BIOS-611-HW4

Matt Johnson

10/11/2020

Load in data

```
DF <- read.csv("500_Person_Gender_Height_Weight_Index.csv")
DF$Gender = ifelse(DF$Gender == "Male", 1, 0) #Male is 1, female is 0

set.seed <- 18 #this is my lucky number
spec = c(train = .6, test = .2, validate = .2)
DF1 = sample(cut(
  seq(nrow(DF)),
  nrow(DF)*cumsum(c(0,spec)),
  labels = names(spec)
))

DF.Split = split(DF, DF1)
```

Q1

```
glm1 <- glm(Gender ~ Height + Weight, data = DF.Split$train, family = "binomial")
glm2 <- step(glm1, trace = 0)
summary(glm1)
```

```
##
## Call:
## glm(formula = Gender ~ Height + Weight, family = "binomial",
##     data = DF.Split$train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.190  -1.129  -1.075   1.219   1.300
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.7601573  1.3109262   0.580    0.562
## Height      -0.0052997  0.0072988  -0.726    0.468
## Weight       0.0001137  0.0035539   0.032    0.974
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 414.81  on 299  degrees of freedom
## Residual deviance: 414.27  on 297  degrees of freedom
## AIC: 420.27
##
## Number of Fisher Scoring iterations: 3
```

```
summary(glm2)
```

```
##
## Call:
## glm(formula = Gender ~ 1, family = "binomial", data = DF.Split$train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.127  -1.127  -1.127   1.229   1.229
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.1201     0.1157  -1.039    0.299
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 414.81  on 299  degrees of freedom
## Residual deviance: 414.81  on 299  degrees of freedom
## AIC: 416.81
##
## Number of Fisher Scoring iterations: 3
```

```
anova(glm2, glm1)
```

```
## Analysis of Deviance Table
##
## Model 1: Gender ~ 1
## Model 2: Gender ~ Height + Weight
##   Resid. Df Resid. Dev Df Deviance
## 1       299     414.81
## 2       297     414.27  2   0.5339
```

```
DF.Split$test$glm1.probs <- predict(glm1, newdata = DF.Split$test, type = "response")
DF.Split$test <- DF.Split$test %>%
  mutate(glm1_pred = 1*(glm1.probs > .5) + 0) %>%
  mutate(accurate.glm1 = 1*(glm1_pred == Gender))
sum(DF.Split$test$accurate.glm1)/nrow(DF.Split$test)
```

```
## [1] 0.48
```

```
DF.Split$validate$glm1.probs <- predict(glm1, newdata = DF.Split$validate, type = "response")
DF.Split$validate <- DF.Split$validate %>%
  mutate(glm1_pred = 1*(glm1.probs > .5) + 0) %>%
  mutate(accurate.glm1 = 1*(glm1_pred == Gender))
sum(DF.Split$validate$accurate.glm1)/nrow(DF.Split$validate)
```

```
## [1] 0.53
```

The Accuracy of the GLM Model with Height and Weight as Predictors is about .5 on the validation set.

Q2

```
library(gbm)
```

```
## Loaded gbm 2.1.8
```

```
gbm1 <- gbm(Gender ~ Height + Weight, data = DF.Split$train, distribution = "bernoulli")

DF.Split$test$gbm1.probs <- predict(gbm1, newdata = DF.Split$test, type = "response")
```

```
## Using 100 trees...
```

```
DF.Split$test <- DF.Split$test %>%
  mutate(gbm1_pred = 1*(gbm1.probs > .5) + 0) %>%
  mutate(accurate.gbm1 = 1*(gbm1_pred == Gender))
sum(DF.Split$test$accurate.gbm1)/nrow(DF.Split$test)
```

```
## [1] 0.49
```

```
DF.Split$validate$gbm1.probs <- predict(gbm1, newdata = DF.Split$validate, type = "response")
```

```
## Using 100 trees...
```

```
DF.Split$validate <- DF.Split$validate %>%
  mutate(gbm1_pred = 1*(gbm1.probs > .5) + 0) %>%
  mutate(accurate.gbm1 = 1*(gbm1_pred == Gender))
sum(DF.Split$validate$accurate.gbm1)/nrow(DF.Split$validate)
```

```
## [1] 0.47
```

The Accuracy of the GBM Model with Height and Weight as Predictors is .5 on the validation set.

Q3

```
#remove all except 50 males in the original dataset.
DF.50.Males <- DF %>%
  filter(Gender == 1) %>%
  tail(50)

DF.Q3 <- anti_join(DF, DF.50.Males)
```

```
## Joining, by = c("Gender", "Height", "Weight", "Index")
```

```
set.seed <- 18 #this is my lucky number
spec = c(train = .6, test = .2, validate = .2)
DF1 = sample(cut(
  seq(nrow(DF.Q3)),
  nrow(DF.Q3)*cumsum(c(0,spec)),
  labels = names(spec)
))

DF.Split.Q3 = split(DF.Q3, DF1)


glm2 <- glm(Gender ~ Height + Weight, data = DF.Split.Q3$train, family = "binomial")

DF.Split.Q3$test$glm2.probs <- predict(glm2, newdata = DF.Split.Q3$test, type = "response")
DF.Split.Q3$test<- DF.Split.Q3$test %>%
  mutate(glm2_pred = 1*(glm2.probs > .5) + 0) %>%
  mutate(accurate.glm2 = 1*(glm2_pred == Gender))
sum(DF.Split.Q3$test$accurate.glm2)/nrow(DF.Split.Q3$test)
```

```
## [1] 0.6333333
```

```
DF.Split.Q3$validate$glm2.probs <- predict(glm2, newdata = DF.Split.Q3$validate, type = "response")
DF.Split.Q3$validate<- DF.Split.Q3$validate %>%
  mutate(glm2_pred = 1*(glm2.probs > .5) + 0) %>%
  mutate(accurate.glm2 = 1*(glm2_pred == Gender))
sum(DF.Split.Q3$validate$accurate.glm2)/nrow(DF.Split.Q3$validate)
```

```
## [1] 0.5222222
```

```
library(MLmetrics)
```

```
##
## Attaching package: 'MLmetrics'
```

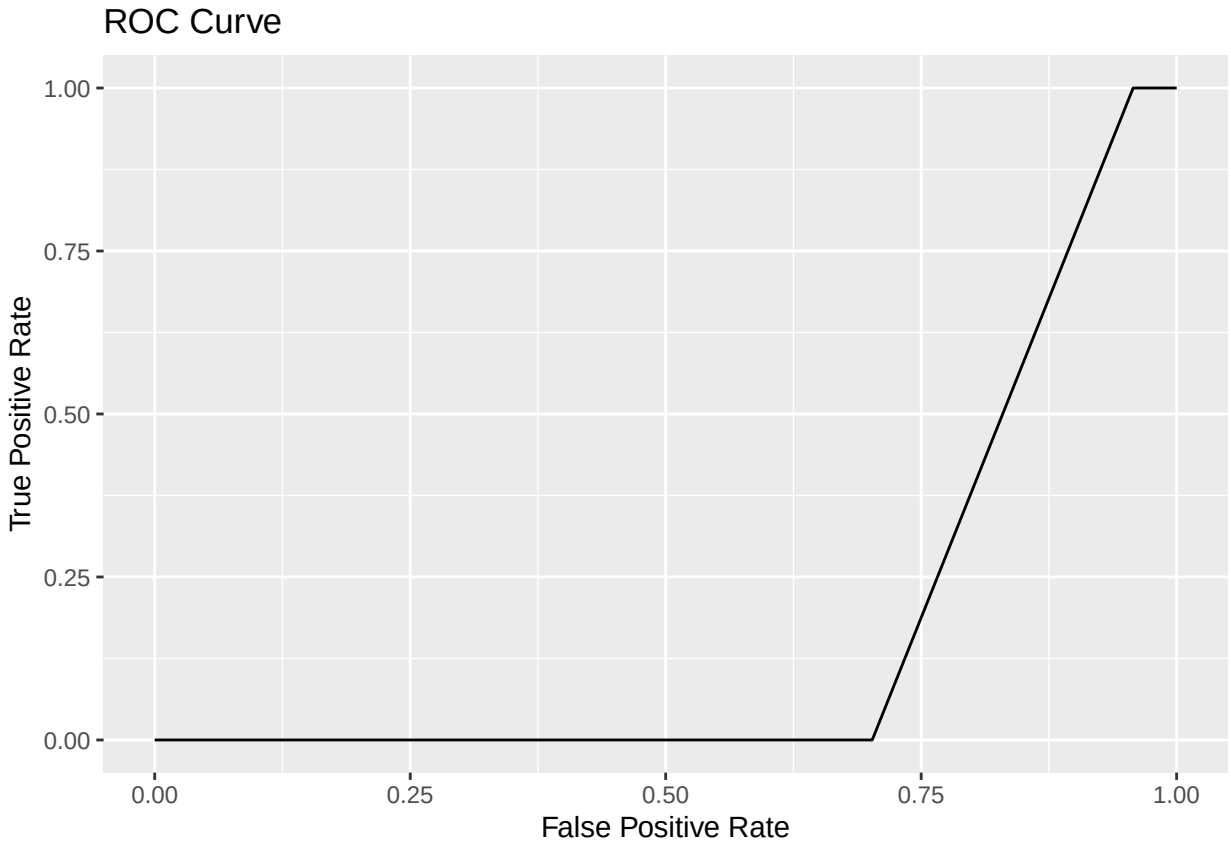
```
## The following object is masked from 'package:base':
##
##     Recall
```

```
f1 <- MLmetrics::F1_Score
DF.Split.Q3$validate$glm2_pred[10] <- 1
f1(DF.Split.Q3$validate$Gender, DF.Split.Q3$validate$glm2_pred) #there is an issue here with the model
```

```
## [1] 0.6911765
```

Q4

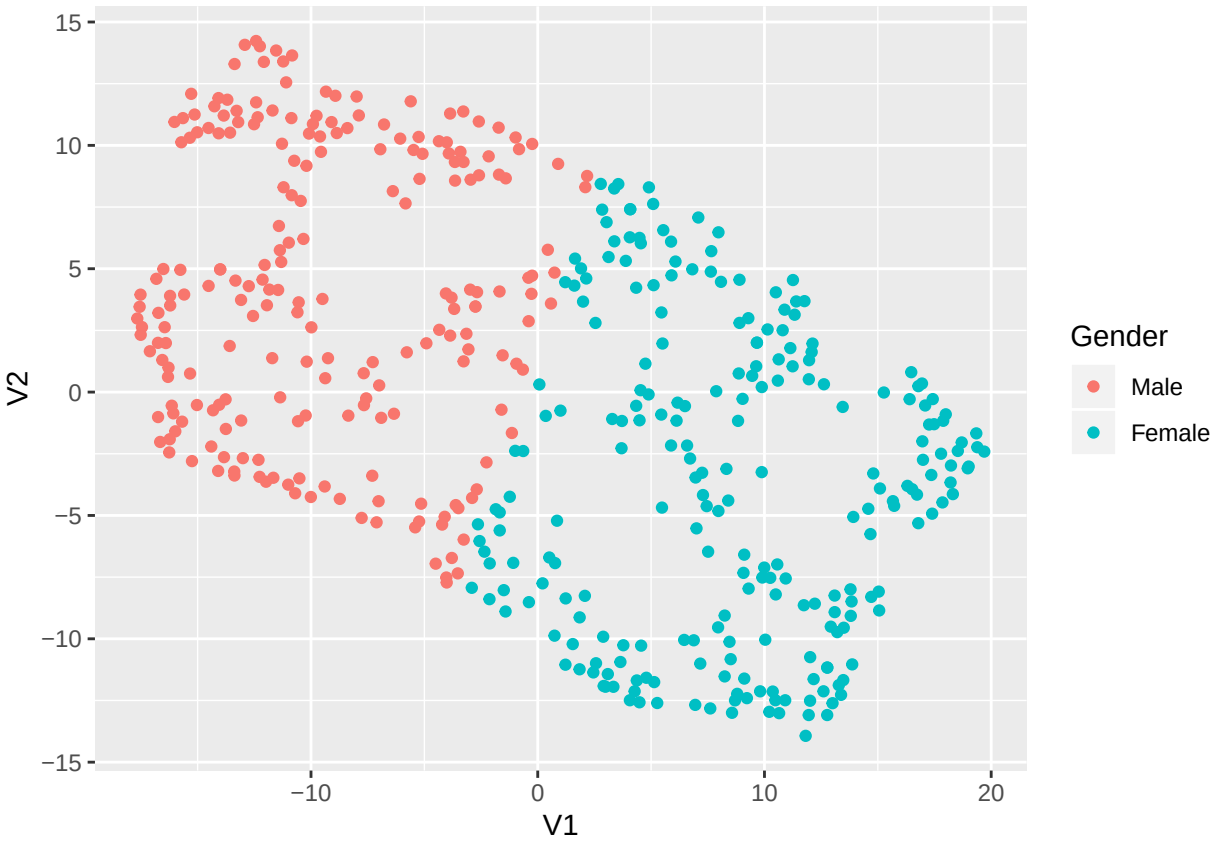
```
roc <- do.call(rbind, Map(function(threshold){
    p <- DF.Split.Q3$validate$glm2.probs > threshold;
    tp <- sum(p[DF.Split.Q3$validate$Gender])/sum(DF.Split.Q3$validate$Gender);
    fp <- sum(p[!DF.Split.Q3$validate$Gender])/sum(!DF.Split.Q3$validate$Gender);
    tibble(threshold=threshold,
           tp=tp,
           fp=fp)
},seq(100)/100))

ggplot(roc, aes(fp,tp)) + geom_line() + xlim(0,1) + ylim(0,1) +
    labs(title="ROC Curve",x="False Positive Rate",y="True Positive Rate")
```



Since we are assesing the test looking at the area under the roc curve, we can say that this is not a good model.

Q5

```
library(Rtsne)
cc <- kmeans(DF.Q3 %>% select(Height, Weight), 2)
fit1 <- Rtsne(DF.Q3 %>% select(Height, Weight), dims=2, check_duplicates = F)
g2 <- ggplot(fit1$Y %>% as.data.frame() %>% as_tibble() %>% mutate(label=cc$cluster),aes(V1,V2)) +
  geom_point(aes(color=factor(label))) +
  scale_color_discrete(name="Gender", labels = c("Male", "Female"))
g2
```



It is difficult to say with great confidence that there are, in fact, two distinct groups. More clutering methods, like Principle Component Analysis, should be considered to further asses the cluster.