# Logistic Regression (Classification)

## Zac Coffey

Libraries

library("tidyverse")

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

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.7  
## v tidyr 0.8.2 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

## -- 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("caret")

## Loading required package: lattice

##   
## Attaching package: 'caret'

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

library("ROCR")

## Loading required package: gplots

##   
## Attaching package: 'gplots'

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

Read In Data

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

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

Data frame Conversions

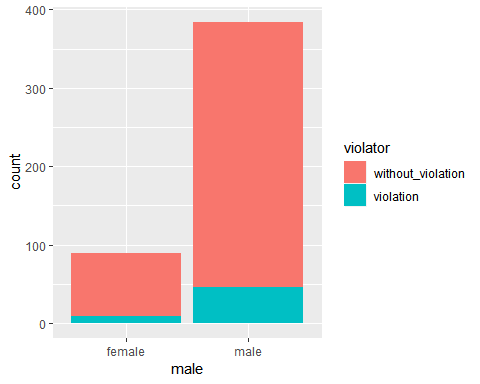
parole = parole %>% mutate(male = as.factor(as.character(male))) %>%  
 mutate(male = fct\_recode(male,  
 "female" = "0",  
 "male" = "1"))  
  
parole = parole %>% mutate(race = as.factor(as.character(race))) %>%  
 mutate(race = fct\_recode(race,  
 "white" = "1",  
 "otherwise" = "2"))  
  
parole = parole %>% mutate(state = as.factor(as.character(state))) %>%  
 mutate(state = fct\_recode(state,  
 "other\_state" = "1",  
 "Kentucky" = "2",  
 "Louisiana" = "3",  
 "Virginia" = "4"))  
  
parole = parole %>% mutate(crime = as.factor(as.character(crime))) %>%  
 mutate(crime = fct\_recode(crime,  
 "other\_crime" = "1",  
 "larceny" = "2",  
 "drug\_related" = "3",  
 "driving\_related" = "4"))  
  
parole = parole %>% mutate(multiple.offenses = as.factor(as.character(multiple.offenses))) %>%  
 mutate(multiple.offenses = fct\_recode(multiple.offenses,  
 "No" = "0",  
 "multiple\_offenses" = "1"))  
  
parole = parole %>% mutate(violator = as.factor(as.character(violator))) %>%  
 mutate(violator = fct\_recode(violator,  
 "without\_violation" = "0",  
 "violation" = "1"))

Task 1

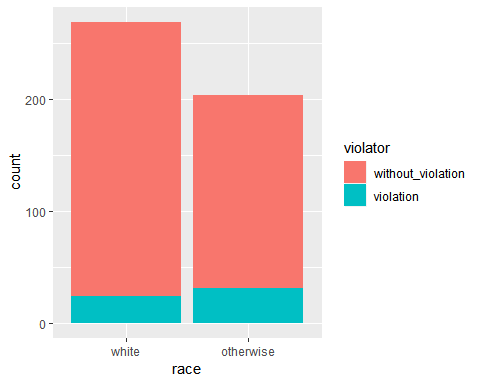
set.seed(12345)  
  
train.rows = createDataPartition(y = parole$violator, p=0.7, list = FALSE) #70% in training  
  
train = parole[train.rows,]   
test = parole[-train.rows,]

Task 2

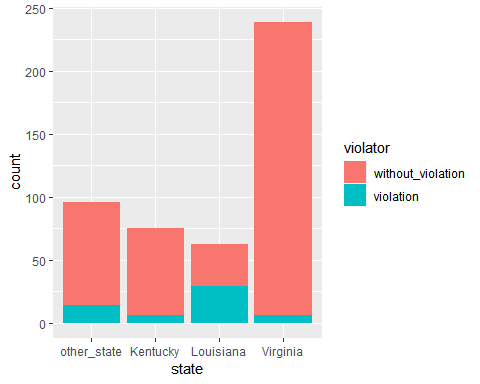
ggplot(train,aes(male, fill = violator))+geom\_bar()



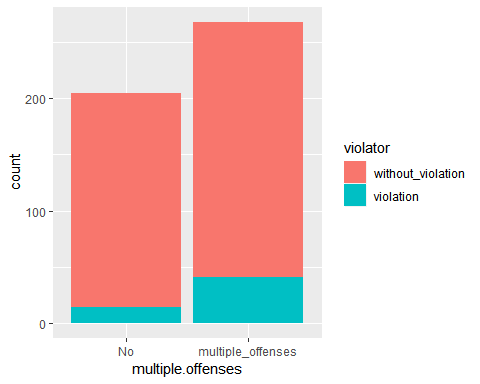
#Male(gender) is not a good predictor becasue male has a much larger count than female. This to me would skew the prediction. I do not think that (male or gender) is an appropriate predictor because males are tradionally incarcerated much more than females.  
  
ggplot(train,aes(race, fill = violator))+geom\_bar()



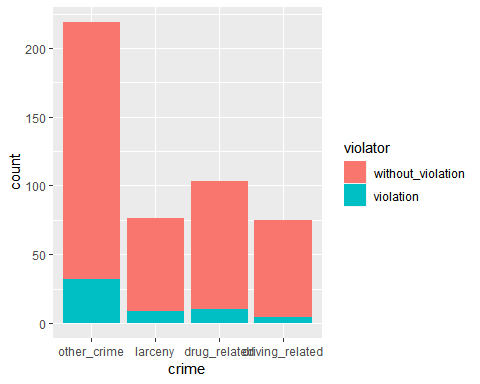
#Race is a good predictor because both variables have a large count. Also both variables appear to have an above average violation count, compared to other factors.However I do not think it is the best predictor because it is specific enough, "otherwise" includes a large quanity of demographics.   
  
ggplot(train,aes(state, fill = violator))+geom\_bar()



#State does not appear to be a good predictor becuase of the variable Virginia. I think that a good predictor should have a relatively equal count across all variables. Virginia has a much larger count than the rest of the variables in this factor.  
  
ggplot(train,aes(multiple.offenses, fill = violator))+geom\_bar()



#Multiple.offenses appears to be a great predictor. Logically this factor makes sense the most because it is expected for individuals who have multiple offenses to have violated thier parole, because they have had more opportunies to do so.   
  
ggplot(train,aes(crime, fill = violator))+geom\_bar()



#Crime does not appear to be a good predictor because of other\_crime variable. I think that the other\_crime variable is much larger than the other variables, and this could potentially skew the results. This factor could have been good if the other\_crime variable was removed.

Task 3

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

##   
## Call:  
## glm(formula = violator ~ multiple.offenses, family = "binomial",   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.5763 -0.5763 -0.3761 -0.3761 2.3169   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.6132 0.2769 -9.438 < 2e-16  
## multiple.offensesmultiple\_offenses 0.9018 0.3247 2.777 0.00549  
##   
## (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: 340.04 on 472 degrees of freedom  
## Residual deviance: 331.50 on 471 degrees of freedom  
## AIC: 335.5  
##   
## Number of Fisher Scoring iterations: 5

#The result appears to show a postive estimate for a individal with mulitple\_offenses.This means that an individual with multiple\_offenses is more likey to be a violator of parole than somone who is not.

Task 4

allmod = glm(violator ~., train, family = "binomial")   
summary(allmod)

##   
## Call:  
## glm(formula = violator ~ ., family = "binomial", data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.9635 -0.3638 -0.2354 -0.1449 2.9869   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -4.98802 1.36340 -3.659 0.000254  
## malemale 0.53377 0.49107 1.087 0.277051  
## raceotherwise 1.06698 0.41324 2.582 0.009824  
## age 0.03361 0.01696 1.982 0.047493  
## stateKentucky -0.30132 0.56939 -0.529 0.596665  
## stateLouisiana 0.87804 0.52428 1.675 0.093984  
## stateVirginia -3.46523 0.63742 -5.436 5.44e-08  
## time.served -0.03009 0.12159 -0.247 0.804537  
## max.sentence 0.08458 0.05644 1.499 0.133963  
## multiple.offensesmultiple\_offenses 1.72841 0.41857 4.129 3.64e-05  
## crimelarceny 0.18508 0.50343 0.368 0.713139  
## crimedrug\_related -0.76563 0.46946 -1.631 0.102918  
## crimedriving\_related -0.87795 0.62271 -1.410 0.158571  
##   
## (Intercept) \*\*\*  
## malemale   
## raceotherwise \*\*   
## age \*   
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## time.served   
## max.sentence   
## multiple.offensesmultiple\_offenses \*\*\*  
## crimelarceny   
## crimedrug\_related   
## crimedriving\_related   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 340.04 on 472 degrees of freedom  
## Residual deviance: 230.16 on 460 degrees of freedom  
## AIC: 256.16  
##   
## Number of Fisher Scoring iterations: 6

emptymod = glm(violator ~1, train, family = "binomial")   
summary(emptymod)

##   
## Call:  
## glm(formula = violator ~ 1, family = "binomial", data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.4972 -0.4972 -0.4972 -0.4972 2.0745   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.0281 0.1434 -14.14 <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: 340.04 on 472 degrees of freedom  
## Residual deviance: 340.04 on 472 degrees of freedom  
## AIC: 342.04  
##   
## Number of Fisher Scoring iterations: 4

Task 4

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

## Start: AIC=342.04  
## violator ~ 1  
##   
## Df Deviance AIC  
## + state 3 264.58 272.58  
## + max.sentence 1 321.79 325.79  
## + multiple.offenses 1 331.50 335.50  
## + race 1 335.64 339.64  
## + time.served 1 336.02 340.02  
## <none> 340.04 342.04  
## + age 1 338.27 342.27  
## + crime 3 334.34 342.34  
## + male 1 339.78 343.78  
##   
## Step: AIC=272.58  
## violator ~ state  
##   
## Df Deviance AIC  
## + multiple.offenses 1 246.88 256.88  
## + race 1 259.14 269.14  
## + age 1 262.48 272.48  
## <none> 264.58 272.58  
## + crime 3 259.43 273.43  
## + male 1 263.58 273.58  
## + time.served 1 264.29 274.29  
## + max.sentence 1 264.49 274.49  
##   
## Step: AIC=256.88  
## violator ~ state + multiple.offenses  
##   
## Df Deviance AIC  
## + race 1 240.42 252.42  
## <none> 246.88 256.88  
## + age 1 245.01 257.01  
## + max.sentence 1 245.58 257.58  
## + male 1 246.13 258.13  
## + time.served 1 246.88 258.88  
## + crime 3 242.93 258.93  
##   
## Step: AIC=252.42  
## violator ~ state + multiple.offenses + race  
##   
## Df Deviance AIC  
## + age 1 238.31 252.31  
## <none> 240.42 252.42  
## + max.sentence 1 238.81 252.81  
## + male 1 239.85 253.85  
## + time.served 1 240.37 254.37  
## + crime 3 236.69 254.69  
##   
## Step: AIC=252.31  
## violator ~ state + multiple.offenses + race + age  
##   
## Df Deviance AIC  
## + max.sentence 1 236.28 252.28  
## <none> 238.31 252.31  
## + male 1 237.41 253.41  
## + crime 3 233.88 253.88  
## + time.served 1 238.18 254.18  
##   
## Step: AIC=252.28  
## violator ~ state + multiple.offenses + race + age + max.sentence  
##   
## Df Deviance AIC  
## <none> 236.28 252.28  
## + male 1 235.38 253.38  
## + crime 3 231.56 253.56  
## + time.served 1 236.12 254.12

summary(forwardstepwise)

##   
## Call:  
## glm(formula = violator ~ state + multiple.offenses + race + age +   
## max.sentence, family = "binomial", data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.7414 -0.3643 -0.2668 -0.1502 2.7714   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -4.60426 1.13428 -4.059 4.92e-05  
## stateKentucky -0.41360 0.54930 -0.753 0.45147  
## stateLouisiana 0.86000 0.51900 1.657 0.09751  
## stateVirginia -3.34208 0.62057 -5.386 7.22e-08  
## multiple.offensesmultiple\_offenses 1.77974 0.41476 4.291 1.78e-05  
## raceotherwise 1.07386 0.40527 2.650 0.00806  
## age 0.02636 0.01660 1.588 0.11224  
## max.sentence 0.07733 0.05475 1.412 0.15788  
##   
## (Intercept) \*\*\*  
## stateKentucky   
## stateLouisiana .   
## stateVirginia \*\*\*  
## multiple.offensesmultiple\_offenses \*\*\*  
## raceotherwise \*\*   
## age   
## max.sentence   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 340.04 on 472 degrees of freedom  
## Residual deviance: 236.28 on 465 degrees of freedom  
## AIC: 252.28  
##   
## Number of Fisher Scoring iterations: 6

#Based on the results the quality of the model is good. The AIC consistently lowers as each variable is added. Not all of the ending variables are significant, however (race-otherwise,State-virginia,multipleoffenses-multiple\_offenses) are. The model appears to be intuitive to me because the intercept is significant and the model logically makes sense.

Task 5

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

##   
## Call:  
## glm(formula = violator ~ state + multiple.offenses + race, family = "binomial",   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4553 -0.3862 -0.2931 -0.1787 2.8791   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.5582 0.3709 -6.898 5.28e-12  
## stateKentucky -0.4816 0.5417 -0.889 0.3740  
## stateLouisiana 0.5292 0.4769 1.110 0.2672  
## stateVirginia -3.2301 0.6028 -5.358 8.39e-08  
## multiple.offensesmultiple\_offenses 1.6596 0.3985 4.165 3.12e-05  
## raceotherwise 1.0024 0.3966 2.528 0.0115  
##   
## (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: 340.04 on 472 degrees of freedom  
## Residual deviance: 240.42 on 467 degrees of freedom  
## AIC: 252.42  
##   
## Number of Fisher Scoring iterations: 6

#Based on the results the variables that are signicant are (state-virginia,multiple.offenses-multiple\_offenses). The AIC appears to almost be equal to the AIC result in the forward stepwise problem. Also the intercept is significant. All of these points leads me to belive that the quality of the model is relatively good.

Task 6

Parolee1 = data.frame(state = "Louisiana", multiple.offenses = "multiple\_offenses", race = "white")  
  
predict(model2,Parolee1,type = "response")

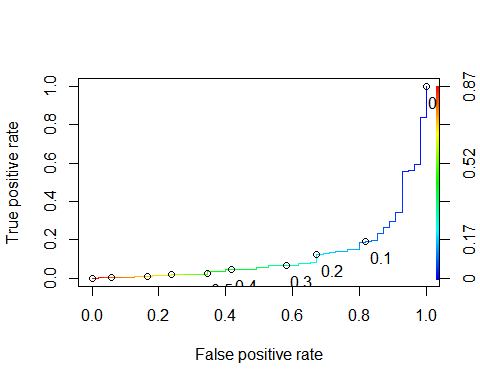
## 1   
## 0.408682

Parolee2 = data.frame(state = "Kentucky",multiple.offenses = "No", race = "otherwise")  
  
predict(model2,Parolee2, type = "response")

## 1   
## 0.1153326

Task 7

Model3 = glm(violator~.,train,family = "binomial")  
  
predictions = predict(Model3, type="response")  
  
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

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 0.83732057  
## specificity 0.01818182  
## cutoff 0.01166761

#Incorrctly classifying parolee will lower the sensitivity level. This data has a high sensitivity level meaning that classification has primarily been completed correctly

Task 9

t1 = table(train$violator,predictions > 0.5)  
t1

##   
## FALSE TRUE  
## without\_violation 409 9  
## violation 36 19

(t1[1,1]+t1[2,2])/nrow(train)

## [1] 0.9048626

t2 = table(train$violator,predictions > 0.6)  
t2

##   
## FALSE TRUE  
## without\_violation 411 7  
## violation 42 13

(t2[1,1]+t2[2,2])/nrow(train)

## [1] 0.8964059

t3 = table(train$violator,predictions > 1)  
t3

##   
## FALSE  
## without\_violation 418  
## violation 55

(t3[1,1])/nrow(train)

## [1] 0.8837209

#The probability threshold that best maximizes the accuracy on training is (> 0.5)

Task 10

Model4 = glm(violator~.,test,family = "binomial")  
  
predictions2 = predict(Model4, type="response")  
  
t4 = table(test$violator,predictions2 > 0.5)  
t4

##   
## FALSE TRUE  
## without\_violation 176 3  
## violation 16 7

(t4[1,1]+t1[2,2])/nrow(test)

## [1] 0.9653465