### Logistic Regression (Classification)

### BAN502

### Module 3; Assignment 2

### Mayberry, Brandon

Libraries

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.5.2

## -- 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(ROCR)

## Warning: package 'ROCR' was built under R version 3.5.2

## Loading required package: gplots

## Warning: package 'gplots' was built under R version 3.5.2

##   
## Attaching package: 'gplots'

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

library(caret)

## Warning: package 'caret' was built under R version 3.5.2

## Loading required package: lattice

##   
## Attaching package: 'caret'

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

Read in dataset Parole

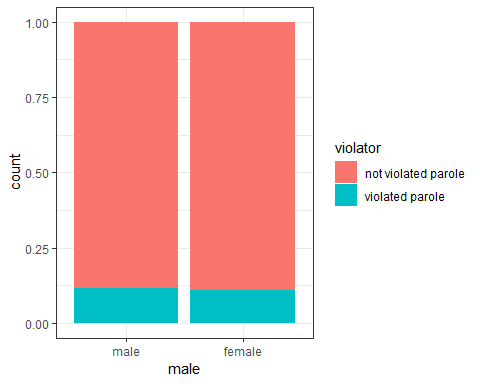
parole = read.csv("parole.csv")   
parole = parole %>% mutate(male = as\_factor(as.character(male))) %>%  
 mutate(male = fct\_recode(male,  
 "male" = "1",  
 "female" = "0")) %>%  
 mutate(race = as\_factor(as.character(race))) %>%  
 mutate(race = fct\_recode(race,  
 "white" = "1",  
 "otherwise" = "2")) %>%  
 mutate(state = as\_factor(as.character(state))) %>%  
 mutate(state = fct\_recode(state,  
 "any other state" = "1",  
 "Kentucky" = "2",  
 "Louisiana" = "3",  
 "Virginia" = "4")) %>%  
 mutate(crime = as\_factor(as.character(crime))) %>%  
 mutate(crime = fct\_recode(crime,  
 "any other crime" = "1",  
 "larceny" = "2",  
 "drug-related crime" = "3",  
 "driving-related crime" = "4")) %>%  
 mutate(multiple.offenses = as\_factor(as.character(multiple.offenses))) %>%  
 mutate(multiple.offenses = fct\_recode(multiple.offenses,  
 "multiple offenses" = "1",  
 "otherwise" ="0")) %>%  
 mutate(violator = as\_factor(as.character(violator))) %>%  
 mutate(violator = fct\_recode(violator,  
 "violated parole" = "1",  
 "not violated parole" = "0"))

Task 1

set.seed(1234)  
train.rows = createDataPartition(y = parole$violator, p=0.7, list = FALSE)  
train = parole[train.rows,]   
test = parole[-train.rows,]

Task 2 Visuals Male

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



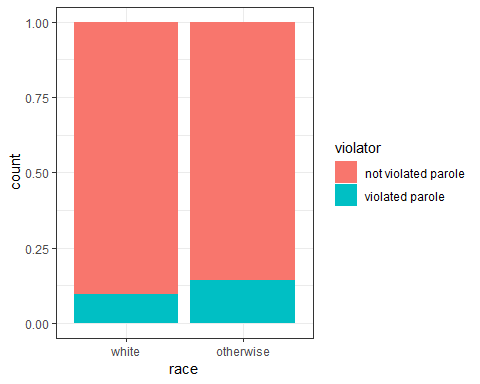
t1 = table(parole$violator, parole$male)  
prop.table(t1, margin = 2 )

##   
## male female  
## not violated parole 0.8825688 0.8923077  
## violated parole 0.1174312 0.1076923

Looks like the gender of the inmate doesn’t have any effect of whether or not they will violat parole. The table shows that the percentages of males who violate parole is very similiar to the percentages of femailes who violate thier parole.

Race

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



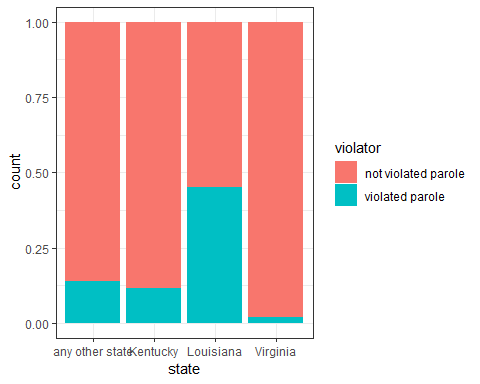
t2 = table(parole$violator, parole$race)  
prop.table(t2, margin = 2 )

##   
## white otherwise  
## not violated parole 0.90488432 0.85664336  
## violated parole 0.09511568 0.14335664

Race does seem to have a slight effect in predicting whether or not an inmate is likely to violate their parole.

State

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



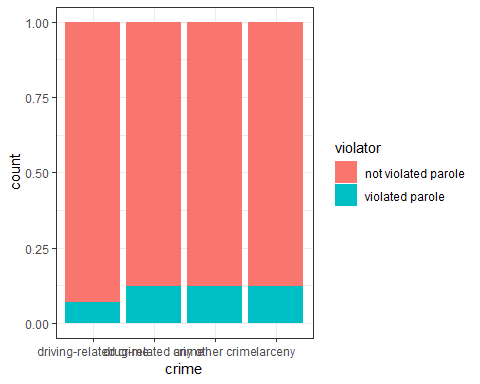
t3 = table(parole$violator, parole$state)  
prop.table(t3, margin = 2 )

##   
## any other state Kentucky Louisiana Virginia  
## not violated parole 0.86013986 0.88333333 0.54878049 0.97878788  
## violated parole 0.13986014 0.11666667 0.45121951 0.02121212

State does seem to have a significant effect in predicting whether or not an inmate is likely to violate thier parole.

Crime

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



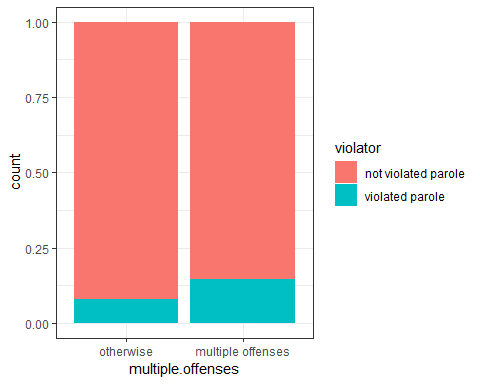
t4 = table(parole$violator, parole$crime)  
prop.table(t4, margin = 2 )

##   
## driving-related crime drug-related crime  
## not violated parole 0.93069307 0.87581699  
## violated parole 0.06930693 0.12418301  
##   
## any other crime larceny  
## not violated parole 0.87619048 0.87735849  
## violated parole 0.12380952 0.12264151

While driving-related crimes look be be less likely to violate thier parole, the other types of crimes do not have an effect on predicting whether or not an inmate will violate their parole.

multiple offenses

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



t5 = table(parole$violator, parole$multiple.offenses)  
prop.table(t5, margin = 2 )

##   
## otherwise multiple offenses  
## not violated parole 0.9201278 0.8535912  
## violated parole 0.0798722 0.1464088

Multiple offenses do seem to make a signifcant difference on whether or not an inmate will violate thier parole.

Task 3 Logistic Regression Model

mod1 = glm(violator ~ multiple.offenses , parole, family = "binomial")  
summary(mod1)

##   
## 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

The quality of the model seems good as the AIC is a low number

Task 4 Building the best model

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

##   
## Call:  
## glm(formula = violator ~ male + race + state + crime + multiple.offenses,   
## family = "binomial", data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5603 -0.4177 -0.2565 -0.1640 2.8884   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.81478 0.59294 -4.747 2.06e-06  
## malefemale 0.09574 0.40274 0.238 0.812090  
## raceotherwise 0.75798 0.38042 1.992 0.046318  
## stateKentucky 0.02719 0.48663 0.056 0.955440  
## stateLouisiana 0.70968 0.47155 1.505 0.132327  
## stateVirginia -3.06153 0.62058 -4.933 8.08e-07  
## crimedrug-related crime 0.38446 0.65151 0.590 0.555124  
## crimeany other crime 0.41937 0.60630 0.692 0.489127  
## crimelarceny 0.81648 0.70055 1.165 0.243823  
## multiple.offensesmultiple offenses 1.30104 0.38482 3.381 0.000722  
##   
## (Intercept) \*\*\*  
## malefemale   
## raceotherwise \*   
## stateKentucky   
## stateLouisiana   
## stateVirginia \*\*\*  
## crimedrug-related crime   
## crimeany other 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: 340.04 on 472 degrees of freedom  
## Residual deviance: 252.43 on 463 degrees of freedom  
## AIC: 272.43  
##   
## 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

Backward Stepwise

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

## Start: AIC=272.43  
## violator ~ male + race + state + crime + multiple.offenses  
##   
## Df Deviance AIC  
## - crime 3 253.90 267.90  
## - male 1 252.49 270.49  
## <none> 252.43 272.43  
## - race 1 256.36 274.36  
## - multiple.offenses 1 264.29 282.29  
## - state 3 329.63 343.63  
##   
## Step: AIC=267.9  
## violator ~ male + race + state + multiple.offenses  
##   
## Df Deviance AIC  
## - male 1 254.07 266.07  
## <none> 253.90 267.90  
## - race 1 257.96 269.96  
## - multiple.offenses 1 266.90 278.90  
## - state 3 333.90 341.90  
##   
## Step: AIC=266.07  
## violator ~ race + state + multiple.offenses  
##   
## Df Deviance AIC  
## <none> 254.07 266.07  
## - race 1 258.03 268.03  
## - multiple.offenses 1 267.26 277.26  
## - state 3 334.69 340.69

summary(backmod)

##   
## Call:  
## glm(formula = violator ~ race + state + multiple.offenses, family = "binomial",   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.3667 -0.4317 -0.2643 -0.1829 2.8632   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.4626 0.3646 -6.754 1.43e-11  
## raceotherwise 0.7457 0.3726 2.001 0.045364  
## stateKentucky 0.1362 0.4638 0.294 0.769013  
## stateLouisiana 0.7838 0.4678 1.675 0.093848  
## stateVirginia -2.9875 0.6113 -4.887 1.03e-06  
## multiple.offensesmultiple offenses 1.3678 0.3830 3.571 0.000355  
##   
## (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: 340.04 on 472 degrees of freedom  
## Residual deviance: 254.07 on 467 degrees of freedom  
## AIC: 266.07  
##   
## Number of Fisher Scoring iterations: 6

Forward Stepwise

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

## Start: AIC=342.04  
## violator ~ 1  
##   
## Df Deviance AIC  
## + state 3 271.34 279.34  
## + multiple.offenses 1 336.26 340.26  
## + race 1 337.55 341.55  
## <none> 340.04 342.04  
## + male 1 339.37 343.37  
## + crime 3 336.46 344.46  
##   
## Step: AIC=279.34  
## violator ~ state  
##   
## Df Deviance AIC  
## + multiple.offenses 1 258.03 268.03  
## + race 1 267.26 277.26  
## <none> 271.34 279.34  
## + male 1 271.15 281.15  
## + crime 3 268.27 282.27  
##   
## Step: AIC=268.03  
## violator ~ state + multiple.offenses  
##   
## Df Deviance AIC  
## + race 1 254.07 266.07  
## <none> 258.03 268.03  
## + male 1 257.96 269.96  
## + crime 3 256.37 272.37  
##   
## Step: AIC=266.07  
## violator ~ state + multiple.offenses + race  
##   
## Df Deviance AIC  
## <none> 254.07 266.07  
## + male 1 253.90 267.90  
## + crime 3 252.49 270.49

summary(forwardmod)

##   
## Call:  
## glm(formula = violator ~ state + multiple.offenses + race, family = "binomial",   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.3667 -0.4317 -0.2643 -0.1829 2.8632   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.4626 0.3646 -6.754 1.43e-11  
## stateKentucky 0.1362 0.4638 0.294 0.769013  
## stateLouisiana 0.7838 0.4678 1.675 0.093848  
## stateVirginia -2.9875 0.6113 -4.887 1.03e-06  
## multiple.offensesmultiple offenses 1.3678 0.3830 3.571 0.000355  
## raceotherwise 0.7457 0.3726 2.001 0.045364  
##   
## (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: 254.07 on 467 degrees of freedom  
## AIC: 266.07  
##   
## Number of Fisher Scoring iterations: 6

When comparing the AIC of each model, it seems that the backward and forward stepwise model provides the best predictor of whether or not an inmate will violate their parole with an AIC of 266.07. In both models it shows that multiple.offenses as well as the state of the inmate are significant. This model does seem intuitive because it makes sense that a multiple offender will be more likely to violate their parole by committing another crime.

Task 5 Logistic Regression Model w/multiple variables

mod2 = glm(violator ~ multiple.offenses + state + race , parole, family = "binomial")  
summary(mod2)

##   
## 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

The quality of this model isn’t quite as good as either of the stepwise models when comparing the AIC numbers. It does show, like the stepwise models that multiple offenses along with the states of Louisiana and Kentucky and race do have significant impacts on violator.

Task 6 Predictions on sample parolees Parolee 1

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

## 1   
## 0.4228788

Parolee 1 would have a 42.28% chance of violating parole.

Parolee 2

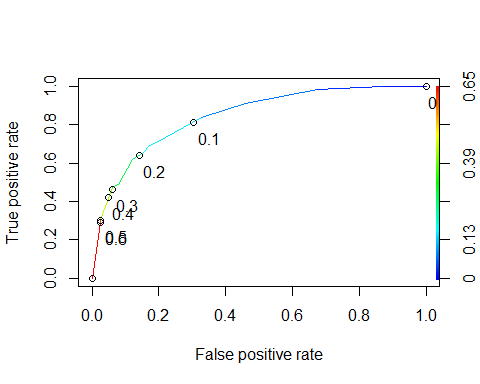
newdata = data.frame(state = "Kentucky", multiple.offenses = "otherwise", race = "otherwise")  
predict(forwardmod, newdata, type="response")

## 1   
## 0.1707006

Parolee 2 would have a 17.07% chance of violating parole.

Task 7 ROC curve

predictions = predict(forwardmod, 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

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

## [1] 0.8433667

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 0.7090909  
## specificity 0.8062201  
## cutoff 0.1522745

Confusion Matrix

t1 = table(train$violator,predictions > 0.1522745)  
t1

##   
## FALSE TRUE  
## not violated parole 346 72  
## violated parole 17 38

Calculate accuracy

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

## [1] 0.8118393

The accuracy of the forward model using the train dataset is 81.18% The sensitivity is .7090909 The specificity is .8062201 The implications of incorrectly classifying a parolee is that you could grant parole to a person who is likely to violate their parole and thus put an innocent person in danger of losing their property, getting injured, or worse.

You could also not grant parole to a person who wouldn’t violate their parole and thus unjustly incarerate them longer than they should be.

Task 9 Using trail and error with a threshold of .5

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

##   
## FALSE TRUE  
## not violated parole 407 11  
## violated parole 39 16

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

## [1] 0.8942918

Threshold of .6

t1 = table(train$violator,predictions > 0.6)  
t1

##   
## FALSE TRUE  
## not violated parole 407 11  
## violated parole 39 16

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

## [1] 0.8942918

Seems the maxium accuracy for the train set is as threshold is > than .5 and a .5 threshold and .6 threshold show the exact same accuracy.

Task 10 Accuracy of model on test data set

allmod1 = glm(violator ~ male + race + state + crime + multiple.offenses, test, family = "binomial")   
predictions1 = predict(allmod1, type="response")  
t1 = table(test$violator,predictions1 > 0.6)  
t1

##   
## FALSE TRUE  
## not violated parole 176 3  
## violated parole 16 7

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

## [1] 0.9059406

The accuracy of the model on the test set is 90.6%