Exam 1 Q1

Trey Davidson

2024-10-11

### A.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
  
set.seed(10)  
  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
n <- nrow(data)  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
test\_data <- data[-train\_indices, ]  
  
  
# Creates logsitc model using gender as y and all other varibales as predictors  
model <- glm(gender ~ ., data = train\_data, family = binomial)  
  
# Summarizes model  
summary(model)

##   
## Call:  
## glm(formula = gender ~ ., family = binomial, data = train\_data)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 12.0657 6.2822 1.921 0.0548 .  
## inconsistencies -0.4608 0.3205 -1.438 0.1505   
## paper 0.1006 0.1872 0.538 0.5909   
## tool -1.0969 0.5689 -1.928 0.0538 .  
## vocab 0.7511 0.4374 1.717 0.0859 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 44.236 on 31 degrees of freedom  
## Residual deviance: 11.641 on 27 degrees of freedom  
## AIC: 21.641  
##   
## Number of Fisher Scoring iterations: 8

# Predicts probabilities of women with test data  
predicted\_probs <- predict(model, newdata = test\_data, type = "response")  
  
# If probablituy of woman > 0.5 then assumes its a women and if its less then assumes its a male  
predicted\_classes <- ifelse(predicted\_probs > 0.5, 1, 0)  
  
  
# Creates a confusion matrix   
confusion\_matrix <- table(Predicted = predicted\_classes, Actual = test\_data$gender)  
  
  
confusion\_matrix

## Actual  
## Predicted 0 1  
## 0 14 2  
## 1 1 15

# Calclues the confusion matrix accuracy by callcuting the sum of acutal correct predicitons over the total predictoons  
accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
  
1 - accuracy

## [1] 0.09375

aprox 9% error rate is quite low and means that this logstic model is clealry able to caputre some type of realtionship between preidcotr and output varibales.

### B.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
set.seed(10)  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
data$gender <- as.factor(data$gender)  
  
  
print(data$gender)

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1  
## [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## Levels: 0 1

n <- nrow(data)  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
valid\_data <- data[-train\_indices, ]  
  
  
library(MASS)  
  
  
lda\_model <- lda(gender ~ ., data = train\_data)  
  
  
lda\_predictions <- predict(lda\_model, valid\_data)$class  
  
  
confusion\_matrix <- table(lda\_predictions, valid\_data$gender)  
  
accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
  
1- accuracy

## [1] 0.125

aprox 12% Error rate is worse then logsitc, but still means linnear discirmnate anayslis is able to caputre some type of realtionship. I predcit that qudratic will have a lower test error.

### C.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
set.seed(10)  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
data$gender <- as.factor(data$gender)  
  
  
print(data$gender)

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1  
## [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## Levels: 0 1

n <- nrow(data)  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
valid\_data <- data[-train\_indices, ]  
  
  
library(MASS)  
  
  
qda\_model <- qda(gender ~ ., data = train\_data)  
  
  
qda\_predictions <- predict(qda\_model, valid\_data)$class  
  
  
confusion\_matrix <- table(Predicted = qda\_predictions, Actual = valid\_data$gender)  
  
  
print(confusion\_matrix)

## Actual  
## Predicted 0 1  
## 0 13 3  
## 1 2 14

accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
1 - accuracy

## [1] 0.15625

Supprsingly to me qda has a higher test error of 15.625%. This must mean that covariances are more similair then I thought.

### D.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
set.seed(10)  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
data$gender <- as.factor(data$gender)  
  
  
print(data$gender)

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1  
## [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## Levels: 0 1

n <- nrow(data)  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
valid\_data <- data[-train\_indices, ]  
  
  
library(e1071)  
  
  
nb\_model <- naiveBayes(gender ~ ., data = train\_data)  
  
summary(nb\_model)

## Length Class Mode   
## apriori 2 table numeric   
## tables 4 -none- list   
## levels 2 -none- character  
## isnumeric 4 -none- logical   
## call 4 -none- call

nb\_predictions <- predict(nb\_model, valid\_data)  
  
  
confusion\_matrix <- table(Predicted = nb\_predictions, Actual = valid\_data$gender)  
  
  
print(confusion\_matrix)

## Actual  
## Predicted 0 1  
## 0 14 4  
## 1 1 13

accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
1 - accuracy

## [1] 0.15625

This one had a test error of 15.625%. THis to me may suggest that the data is not indepdentent, and is better suited for a da.

### E.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
set.seed(10)  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
data$gender <- as.factor(data$gender)  
  
  
print(data$gender)

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1  
## [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## Levels: 0 1

n <- nrow(data)  
  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
test\_data <- data[-train\_indices, ]  
  
  
train\_features <- train\_data[, -1]   
test\_features <- test\_data[, -1]   
train\_labels <- train\_data$gender  
test\_labels <- test\_data$gender  
  
  
  
library(class)  
  
k <- 4  
  
knn\_predictions <- knn(train = train\_features, test = test\_features, cl = train\_labels, k = k)  
  
confusion\_matrix <- table(Predicted = knn\_predictions, Actual = test\_labels)  
  
print(confusion\_matrix)

## Actual  
## Predicted 0 1  
## 0 14 1  
## 1 1 16

accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
  
error\_rate <- 1 - accuracy  
print(error\_rate)

## [1] 0.0625

Knn performed the best with 6.25% accuracy.

### F.

data <- read.table("~/Desktop/HW/EXAM1/T5\_1\_PSYCH.DAT", header = FALSE)  
  
  
colnames(data)[colnames(data) == "V1"] <- "gender"  
colnames(data)[colnames(data) == "V2"] <- "inconsistencies"  
colnames(data)[colnames(data) == "V3"] <- "paper"  
colnames(data)[colnames(data) == "V4"] <- "tool"  
colnames(data)[colnames(data) == "V5"] <- "vocab"  
  
  
set.seed(10)  
  
  
data$gender <- ifelse(data$gender == 1, 0, 1)  
  
  
data$gender <- as.factor(data$gender)  
  
  
print(data$gender)

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1  
## [39] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## Levels: 0 1

n <- nrow(data)  
  
train\_indices <- sample(1:n, size = 0.5 \* n)  
train\_data <- data[train\_indices, ]  
test\_data <- data[-train\_indices, ]  
  
  
train\_features <- train\_data[, -1]   
test\_features <- test\_data[, -1]   
train\_labels <- train\_data$gender  
test\_labels <- test\_data$gender  
  
  
  
library(class)  
  
k <- 4  
  
knn\_predictions <- knn(train = train\_features, test = data[, -1], cl = train\_labels, k = k)  
  
confusion\_matrix <- table(Predicted = knn\_predictions, Actual = data$gender)  
  
print(confusion\_matrix)

## Actual  
## Predicted 0 1  
## 0 27 0  
## 1 5 32

accuracy <- sum(diag(confusion\_matrix)) / sum(confusion\_matrix)  
  
  
error\_rate <- 1 - accuracy  
  
error\_rate

## [1] 0.078125

I ran knn with all the data and got 7.8125%, while only traing it with 50%, and testing on 100%