval_transformed_df$nox < -crime_e val_t ransformed_d f$nox ^ (filter(box_cox_lambdas, variable==“nox”)$lambda)

crime_eval_transformed_df$nox < -crime_e val_t ransformed_d f$nox ^ (filter(box_cox_lambdas, variable==“rm”)$lambda)

crime_eval_transformed_df$age < -crime_e val_t ransformed_d f$age ^ (filter(box_cox_lambdas, variable==“age”)$lambda)

crime_eval_transformed_df$dis < -crime_e val_t ransformed_d f$dis ^ (filter(box_cox_lambdas, variable==“dis”)$lambda)

crime_eval_transformed_df$rad < -crime_e val_t ransformed_d f$rad ^ (filter(box_cox_lambdas, variable==“rad”)$lambda)

crime_eval_transformed_df$tax < -crime_e val_t ransformed_d f$tax ^ (filter(box_cox_lambdas, variable==“tax”)$lambda)

crime_eval_transformed_df$ptratio < -crime_e val_t ransformed_d f$ptratio ^ (filter(box_cox_lambdas, variable==“ptratio”)$lambda)

probs <- predict(model2,crime_eval_df)

prediction <- ifelse (probs > .5 ,1,0)

cbind(prediction, crime_eval_df)