DATA 621 - Business Analytics and Data Mining - Homework 3

Daniel Hong, Mauricio Alarcon, Maxwell Wagner

October 30, 2016

INTRODUCTION

Crime is a major concern especially in urban settings and can affect the development of these areas as well as adjacent ones. In recent years increased development of urban and surrounding areas has led to a revitalization and in some cases a significant reduction in crime. We will attempt to develop a model to predict whether a specific neighborhood will be at risk for high crime levels. This important analysis can be utilized in a number of ways ranging from where to target anti-crime initiatives to where developers should target projects.

DATA EXPLORATION

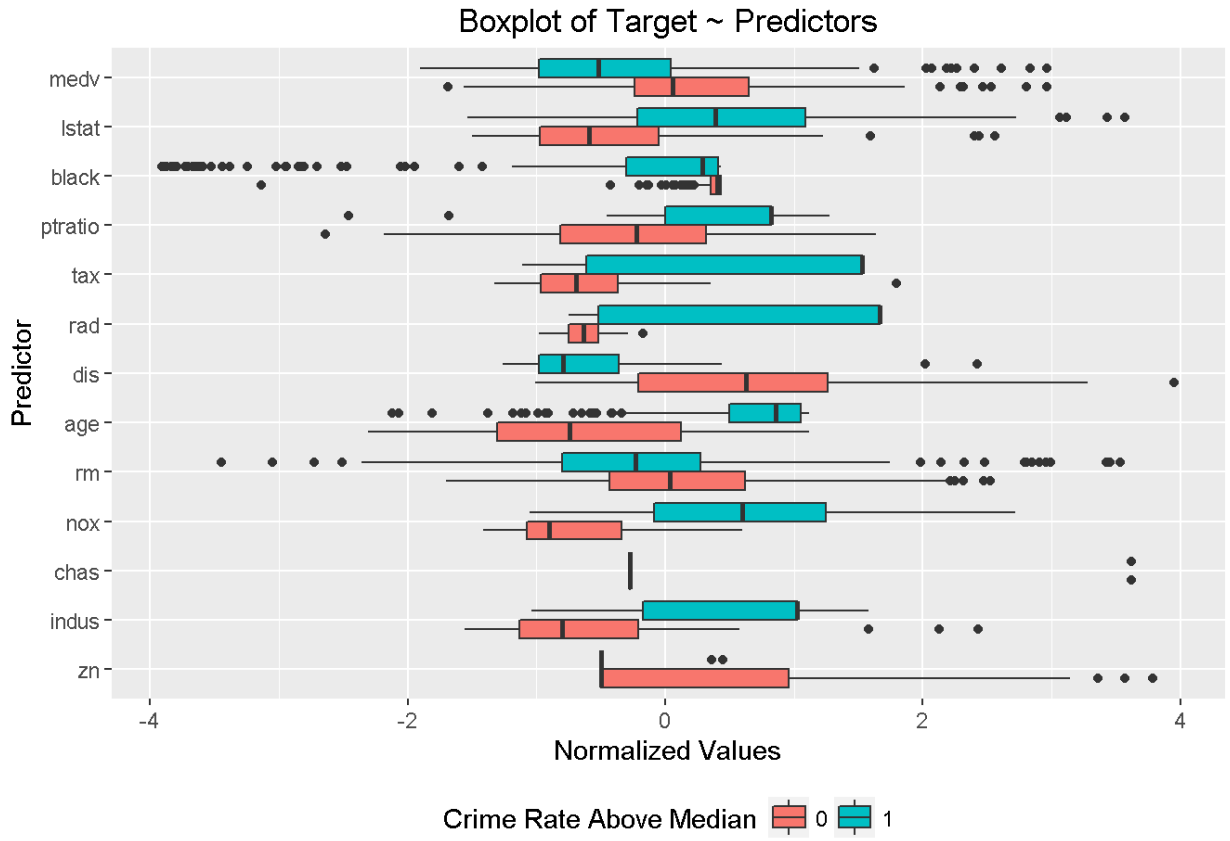
The binary response variable that indicates if the crime is above the median (target), and 13 predictor variables which are various demographic, economic and environmental indicators are given with varying degrees of validity. It's worth mentioning that we have no missing data. The target variable appears to be evenly distributed, with 230 positive cases (1, above the mean) and 237 negative cases (0, below the mean). With the mean and sd we calculated the coefficient of variation, so that we can have an intuitive comprehension of variation across variables. Since CV cannot be calculated with a 0 mean and we need to be careful with variables with positive and negative values that get us a mean close to zero as the results could be misleading.

Variable	n	mean	sd	median	min	max	completeness	cv(%)
target	466	0.491416	0.500464	0	0	1	100%	101.8411
zn	466	11.57725	23.36465	0	0	100	100%	201.8152
indus	466	11.10502	6.845855	9.69	0.46	27.74	100%	61.64648
chas	466	0.070816	0.256792	0	0	1	100%	362.6214
nox	466	0.554311	0.116667	0.538	0.389	0.871	100%	21.04717
rm	466	6.290674	0.704851	6.21	3.863	8.78	100%	11.2047
age	466	68.3676	28.32138	77.15	2.9	100	100%	41.42515
dis	466	3.795693	2.10695	3.19095	1.1296	12.1265	100%	55.50896
rad	466	9.530043	8.685927	5	1	24	100%	91.14258
tax	466	409.5021	167.9001	334.5	187	711	100%	41.00103

ptratio	466	18.3985	2.196845	18.9	12.6	22	100%	11.94035
black	466	357.1202	91.32113	391.34	0.32	396.9	100%	25.57154
lstat	466	12.63146	7.101891	11.35	1.73	37.97	100%	56.22383
medv	466	22.58927	9.239681	21.2	5	50	100%	40.90297

Potential Correlations

The boxplot shows every predictor against the two possible states of the response variable (positive=1, negative=0). There appears to be some outliers that we need to investigate further, but it looks like a number of variables explained below (rox, age, rad, lstat, indus, medv) are potentially more meaningful predictors. Other variables such as the number of rooms per dwelling (rm) and the dummy variable for whether the suburb borders the Charles River (chas), don't appear to be meaningful at all.



Multicollinearity is an area of concern that we should address, but there is a certain degree of expectation that these variables will correlate with each other especially since we are looking at attributes of more desirable areas than others. We can further detail the clear differences between the values observed in high crime (less desirable) areas:

Variable	Definition	CR Above Median - Avg	CR Above Median - n	CR Below Median - Avg	CR Below Median - n	delta
nox	nitrogen oxides concentration (parts per 10 million)	0.7378874	229	-0.7129798	237	1.4508672
age	proportion of owner-occupied units built prior to 1940	0.6403298	229	-0.6187153	237	1.2590452
rad	index of accessibility to radial highways	0.638296	229	-0.6167502	237	1.2550462
dis	weighted mean of distances to five Boston employment centers	-0.6287112	229	0.6074889	237	1.2362001
tax	full-value property-tax rate per $10,000	0.6210287	229	-0.6000657	237	1.2210945
indus	proportion of non-retail business acres per suburb	0.6146645	229	-0.5939164	237	1.2085809
lstat	lower status of the population (percent)	0.4767387	229	-0.4606462	237	0.9373849
zn	proportion of residential land zoned for large lots (over 25000 square feet)	-0.4386859	229	0.4238779	237	0.8625638
black	1000(Bk - 0.63)2 where Bk is the proportion of	-0.3586836	229	0.3465761	237	0.7052597

	blacks by town					
medv	median value of owner-occupied homes in $1000s	-0.2749404	229	0.2656598	237	0.5406002
ptratio	pupil-teacher ratio by town	0.254919	229	-0.2463141	237	0.5012331
rm	average number of rooms per dwelling	-0.1550285	229	0.1497955	237	0.3048241
chas	a dummy var. for whether the suburb borders the Charles River (1) or not (0)	0.0813406	229	-0.0785949	237	0.1599355

From the above table we can see some striking differences. For instance, areas with lower crime rates have (the most obvious):

- A lower nitrogen oxide concentration (less pollution)
- Lower proportion of owner-occupied units built prior to 1940 (less gentrified)
- Lower index of accessibility to radial highways
- Lower percentage of lower status of the population
- Greater proportion of residential land zoned for large lots (over 25000 square feet)
- Higher median value of owner-occupied homes in $1000s

DATA TRANSFORMATIONS

A preliminary logistic regression was generated. Based on the results and in light of the other artifacts presented above, we consider, the following transformations appropriate:

- Replace all outliers with the values at quantiles 0.05 and 0.95

MODEL BUILDING

Methodology

Starting with the imputed variables, we calculated an initial model. From this model, we eliminated those predictors that had a high P-value (>0.05). The selected model is the result of re-iterating over the training data until we ended up with all the statistically significant predictors.

In addition, we used 75% of the data for training, evenly distributed across the positive and negative cases. The remaining 25% of the data was used for evaluation (ROC & confusion matrix).

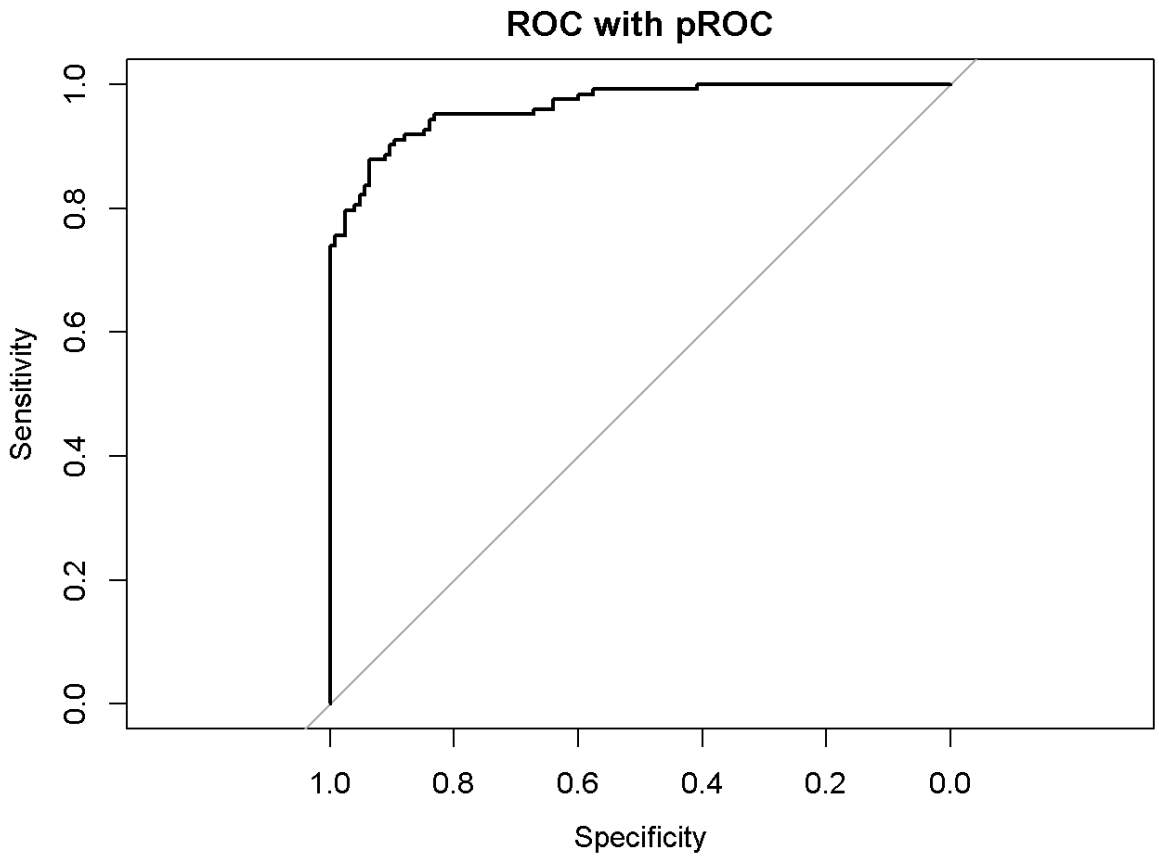
Regression Results

The results obtained for this regression are as follows:

```
##
## Call:
## glm(formula = I(target) ~ ., family = binomial, data = train[,
##     c(predictors, target)])
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.81652  -0.16500  -0.00849   0.00404   2.68018
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -48.085247   9.414239  -5.108 3.26e-07 ***
## nox          56.208776  10.534838   5.336 9.53e-08 ***
## age           0.034477   0.014868   2.319 0.020407 *
## dis           0.489997   0.286398   1.711 0.087100 .
## rad           0.745392   0.192047   3.881 0.000104 ***
## tax          -0.007141   0.003124  -2.286 0.022266 *
## ptratio       0.422802   0.141278   2.993 0.002765 **
## medv          0.174909   0.057124   3.062 0.002199 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 481.50  on 347  degrees of freedom
## Residual deviance: 133.95  on 340  degrees of freedom
## AIC: 149.95
##
## Number of Fisher Scoring iterations: 9
```

target = -48.09 + 56.21 nox + 0.03 age + 0.49 dis + 0.75 rad - 0.01 tax + 0.42 ptratio + 0.17 medv. Tax had a slightly negative effect on the target but the other variables (nox, age, dis, rad, ptratio, mev) had positive effects. Surprisingly in this model, the weighted mean of distances to five Boston employment centers (dis) and the median value of owner-occupied homes (medv) had a positive effect on crime despite the negative relationship observed earlier.

Model's Predictive Power




```
##  Confusion  Matrix
##       Predicted
##          0   1
##  O   0 117   8
##  b   1  18 105
##  s
```

ROC	0.9596
Accuracy:	0.8669355
Classification Error Rate:	0.1330645
Precision:	0.9090909
Sensitivity:	0.8130081
Specificity:	0.92

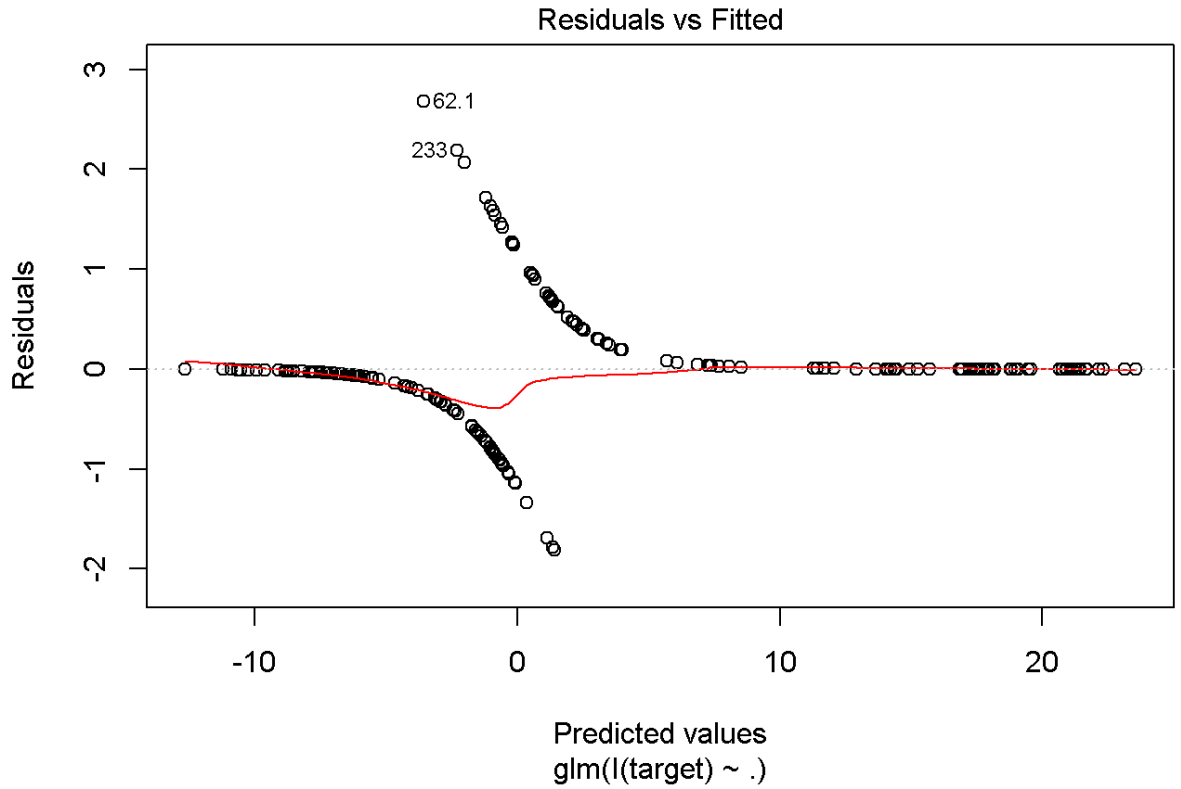
F1 Score:	0.8583691

The above presented metrics were calculated based on an evaluation partition (25% off the initial dataset). This dataset contained 123 positive and 125 negative observed cases. The model predicted a total of 113 positive cases, out of which 105 were true positive. The model failed to correctly classify as positive 18 cases. The total classification error rate is a low 0.13.

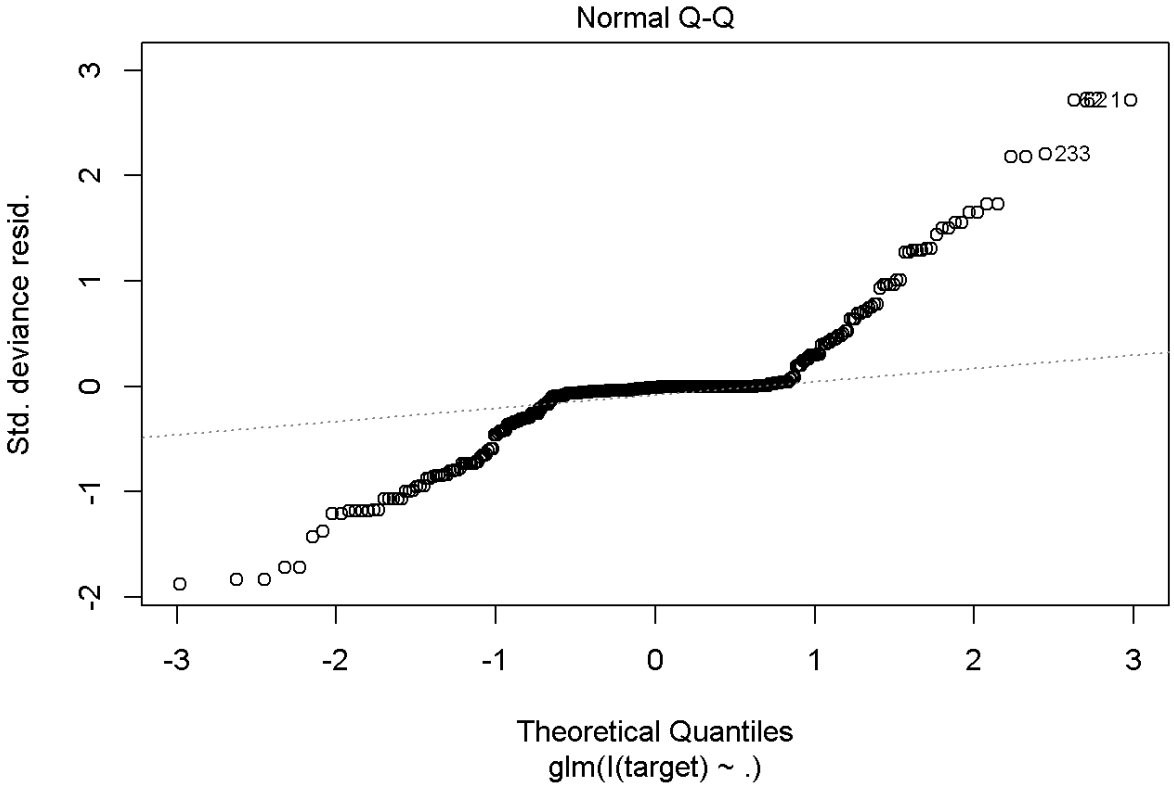
In overall, the classification performance metrics are on the high-performing side.

Diagnostic Charts

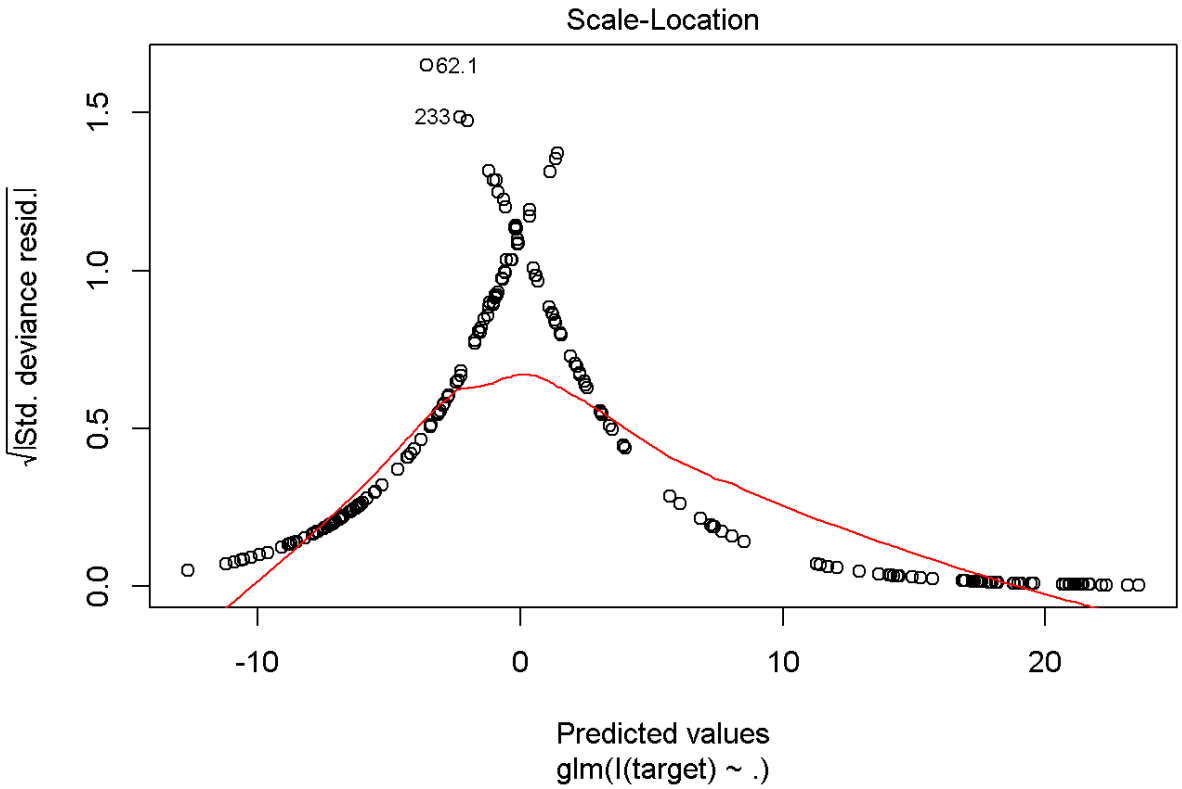
A caveat about the following diagnostic charts is that the plotting method, `plot.lm()`, is meant for linear regression models. This has the effect of giving the residuals vs fitted and scale-location plots some odd trends.



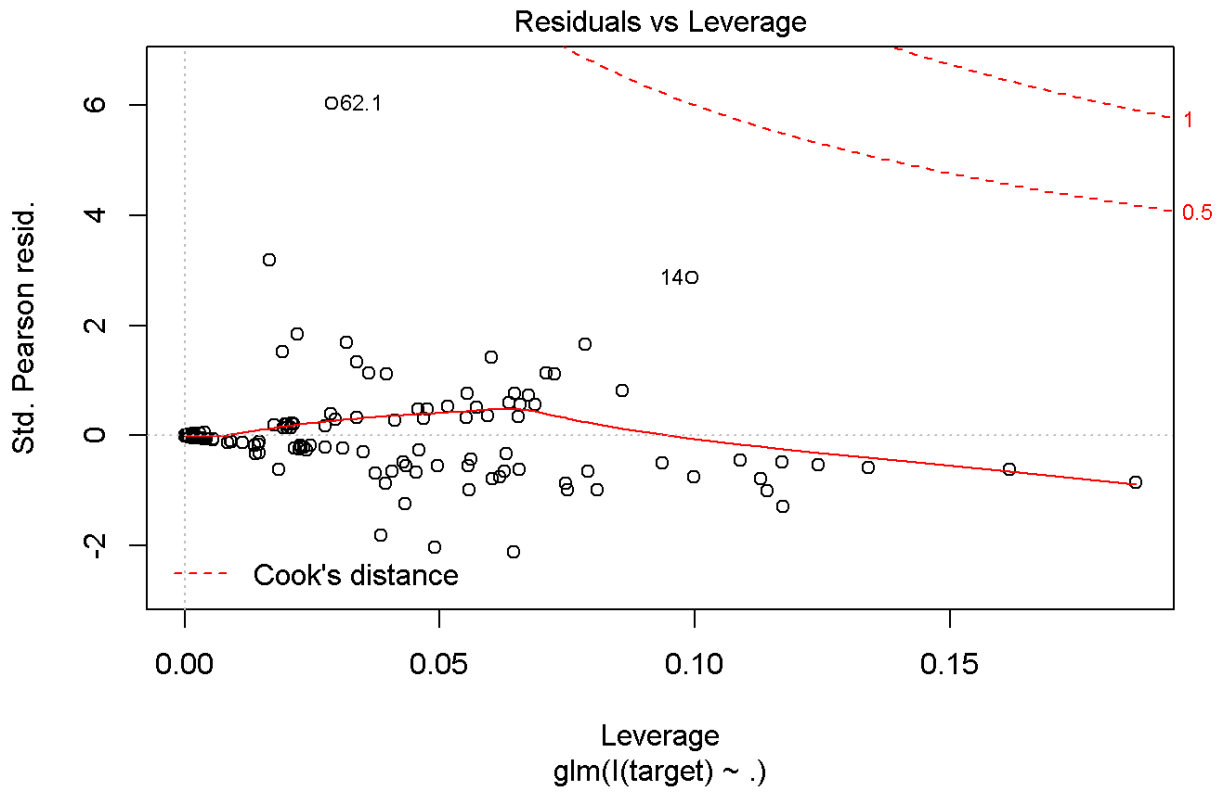
As expected with this method of model, two curvilinear lines form. This plot typically diagnoses nonlinearity, but with a binomial GLM, nonlinearity is expected.



The normal Q-Q plot depicts a middle section that follows the normal distribution, but both tails of the graph are heavy which could indicate a departure from normal distribution. To reiterate, being a binomial GLM the residuals are not required to be normally distributed in order for the model to be accurate.



A scale-location plot typically diagnoses homoscedasticity/heteroscedasticity and would be a random scattering were the model linear. A binomial GLM can fulfill either definition and remain valid.



The residuals vs leverage plot does not show any points beyond Cook's distance lines. The influence of leverage points on the regression is minimal to zero.

PREDICTIONS

Below are the predictions for the provided dataset using the model. We performed imputations as we did in the original dataset.

zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv	predict
0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	FALSE
0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	TRUE

0	8.14	0	0.538	6.495	94.4	4.4547	4	307	21	387.94	12.8	18.4	TRUE
0	8.14	0	0.538	5.95	82	3.99	4	307	21	232.6	27.71	13.2	FALSE
0	5.96	0	0.499	5.85	41.5	3.9342	5	279	19.2	396.9	8.77	21	FALSE
25	5.13	0	0.453	5.741	66.2	7.2254	8	284	19.7	395.11	13.15	18.7	FALSE
25	5.13	0	0.453	5.966	93.4	6.8185	8	284	19.7	378.08	14.44	16	FALSE
0	4.49	0	0.449	6.63	56.1	4.4377	3	247	18.5	392.3	6.53	26.6	FALSE
0	4.49	0	0.449	6.121	56.8	3.7476	3	247	18.5	395.15	8.44	22.2	FALSE
0	2.89	0	0.445	6.163	69.6	3.4952	2	276	18	391.83	11.34	21.4	FALSE
0	25.65	0	0.581	5.856	97	1.9444	2	188	19.1	370.31	25.41	17.3	TRUE
0	25.65	0	0.581	5.613	95.6	1.7572	2	188	19.1	359.29	27.26	15.7	FALSE
0	21.89	0	0.624	5.637	94.7	1.9799	4	437	21.2	396.9	18.34	14.3	TRUE
0	19.58	0	0.605	6.101	93	2.2834	5	403	14.7	240.16	9.81	25	TRUE
0	19.58	0	0.605	5.88	97.3	2.3887	5	403	14.7	348.13	12.03	19.1	TRUE
0	10.59	1	0.489	5.96	92.1	3.8771	4	277	18.6	393.25	17.27	21.7	FALSE
0	6.2	0	0.504	6.552	21.4	3.3751	8	307	17.4	380.34	3.76	31.5	FALSE
0	6.2	0	0.507	8.247	70.4	3.6519	8	307	17.4	378.95	3.95	48.3	TRUE
22	5.86	0	0.431	6.957	6.8	8.9067	7	330	19.1	386.09	3.53	29.6	FALSE
90	2.97	0	0.4	7.088	20.8	7.3073	1	285	15.3	394.72	7.85	32.2	FALSE

80	1.76	0	0.385	6.23	31.5	9.0892	1	241	18.2	341.6	12.93	20.1	FALSE
33	2.18	0	0.472	6.616	58.1	3.37	7	222	18.4	393.36	8.93	28.4	FALSE
0	9.9	0	0.544	6.122	52.8	2.6403	4	304	18.4	396.9	5.98	22.1	FALSE
0	7.38	0	0.493	6.415	40.1	4.7211	5	287	19.6	396.9	6.12	25	FALSE
0	7.38	0	0.493	6.312	28.9	5.4159	5	287	19.6	396.9	6.15	23	FALSE
0	5.19	0	0.515	5.895	59.6	5.615	5	224	20.2	394.81	10.56	18.5	FALSE
80	2.01	0	0.435	6.635	29.7	8.344	4	280	17	390.94	5.99	24.5	FALSE
0	18.1	0	0.718	3.561	87.9	1.6132	24	666	20.2	354.7	7.12	27.5	TRUE
0	18.1	1	0.631	7.016	97.5	1.2024	24	666	20.2	392.05	2.96	50	TRUE
0	18.1	0	0.584	6.348	86.1	2.0527	24	666	20.2	83.45	17.64	14.5	TRUE
0	18.1	0	0.74	5.935	87.9	1.8206	24	666	20.2	68.95	34.02	8.4	TRUE
0	18.1	0	0.74	5.627	93.9	1.8172	24	666	20.2	396.9	22.88	12.8	TRUE
0	18.1	0	0.74	5.818	92.4	1.8662	24	666	20.2	391.45	22.11	10.5	TRUE
0	18.1	0	0.74	6.219	100	2.0048	24	666	20.2	395.69	16.59	18.4	TRUE
0	18.1	0	0.74	5.854	96.6	1.8956	24	666	20.2	240.52	23.79	10.8	TRUE
0	18.1	0	0.713	6.525	86.5	2.4358	24	666	20.2	50.92	18.13	14.1	TRUE
0	18.1	0	0.713	6.376	88.4	2.5671	24	666	20.2	391.43	14.65	17.7	TRUE
0	18.1	0	0.655	6.209	65.4	2.9634	24	666	20.2	396.9	13.22	21.4	TRUE

0	9.69	0	0.585	5.794	70.6	2.8927	6	391	19.2	396.9	14.1	18.3	TRUE
0	11.93	0	0.573	6.976	91	2.1675	1	273	21	396.9	5.64	23.9	TRUE

Appendix - Code

```
---
title: "DATA621 - Crime"
author: "Daniel Hong, Mauricio Alarcon, Maxwell Wagner"
date: "October 10, 2016"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```


* * *

1. DATA EXPLORATION (25 Points)
Describe the size and the variables in the moneyball training data set. Consider that too much detail will cause a manager to lose interest while too little detail will make the manager consider that you aren't doing your job. Some suggestions are given below. Please do NOT treat this as a check list of things to do to complete the assignment. You should have your own thoughts on what to tell the boss. These are just ideas.

a. Mean / Standard Deviation / Median

```{r}
require("plyr")
require("knitr")
require("psych")
# Let's load the data

training <-
read.csv(url('https://raw.githubusercontent.com/rmalarc/DATA621/master/hw03/crime-train
ing-data.csv'))
metadata <-
read.csv(url('https://raw.githubusercontent.com/rmalarc/DATA621/master/hw03/crime-metad
ata.csv'))
evaluation <-
read.csv(url('https://raw.githubusercontent.com/rmalarc/DATA621/master/hw03/crime-evalu
ation-data.csv'))

kable(metadata)
columns <- colnames(training)
target <- "target"
inputs <- columns[!columns %in% c(target,"INDEX")]
```
```

```
summary <-
describe(training[,c(target,inputs)])[,c("n","mean","sd","median","min","max")]
summary$completeness  <- summary$n/nrow(training)
summary$cv  <- 100*summary$sd/summary$mean

kable(summary)

```

b. Bar Chart or Box Plot of the data

How are the input values distributed?, do we need to do something about them?

Here's the distribution of the values for each of the variables

```{r}
require("reshape2")
require("ggplot2")
# Let's  melt the DF so that we can plot it more easily

ggplot(melt(training,  measure.vars  = inputs)
       ,aes(x=variable,y=value)
       )+
    geom_boxplot(aes(fill  = factor(target)))  +
    coord_flip()
```

Some of these probably need to be rescaled: TEAM_PITCHING_H, TEAM_PITCHING_SO   (what is this????)

Let's get a view of the normalized values:

```{r}
require("reshape2")
require("ggplot2")
# Let's  melt the DF so that we can plot it more easily

ggplot(melt(data.frame(scale(training[,inputs]),target=training[,target]),     measure.vars = inputs),
       aes(x=variable,y=value)
       )+
    geom_boxplot(aes(fill  = factor(target)))+
  guides(fill=guide_legend(title="Crime   Rate  Above  Median"))  +
   theme(legend.position="bottom")+
    coord_flip()+
  labs(title="Boxplot  of Target  ~ Predictors",  y="Normalized  Values",  x="Predictor")

```

possible correlations

```{r}

training_normalized  <- data.frame(scale(training[,inputs]),target=training[,target])

summary_positive  <-
describe(training_normalized[training_normalized$target==1,c(target,inputs)])[,c("mean"
,"n")]
summary_negative  <-
```

```
describe(training_normalized[training_normalized$target==0,c(target,inputs)])[,c("mean"
,"n")]
summary_by_target  <- merge(summary_positive,summary_negative,by="row.names")
colnames(summary_by_target)   <- c("Variable","CR  Above Median  - Avg","CR  Above Median  -
n","CR  Below Median  - Avg",  "CR Below Median  - n")
summary_by_target$delta   <- abs(summary_by_target[,"CR   Above Median  -
Avg"]-summary_by_target[,"CR   Below Median  - Avg"])

kable(merge(metadata,summary_by_target)[order(-summary_by_target$delta),])
```


2. DATA PREPARATION (25 Points)
Describe how you have transformed the data by changing the original variables or
creating new variables. If you did transform the data or create new variables, discuss
why you did this. Here are some possible transformations.

a. Fix missing values (maybe with a Mean or Median value)
b. Create flags to suggest if a variable was missing
c. Transform data by putting it into buckets
d. Mathematical transforms such as log or square root (or use Box-Cox)
e. Combine variables (such as ratios or adding or multiplying) to create new variables

regression before transformations

Cap all values to their 5 and 95 percentiles

```{r}

transformed  <- training

cap <- function(x){
    quantiles  <- c( qnorm(0.05,mean(x),sd(x)),   qnorm(0.95,mean(x),sd(x))   )
    x[ x < quantiles[1]  ] <- max(0,quantiles[1])
    x[ x > quantiles[2]  ] <- quantiles[2]
    x
}

#transformed$black  <- log(transformed$black)
transformed<-data.frame(apply(   transformed[,inputs],2,   cap),target=transformed$target)


```


split the dataset into training and testing

```{r}


## 75% of the sample  size
smp_size  <- floor(0.75  * nrow(transformed))

## set the seed to make your partition  reproductible
set.seed(123)
train_ind  <- c(sample(seq_len(nrow(transformed[transformed$target==1,])),     size =
```

```
smp_size/2),sample(seq_len(nrow(transformed[transformed$target==0,])),     size =
smp_size/2))

train <- transformed[train_ind,   ]
test <- transformed[-train_ind,   ]
```



```{r}

model <- glm(I(target)~.,data=train,family   = binomial)

summary(model)

predicted  <- predict(model,test,type='response')
require("pROC")
d_roc <- roc(ifelse(test$target>0.5,1,0),predicted)
plot(d_roc,  main = "ROC with pROC")
#ci(d_roc)

require("caret")
table(ifelse(test$target>0.5,1,0),ifelse(predicted>0.5,1,0))

```


regression after transformations

3. BUILD MODELS (25 Points)

Using the training data, build at least three different binary logistic regression models, using different variables (or the same variables with different transformations). You may select the variables manually, use an approach such as Forward or Stepwise, use a different approach, or use a combination of techniques. Describe the techniques you used. If you manually selected a variable for inclusion into the model or exclusion into the model, indicate why this was done.

Be sure to explain how you can make inferences from the model, as well as discuss other relevant model output. Discuss the coefficients in the models, do they make sense? Are you keeping the model even though it is counter intuitive? Why? The boss needs to know.

```{r}
valid_data  <- transformed
predictors  <- inputs[!inputs  %in% c("indus","chas","lstat","rm","black","zn")]

model <- glm(I(target)~.,data=train[,c(predictors,target)],family    = binomial)

summary(model)

predicted  <- predict(model,test,type='response')
require("pROC")
d_roc <- roc(test$target,predicted)
plot(d_roc,  main = "ROC with pROC")
#ci(d_roc)
```

```
plot(model)

require("caret")
table(test$target,ifelse(predicted>0.5,1,0))

```


```{r}

d<- data.frame(class=test$target,scored.class=ifelse(predicted>0.5,1,0))

# let's use this helper function that will return all the rates for future calculations
confusion_matrix  <- function(d){
  data.frame(tp=nrow(d[d$class==1   & d$scored.class==1,]),
             tn=nrow(d[d$class==0  & d$scored.class==0,]),
             fp=nrow(d[d$class==0  & d$scored.class==1,]),
             fn=nrow(d[d$class==1  & d$scored.class==0,])
  )
}

confusion_matrix(d)
accuracy<-function(d){
  f <- confusion_matrix(d)
  (f$tp+f$tn)/(f$tp+f$fp+f$tn+f$fn)
}
accuracy(d)

classification_error_rate<-function(d){
  f <- confusion_matrix(d)
  (f$fp+f$fn)/(f$tp+f$fp+f$tn+f$fn)
}
classification_error_rate(d)

precision_c<-function(d){
  f <- confusion_matrix(d)
  (f$tp)/(f$tp+f$fp)
}
precision_c(d)

sensitivity_c<-function(d){
  f <- confusion_matrix(d)
  (f$tp)/(f$tp+f$fn)
}
sensitivity_c(d)

specificity_c<-function(d){
  f <- confusion_matrix(d)
  (f$tn)/(f$tn+f$fp)
}
specificity_c(d)


f1_score<-function(d){
  p<- precision_c(d)
  s<- sensitivity_c(d)
  2*p*s/(p+s)
}
f1_score(d)
```

```

## 4. SELECT MODELS (25 Points)

Decide on the criteria for selecting the best binary logistic regression model. Will you select models with slightly worse performance if it makes more sense or is more parsimonious? Discuss why you selected your models.

For the binary logistic regression model, will you use a metric such as log likelihood, AIC, ROC curve, etc.? Using the training data set, evaluate the binary logistic regression model based on (a) accuracy, (b) classification error rate, (c) precision, (d) sensitivity, (e) specificity, (f) F1 score, (g) AUC, and (h) confusion matrix. Make predictions using the evaluation data set.


# Predictions

```{r}

mean_sd <- function(x){
  c( qnorm(0.05,mean(x),sd(x)),   qnorm(0.95,mean(x),sd(x))   )
}


trans_params<-data.frame(apply(   transformed[,inputs],2,   mean_sd))

cap <- function(col){
    quantiles <- trans_params[,col]
    x <- evaluation[,col]
    x[ x < quantiles[1]  ] <- max(0,quantiles[1])
    x[ x > quantiles[2]  ] <- quantiles[2]
    x
}

#transformed$black  <- log(transformed$black)

evaluation_transformed  <- data.frame(sapply(inputs,function(x){cap(x)}))
evaluation$predict  <- predict(model,evaluation_transformed,type='response')>0.5

kable(data.frame(evaluation))
```
```