# Logistic Regression Assignment

## BAN502 Predictive Analytics

### Susan Wiggins

Libraries for this Project:

library(tidyverse)

## -- Attaching packages ------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.1 v purrr 0.3.2  
## v tibble 2.1.1 v dplyr 0.8.1  
## v tidyr 0.8.3 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts ---------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(leaps)  
library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(e1071) #often needed for various statistical tasks  
library(ROCR) #for threshold selction

## Loading required package: gplots

##   
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':  
##   
## lowess

library(readr)  
parole <- read\_csv("parole.csv")

## Parsed with column specification:  
## cols(  
## male = col\_double(),  
## race = col\_double(),  
## age = col\_double(),  
## state = col\_double(),  
## time.served = col\_double(),  
## max.sentence = col\_double(),  
## multiple.offenses = col\_double(),  
## crime = col\_double(),  
## violator = col\_double()  
## )

View(parole)

Code to convert Parole data:

parole = parole %>% mutate(male = as\_factor(as.character(male)))%>%   
mutate(male = fct\_recode(male, "male" = "1", "female" = "0"))  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <dbl> 1, 1, 2, 1, 2, 2, 1, 1, 1, 2, 2, 1, 1, 1, 1,...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,...  
## $ crime <dbl> 4, 3, 3, 1, 1, 4, 3, 1, 3, 2, 1, 1, 3, 3, 3,...  
## $ violator <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...

parole = parole %>% mutate(race = as\_factor(as.character(race)))%>%   
mutate(race = fct\_recode(race, "white" = "1", "otherwise" = "2"))  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <fct> white, white, otherwise, white, otherwise, o...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,...  
## $ crime <dbl> 4, 3, 3, 1, 1, 4, 3, 1, 3, 2, 1, 1, 3, 3, 3,...  
## $ violator <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...

parole = parole %>% mutate(state = as\_factor(as.character(state)))%>%   
mutate(state = fct\_recode(state, "Kentucky" = "2", "Louisiana" = "3", "Virginia" = "4", "other state" = "1"))  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <fct> white, white, otherwise, white, otherwise, o...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <fct> other state, other state, other state, other...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,...  
## $ crime <dbl> 4, 3, 3, 1, 1, 4, 3, 1, 3, 2, 1, 1, 3, 3, 3,...  
## $ violator <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...

parole = parole %>% mutate(crime = as\_factor(as.character(crime)))%>%   
mutate(crime = fct\_recode(crime, "larceny" = "2", "drug-related crime" = "3", "driving-related crime" = "4", "other crime" = "1"))  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <fct> white, white, otherwise, white, otherwise, o...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <fct> other state, other state, other state, other...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,...  
## $ crime <fct> driving-related crime, drug-related crime, d...  
## $ violator <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...

parole = parole %>% mutate(violator = as\_factor(as.character(violator)))%>%   
mutate(violator = fct\_recode(violator, "violated parole" = "1", "No violation parole" = "0"))  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <fct> white, white, otherwise, white, otherwise, o...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <fct> other state, other state, other state, other...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,...  
## $ crime <fct> driving-related crime, drug-related crime, d...  
## $ violator <fct> No violation parole, No violation parole, No...

parole = parole %>% mutate(multiple.offenses = as\_factor(as.character(multiple.offenses)))%>%   
mutate(multiple.offenses = fct\_recode(multiple.offenses, "Multiple Offenses" = "1", "Otherwise" = "0"))  
  
  
glimpse(parole)

## Observations: 675  
## Variables: 9  
## $ male <fct> male, female, male, male, male, male, male, ...  
## $ race <fct> white, white, otherwise, white, otherwise, o...  
## $ age <dbl> 33.2, 39.7, 29.5, 22.4, 21.6, 46.7, 31.0, 24...  
## $ state <fct> other state, other state, other state, other...  
## $ time.served <dbl> 5.5, 5.4, 5.6, 5.7, 5.4, 6.0, 6.0, 4.8, 4.5,...  
## $ max.sentence <dbl> 18, 12, 12, 18, 12, 18, 18, 12, 13, 12, 12, ...  
## $ multiple.offenses <fct> Otherwise, Otherwise, Otherwise, Otherwise, ...  
## $ crime <fct> driving-related crime, drug-related crime, d...  
## $ violator <fct> No violation parole, No violation parole, No...

### Task 1

Split the data into training and testing sets.

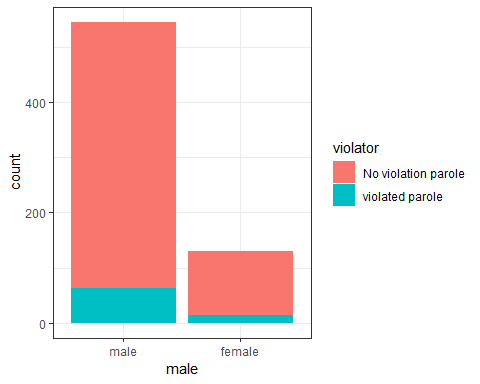
set.seed(12345)  
train.rows = createDataPartition(y = parole$violator, p=0.7, list = FALSE) #70% in training  
train = parole[train.rows,]   
test = parole[-train.rows,]  
  
  
y <- train[,9]  
y\_test <- test[,9]

### Task 2

In this task, use appropriate data visualizations and/or tables to identify which variables in the training set appear to be most predictive of the response variable “violator”. Provide a brief explanation of your thought process.

Variable Gender:

ggplot(parole, aes(x=male, fill = violator)) + geom\_bar() + theme\_bw()

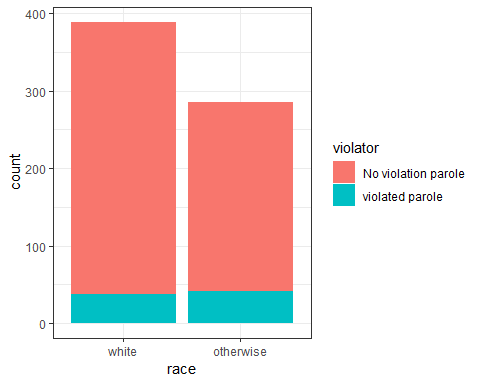
 Tabular data for Gender:

t1 = table(parole$violator, parole$male) #create a table object  
prop.table(t1, margin = 2 ) #crosstab with proportions

##   
## male female  
## No violation parole 0.8825688 0.8923077  
## violated parole 0.1174312 0.1076923

variable race:

ggplot(parole, aes(x=race, fill = violator)) + geom\_bar() + theme\_bw()



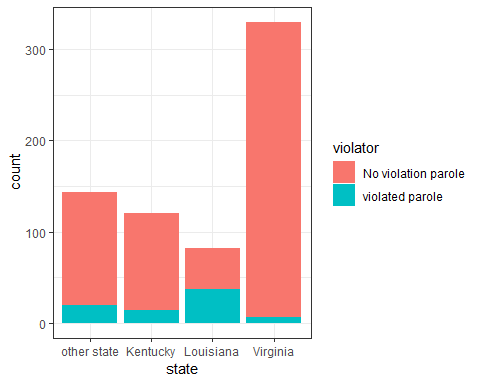
Tabular data for race:

t2 = table(parole$violator, parole$race) #create a table object  
prop.table(t2, margin = 2 ) #crosstab with proportions

##   
## white otherwise  
## No violation parole 0.90488432 0.85664336  
## violated parole 0.09511568 0.14335664

variable state:

ggplot(parole, aes(x=state, fill = violator)) + geom\_bar() + theme\_bw()



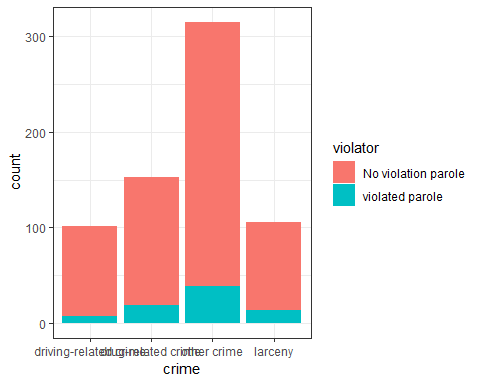
Tabular data for state:

t3 = table(parole$violator, parole$state) #create a table object  
prop.table(t3, margin = 2 ) #crosstab with proportions

##   
## other state Kentucky Louisiana Virginia  
## No violation parole 0.86013986 0.88333333 0.54878049 0.97878788  
## violated parole 0.13986014 0.11666667 0.45121951 0.02121212

variable crime:

ggplot(parole, aes(x=crime, fill = violator)) + geom\_bar() + theme\_bw()



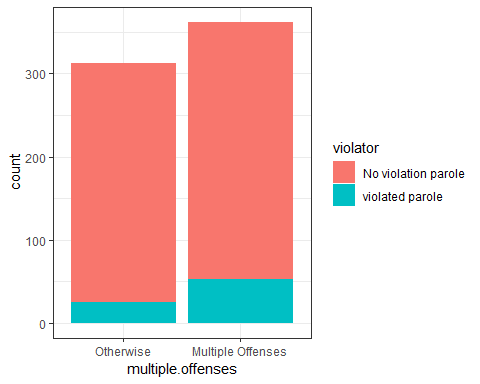
Tabular data for crime:

t4 = table(parole$violator, parole$crime) #create a table object  
prop.table(t4, margin = 2 ) #crosstab with proportions

##   
## driving-related crime drug-related crime other crime  
## No violation parole 0.93069307 0.87581699 0.87619048  
## violated parole 0.06930693 0.12418301 0.12380952  
##   
## larceny  
## No violation parole 0.87735849  
## violated parole 0.12264151

variable multiple Offenses:

ggplot(parole, aes(x=multiple.offenses, fill = violator)) + geom\_bar() + theme\_bw()



Tabular data for multiple Offenses:

t5 = table(parole$violator, parole$multiple.offenses) #create a table object  
prop.table(t5, margin = 2 ) #crosstab with proportions

##   
## Otherwise Multiple Offenses  
## No violation parole 0.9201278 0.8535912  
## violated parole 0.0798722 0.1464088

### Task 2 (Continued)

Provide a brief explanation of your thought process.

Gender tends to have a significance by visualization aspects and also looking towards the tabular data generated in the analysis. There are more males who violate their parole than females; however, the tabular data show only a difference of .0097389.

Race tends to have a slightly higher level in parole violators for the other categorical listed as “otherwise”. An increase in 4.824 % for violators for otherwise. Visually the data dosn’t indicate significance in value.

State looks that it does indicate that it has significance on the violation of parole, the people in Louisiana tend to violate parole more than the other states at 45.12 % and other states is next at percentage rate of 13.98 %. Virginia has the least people who violate their parole.

Crime looks as though the people with driving related crimes are less likely to violate their parole than the other offenses. Drug-related crime is the most likely to violate their parole.

Mutiple Offenses tends to have a greater significane also with people skipping out on their parole due to having many offenses. Mutiple offenses skip out on parole at 14% and otherwise less offenses is around 7.9 %.

### Task 3

Identify the variable from Task 2 that appears to you to be most predictive of “violator”.

When looking at the comparison of the most predictive of violator, I thought that the variable mutiple offenses was the mos significant of variables towards violation of parole followed by what state the criminal offender was in next.

model1 = glm(violator ~ multiple.offenses, parole, family = "binomial")  
summary(model1)

##   
## Call:  
## glm(formula = violator ~ multiple.offenses, family = "binomial",   
## data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.5627 -0.5627 -0.4080 -0.4080 2.2483   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.4441 0.2085 -11.722 < 2e-16  
## multiple.offensesMultiple Offenses 0.6810 0.2561 2.659 0.00783  
##   
## (Intercept) \*\*\*  
## multiple.offensesMultiple Offenses \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 475.81 on 673 degrees of freedom  
## AIC: 479.81  
##   
## Number of Fisher Scoring iterations: 5

### Task 3 (Continued)

Comment on the quality of the model.

Multiple Offenses does show significant value the p value is less than .05 at a value of .00783.The AIC also is significantly low at 479.81. We must look for changes in a decrease in AIC for an indication of the degree of how good the model is in this instance. The negative value also lets us know that less likely for the person to be a parole violator of -2.44.

model2 = glm(violator ~ multiple.offenses + state, parole, family = "binomial")  
summary(model2)

##   
## Call:  
## glm(formula = violator ~ multiple.offenses + state, family = "binomial",   
## data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.2897 -0.4398 -0.2282 -0.2282 2.7061   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.3949 0.2955 -8.104 5.34e-16  
## multiple.offensesMultiple Offenses 1.5257 0.3166 4.819 1.44e-06  
## stateKentucky 0.1075 0.3927 0.274 0.78429  
## stateLouisiana 1.1295 0.3521 3.208 0.00134  
## stateVirginia -2.7662 0.4783 -5.784 7.32e-09  
##   
## (Intercept) \*\*\*  
## multiple.offensesMultiple Offenses \*\*\*  
## stateKentucky   
## stateLouisiana \*\*   
## stateVirginia \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 358.69 on 670 degrees of freedom  
## AIC: 368.69  
##   
## Number of Fisher Scoring iterations: 6

### Task 4

Using forward stepwise, backward stepwise, or by manually building a model, create the best model you can to predict “violator”.

allmod = glm(violator ~ male + race + state + crime + multiple.offenses, parole, family = "binomial")   
summary(allmod)

##   
## Call:  
## glm(formula = violator ~ male + race + state + crime + multiple.offenses,   
## family = "binomial", data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5968 -0.3974 -0.2598 -0.1570 2.9083   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.49884 0.46790 -5.341 9.27e-08  
## malefemale -0.32572 0.36762 -0.886 0.3756  
## raceotherwise 0.76012 0.32104 2.368 0.0179  
## stateKentucky 0.13671 0.41244 0.331 0.7403  
## stateLouisiana 0.73326 0.39741 1.845 0.0650  
## stateVirginia -3.22589 0.52179 -6.182 6.32e-10  
## crimedrug-related crime -0.17562 0.53301 -0.329 0.7418  
## crimeother crime 0.08241 0.47660 0.173 0.8627  
## crimelarceny 0.44252 0.57345 0.772 0.4403  
## multiple.offensesMultiple Offenses 1.51015 0.32166 4.695 2.67e-06  
##   
## (Intercept) \*\*\*  
## malefemale   
## raceotherwise \*   
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## crimedrug-related crime   
## crimeother crime   
## crimelarceny   
## multiple.offensesMultiple Offenses \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 351.00 on 665 degrees of freedom  
## AIC: 371  
##   
## Number of Fisher Scoring iterations: 6

emptymod = glm(violator ~1, parole, family = "binomial") #use ~1 to build an empty model   
summary(emptymod)

##   
## Call:  
## glm(formula = violator ~ 1, family = "binomial", data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.4956 -0.4956 -0.4956 -0.4956 2.0775   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.0352 0.1204 -16.9 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 483.27 on 674 degrees of freedom  
## AIC: 485.27  
##   
## Number of Fisher Scoring iterations: 4

Backward Stepwise:

#backward  
backmod = stepAIC(allmod, direction = "backward", trace = TRUE)

## Start: AIC=371  
## violator ~ male + race + state + crime + multiple.offenses  
##   
## Df Deviance AIC  
## - crime 3 352.71 366.71  
## - male 1 351.81 369.81  
## <none> 351.00 371.00  
## - race 1 356.55 374.55  
## - multiple.offenses 1 374.04 392.04  
## - state 3 469.86 483.86  
##   
## Step: AIC=366.71  
## violator ~ male + race + state + multiple.offenses  
##   
## Df Deviance AIC  
## - male 1 353.26 365.26  
## <none> 352.71 366.71  
## - race 1 358.04 370.04  
## - multiple.offenses 1 376.16 388.16  
## - state 3 473.67 481.67  
##   
## Step: AIC=365.26  
## violator ~ race + state + multiple.offenses  
##   
## Df Deviance AIC  
## <none> 353.26 365.26  
## - race 1 358.69 368.69  
## - multiple.offenses 1 376.71 386.71  
## - state 3 473.73 479.73

summary(backmod)

##   
## Call:  
## glm(formula = violator ~ race + state + multiple.offenses, family = "binomial",   
## data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4012 -0.4051 -0.2604 -0.1801 2.8739   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.50359 0.30055 -8.330 < 2e-16  
## raceotherwise 0.74594 0.31828 2.344 0.0191  
## stateKentucky 0.04449 0.39449 0.113 0.9102  
## stateLouisiana 0.75016 0.39147 1.916 0.0553  
## stateVirginia -3.12945 0.51147 -6.119 9.44e-10  
## multiple.offensesMultiple Offenses 1.51964 0.32027 4.745 2.09e-06  
##   
## (Intercept) \*\*\*  
## raceotherwise \*   
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## multiple.offensesMultiple Offenses \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 353.26 on 669 degrees of freedom  
## AIC: 365.26  
##   
## Number of Fisher Scoring iterations: 6

Forward Stepwise:

#forward  
forwardmod = stepAIC(emptymod, direction = "forward", scope=list(upper=allmod,lower=emptymod),  
 trace = TRUE)

## Start: AIC=485.27  
## violator ~ 1  
##   
## Df Deviance AIC  
## + state 3 382.89 390.89  
## + multiple.offenses 1 475.81 479.81  
## + race 1 479.56 483.56  
## <none> 483.27 485.27  
## + male 1 483.17 487.17  
## + crime 3 480.48 488.48  
##   
## Step: AIC=390.89  
## violator ~ state  
##   
## Df Deviance AIC  
## + multiple.offenses 1 358.69 368.69  
## + race 1 376.71 386.71  
## <none> 382.89 390.89  
## + male 1 382.16 392.16  
## + crime 3 380.87 394.87  
##   
## Step: AIC=368.69  
## violator ~ state + multiple.offenses  
##   
## Df Deviance AIC  
## + race 1 353.26 365.26  
## <none> 358.69 368.69  
## + male 1 358.04 370.04  
## + crime 3 357.47 373.47  
##   
## Step: AIC=365.26  
## violator ~ state + multiple.offenses + race  
##   
## Df Deviance AIC  
## <none> 353.26 365.26  
## + male 1 352.71 366.71  
## + crime 3 351.81 369.81

summary(forwardmod)

##   
## Call:  
## glm(formula = violator ~ state + multiple.offenses + race, family = "binomial",   
## data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4012 -0.4051 -0.2604 -0.1801 2.8739   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.50359 0.30055 -8.330 < 2e-16  
## stateKentucky 0.04449 0.39449 0.113 0.9102  
## stateLouisiana 0.75016 0.39147 1.916 0.0553  
## stateVirginia -3.12945 0.51147 -6.119 9.44e-10  
## multiple.offensesMultiple Offenses 1.51964 0.32027 4.745 2.09e-06  
## raceotherwise 0.74594 0.31828 2.344 0.0191  
##   
## (Intercept) \*\*\*  
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## multiple.offensesMultiple Offenses \*\*\*  
## raceotherwise \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 353.26 on 669 degrees of freedom  
## AIC: 365.26  
##   
## Number of Fisher Scoring iterations: 6

The models for forward and backward stepwise regression are the same. The AIC value is listed at the same value. The forward and backward stepwise regression models displayed a better AIC value than the previous model that I built manually with the variables of mutiple.offenses and state towards response variable of violator. The forward regression shows that state Kentucky is not significant and state Louisiana, whereas, state Virginia, mutiple offenses, and race otherwise were significant due to p-value. The backward regression shows the same significance in these variables due to p-value.

### Task 5

Create a logistic regression model using the training set to predict “violator” using the variables: state, multiple offenses, and race.

model3 = glm(violator ~ multiple.offenses + state + race, parole, family = "binomial")  
summary(model3)

##   
## Call:  
## glm(formula = violator ~ multiple.offenses + state + race, family = "binomial",   
## data = parole)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4012 -0.4051 -0.2604 -0.1801 2.8739   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.50359 0.30055 -8.330 < 2e-16  
## multiple.offensesMultiple Offenses 1.51964 0.32027 4.745 2.09e-06  
## stateKentucky 0.04449 0.39449 0.113 0.9102  
## stateLouisiana 0.75016 0.39147 1.916 0.0553  
## stateVirginia -3.12945 0.51147 -6.119 9.44e-10  
## raceotherwise 0.74594 0.31828 2.344 0.0191  
##   
## (Intercept) \*\*\*  
## multiple.offensesMultiple Offenses \*\*\*  
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## raceotherwise \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 483.27 on 674 degrees of freedom  
## Residual deviance: 353.26 on 669 degrees of freedom  
## AIC: 365.26  
##   
## Number of Fisher Scoring iterations: 6

### Task 5 (Continued)

The variables that are less likely to commit violation of parole would be mutiple offenses otherwise which indicates a negative value as well as the state of Virginia. The significant variables shown by there p-value are mutiple offenses otherwise and multiple offenses, state of Virginis and race otherwise in this model. The AIC value for this model is the same in value as the other models performed earlier for forward, backward, and the previous model os model1 and model2.

### Task 6

Parolee 1: Parolee 1 has a 44.18 % likely to violate parole.

newdata = data.frame(multiple.offenses = "Multiple Offenses", state = "Louisiana", race = "white")  
predict(forwardmod, newdata, type="response")

## 1   
## 0.4418174

Parolee 2: Parolee 2 has a 15.28 % likely to violate parole. Therefore, Parolee 1 would be the most likely to violate parole with mutiple offenses, white, and coming from state of Louisiana.

newdata = data.frame(multiple.offenses = "Otherwise", state = "Kentucky", race = "otherwise")  
predict(forwardmod, newdata, type="response")

## 1   
## 0.152755

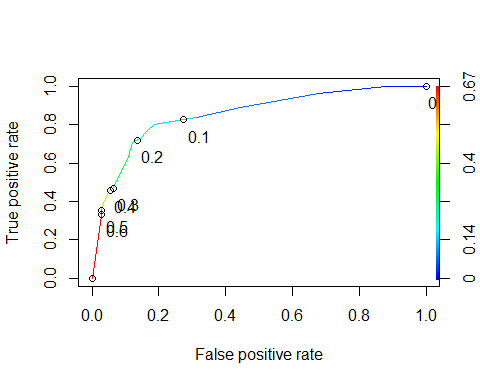
### Task 7

Develop an ROC curve and determine the probability threshold that best balances specifiicity and sensitivity (on the training set).

predictions = predict(model3,newdata = train, type="response") #develop predicted probabilities  
head(predictions)

## 1 2 3 4 5 6   
## 0.07560706 0.14708578 0.07560706 0.14708578 0.14708578 0.07560706

ROCRpred = prediction(predictions,train$violator)   
  
ROCRperf = performance(ROCRpred, "tpr", "fpr")  
plot(ROCRperf, colorize=TRUE, print.cutoffs.at=seq(0,1,by=0.1), text.adj=c(-0.2,1.7))



### Task 8

What is the accuracy, sensitivity, and speci???city of the model on the training set given the cuto??? from Task 7?

The cut off value is at 0.1470858, the sensitivity of the model is 0.8000000, and the specificity is at 0.8133971 on the training data set. The Accuracy of the model is listed futher down and is 0.8287526.

#Determine threshold to balance sensitivity and specificity  
#DO NOT modify this code  
opt.cut = function(perf, pred){  
 cut.ind = mapply(FUN=function(x, y, p){  
 d = (x - 0)^2 + (y-1)^2  
 ind = which(d == min(d))  
 c(sensitivity = y[[ind]], specificity = 1-x[[ind]],   
 cutoff = p[[ind]])  
 }, perf@x.values, perf@y.values, pred@cutoffs)  
}  
print(opt.cut(ROCRperf, ROCRpred))

## [,1]  
## sensitivity 0.8000000  
## specificity 0.8133971  
## cutoff 0.1470858

Area under the curve (AUC). AUC is a measure of the strength of the model. Values closer to 1 are better. Can be used to compare models.

as.numeric(performance(ROCRpred, "auc")@y.values)

## [1] 0.8540887

The value of the model looks good due to the AUC value listed at 0.8540887, whhich is close to one.

### Task 9

Identify a probability threshold (via trial-and-error) that best maximizes accuracy on the training set.

test threshold:

#confusion matrix  
t6 = table(train$violator,predictions > 0.1470858)  
t6

##   
## FALSE TRUE  
## No violation parole 350 68  
## violated parole 13 42

Correct classification are that there are 350 at no violation parole false and 42 people who violated their parole and classified as true. Incorrect classification was No violation parole at true 68 and violated parole at 13 false.

Caculate Accuracy:

(t6[1,1]+t6[2,2])/nrow(train)

## [1] 0.8287526

Can apply trial and error to maximize accuracy (here trying 0.5 as threshold)

t6 = table(train$violator,predictions > 0.5)  
t6

##   
## FALSE TRUE  
## No violation parole 406 12  
## violated parole 37 18

(t6[1,1]+t6[2,2])/nrow(train)

## [1] 0.8964059

Accuracy increases from 0.8287526 to 0.8964059 using .5.

t6 = table(train$violator,predictions > 0.6)  
t6

##   
## FALSE TRUE  
## No violation parole 406 12  
## violated parole 37 18

(t6[1,1]+t6[2,2])/nrow(train)

## [1] 0.8964059

Accuracy remains the same when increasing to .6.

A naive prediction (everyone not skip parole)

t6 = table(train$violator,predictions > 1) #set threshold to 1 so all are classifed as not not skip parole  
t6

##   
## FALSE  
## No violation parole 418  
## violated parole 55

(t6[1])/nrow(train)

## [1] 0.8837209

The margin is not that great in a naive model towards other comparisons in trial and error. I would recommend the .5 for the threshold in the trial and error the accuracy increases at this particular threshold and doesn’t change from that point.

### Task 10

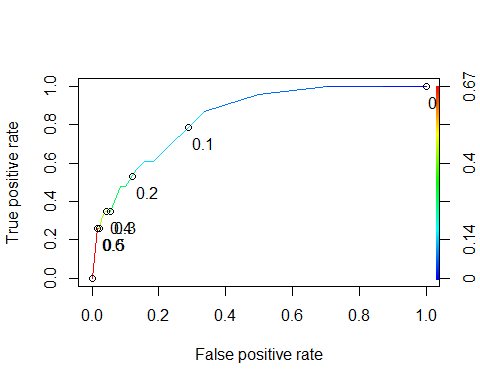
Use your probability threshold from Task 9 to determine accuracy of the model on the testing set.

test threshold:

predictions\_test = predict(model3,newdata = test, type="response") #develop predicted probabilities  
head(predictions\_test)

## 1 2 3 4 5 6   
## 0.07560706 0.44077735 0.07560706 0.07560706 0.07560706 0.07560706

ROCRpred\_test = prediction(predictions\_test,test$violator)   
  
ROCRperf\_test = performance(ROCRpred\_test, "tpr", "fpr")  
plot(ROCRperf\_test, colorize=TRUE, print.cutoffs.at=seq(0,1,by=0.1), text.adj=c(-0.2,1.7))



#Determine threshold to balance sensitivity and specificity  
#DO NOT modify this code  
opt.cut = function(perf, pred){  
 cut.ind = mapply(FUN=function(x, y, p){  
 d = (x - 0)^2 + (y-1)^2  
 ind = which(d == min(d))  
 c(sensitivity = y[[ind]], specificity = 1-x[[ind]],   
 cutoff = p[[ind]])  
 }, perf@x.values, perf@y.values, pred@cutoffs)  
}  
print(opt.cut(ROCRperf\_test, ROCRpred\_test))

## [,1]  
## sensitivity 0.86956522  
## specificity 0.66480447  
## cutoff 0.07877567

as.numeric(performance(ROCRpred\_test, "auc")@y.values)

## [1] 0.8395676

#confusion matrix  
t7 = table(train$violator,predictions > 0.07877567)  
t7

##   
## FALSE TRUE  
## No violation parole 340 78  
## violated parole 11 44

Calculate Accuracy:

(t7[1,1]+t7[2,2])/nrow(test)

## [1] 1.90099

t7 = table(test$violator,predictions\_test > 0.5)  
t7

##   
## FALSE TRUE  
## No violation parole 176 3  
## violated parole 17 6

(t7[1,1]+t7[2,2])/nrow(test)

## [1] 0.9009901

t7 = table(test$violator,predictions\_test > 0.6)  
t7

##   
## FALSE TRUE  
## No violation parole 176 3  
## violated parole 17 6

(t7[1,1]+t7[2,2])/nrow(test)

## [1] 0.9009901

t7 = table(test$violator,predictions\_test > 1) #set threshold to 1 so all are classifed as no skipping parole  
(t7[1])/nrow(test)

## [1] 0.8861386

The test data set moves to decreasing in value for skipping out on parole when setting threshold to one.