Multivariate Analysis
Homework #9

Aidan Dykstal
April 8th, 2020

Preliminaries

First, we load the cereal data:

```
# Load the Cereal Data
Cereal <- read.csv('cereal.csv')

# Remove the Names of the Cereals
CerealNoNames <- Cereal[-1]
```

Now, we are ready to proceed.

Problem 1

We will use logistic regression to classify each cereal as to manufacturer.

```
# Fit the Model
fit <- multinom(Manufacturer ~ .,
                data = CerealNoNames,
                trace = FALSE,
                maxit = 10000)
coef <- summary(fit)$coefficients

# Use Coefficients to Find the Log Odds
nCereal <- t(CerealNoNames[-1])
logOdds <- coef[,1] + coef[,-1] %*% nCereal
logOdds <- cbind(0, t(logOdds))
classHat <- apply(logOdds, 1, which.max)

# Print the Classes Predicted
classHat
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 2 2 1 2 1 2 2 2 2 1 2 2 2 2 2 1 2 1 2 2 2 1 1 2 2 3
## [39] 3 3 3 3 3
```

Note that above, class 1 is General Mills (G), class 2 is Kellog's (K), and class 3 is Quaker (Q). Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
```

```
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
## [1,]   14    3    0
## [2,]    5   15    0
## [3,]    0    0    6
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above. We will now estimate the misclassification rate:

```
# Misclassification Rate from Confusion Matrix
classRate <- sum(diag(C)) / sum(C)
misclassRate <- 1.0 - classRate
cat("Misclassification Rate Estimate:", misclassRate)
```

```
## Misclassification Rate Estimate: 0.1860465
```

The estimated misclassification rate is given above.

Problem 2

We will use logistic regression to classify each cereal as to manufacturer.

```
# Fit the Model
n <- dim(Cereal)[1]
cvClassHat <- rep(0, n)
for (i in 1:n) {
  fit <- multinom(Manufacturer ~ .,
                  data = CerealNoNames[-i,],
                  trace = FALSE,
                  maxit = 10000)
  coef <- summary(fit)$coefficients

  # Use Coefficients to Find the Log Odds
  nCereal <- t(CerealNoNames[-1][i,])
  logOdds <- coef[,1] + coef[,-1] %*% nCereal
  logOdds <- cbind(0, t(logOdds))
  cvClassHat[i] <- which.max(logOdds)
}

# Print the Classes Predicted
classHat <- cvClassHat
classHat
```

```
##  [1] 3 3 1 1 1 2 1 1 1 1 1 3 2 2 1 2 1 1 2 2 2 1 2 2 2 2 2 1 2 1 2 2 1 1 1 2 3 1
## [39] 2 1 3 3 3
```

Note that above, class 1 is General Mills (G), class 2 is Kellog's (K), and class 3 is Quaker (Q). Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
```

```
## [1,]   10    4    3
## [2,]    7   12    1
## [3,]    2    1    3
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above. We will now estimate the misclassification rate:

```
# Misclassification Rate from Confusion Matrix
classRate <- sum(diag(C)) / sum(C)
misclassRate <- 1.0 - classRate
cat("Misclassification Rate Estimate:", misclassRate)
```

```
## Misclassification Rate Estimate: 0.4186047
```

The estimated misclassification rate is given above.

Problem 3

We will use k-nearest neighbors to classify each cereal as to manufacturer.

```
# Fit the Model
fit <- knn(CerealNoNames[-1],
           CerealNoNames[-1],
           cl = Cereal[, 'Manufacturer'],
           k = 1)
fit
```

```
##  [1]  G  G  G  G  G  G  G  G  G  G  G  G  G  G  G  G  G  K  K  K  K  K  K  K  K
## [26]  K  K  K  K  K  K  K  K  K  K  K  Q  Q  Q  Q  Q  Q
## Levels:  G  K  Q
```

Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
classHat <- rep(0, n)
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}
for (i in 1:n) {
  if (fit[i] == G) {
    classHat[i] <- 1
  } else if (fit[i] == K) {
    classHat[i] <- 2
  } else if (fit[i] == Q) {
    classHat[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
## [1,]   17    0    0
## [2,]    0   20    0
## [3,]    0    0    6
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above.

We notice that the confusion matrix has only zero off the main diagonal. This implies that the KNN approach with $k = 1$ classified our test set perfectly. This occurs because – with our test and train sets being the same – the one nearest neighbor to each point in the test set was itself (after all, that point is also identically in the training set). Thus, KNN just extracted the class of the point in the test set from its copy in the train set, so every single point was classified correctly.

Problem 4

Part A

We will use k-nearest neighbors to classify each cereal as to manufacturer. We use LOOCV with $k = 1$, $k = 3$, and $k = 7$. We begin with $k = 1$.

$k = 1$:

```
# Fit the Model
n <- dim(Cereal)[1]
fitCV <- rep(0, n)
for (i in 1:n) {
  fit <- knn(CerealNoNames[-1][-i,],
             CerealNoNames[-1][i,],
             cl = Cereal[-i, 'Manufacturer'],
             k = 1)
  fitCV[i] <- fit
}
classHat <- fitCV
classHat
```

```
##  [1] 1 1 1 1 2 1 1 1 1 3 2 2 2 1 2 1 1 2 2 2 2 1 3 2 1 3 2 1 2 1 1 3 2 1 2 2 3 3
## [39] 3 2 3 3 2
```

Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
## [1,]   11    5    1
## [2,]    6   10    4
## [3,]    0    2    4
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above.

```r
# Misclassification Rate from Confusion Matrix
classRate <- sum(diag(C)) / sum(C)
misclassRate <- 1.0 - classRate
cat("Misclassification Rate Estimate:", misclassRate)
```

```
## Misclassification Rate Estimate: 0.4186047
```

The estimated misclassification rate is given above.

$k = 3$:

```
# Fit the Model
n <- dim(Cereal)[1]
fitCV <- rep(0, n)
for (i in 1:n) {
  fit <- knn(CerealNoNames[-1][-i,],
             CerealNoNames[-1][i,],
             cl = Cereal[-i, 'Manufacturer'],
             k = 3)
  fitCV[i] <- fit
}
classHat <- fitCV
classHat
```

```
##  [1] 1 1 1 1 2 1 2 1 1 1 1 2 2 1 2 1 1 2 2 2 2 1 3 2 2 3 2 1 1 1 1 1 2 2 1 2 3 2
## [39] 2 2 2 3 2
```

Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
## [1,]   12    5    0
## [2,]    7   10    3
## [3,]    0    5    1
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above.

```
# Misclassification Rate from Confusion Matrix
classRate <- sum(diag(C)) / sum(C)
misclassRate <- 1.0 - classRate
cat("Misclassification Rate Estimate:", misclassRate)
```

```
## Misclassification Rate Estimate: 0.4651163
```

The estimated misclassification rate is given above.

$k = 7$:

```
# Fit the Model
n <- dim(Cereal)[1]
fitCV <- rep(0, n)
for (i in 1:n) {
  fit <- knn(CerealNoNames[-1][-i,],
             CerealNoNames[-1][i,],
             cl = Cereal[-i, 'Manufacturer'],
             k = 7)
  fitCV[i] <- fit
}
classHat <- fitCV
classHat
```

```
##  [1] 1 1 1 1 2 1 2 1 1 1 2 2 2 1 2 1 1 2 2 2 2 1 2 2 1 2 1 1 1 1 1 1 1 2 2 2 1 2
## [39] 2 1 2 2 2
```

Now, we build the confusion matrix:

```
# Find the True Classes
n <- dim(Cereal)[1]
trueClasses <- rep(0, n)
G <- Cereal[, 'Manufacturer'][1]
K <- Cereal[, 'Manufacturer'][20]
Q <- Cereal[, 'Manufacturer'][43]
for (i in 1:n) {
  if (Cereal[, 'Manufacturer'][i] == G) {
    trueClasses[i] <- 1
  } else if (Cereal[, 'Manufacturer'][i] == K) {
    trueClasses[i] <- 2
  } else if (Cereal[, 'Manufacturer'][i] == Q) {
    trueClasses[i] <- 3
  }
}

# Create the Confusion Matrix
C <- matrix(data = c(0,0,0,0,0,0,0,0,0),
            nrow = 3, ncol = 3)
for (i in 1:n) {
  C[trueClasses[i], classHat[i]] = C[trueClasses[i], classHat[i]] + 1
}
C
```

```
##      [,1] [,2] [,3]
## [1,]   11    6    0
## [2,]   10   10    0
## [3,]    1    5    0
```

The confusion matrix – with actual classes on the rows and predicted values on the columns – is given above.

```
# Misclassification Rate from Confusion Matrix
classRate <- sum(diag(C)) / sum(C)
misclassRate <- 1.0 - classRate
cat("Misclassification Rate Estimate:", misclassRate)
```

```
## Misclassification Rate Estimate: 0.5116279
```

The estimated misclassification rate is given above.

Part B

In our data, there are only six observations of the class "Q". There are 37 observations of the other classes. Thus, as k increases, especially beyond six, the k-nearest neighbors to a given point in the test set are more likely to be a class other than "Q"; in fact, as k grows beyond six, there must be at least one point in the k-nearest neighbors that is *not* "Q". What this means is that – even if the covariate values are close to those of type "Q" – the small number of "Q" observations means that the k-nearest neighbors approach will still pick up many more observations whose classes have counts outnumbering "Q". Class "Q" becomes much less likely to typify the majority of the k-nearest neighbors as k grows due to its relatively small representation in our sample of cereals.