Data Analysis Assignment

Elijah Verdoorn

September 25, 2016

Classification

Setup

```
maxKVal <- 100
means <- matrix(nrow = maxKVal, ncol = 1)
```

Build a matrix in which to store the means that get calculated, as well as the limiter of the `for` loop.

Main Loop

```
for (i in 1:maxKVal) {
  kval <- i # number of neighbors

  train.df <- read.csv("adult.csv") # read in the training data
  test.df <- read.csv("adult_test.csv") # read in the testing data

  response.predict <- knn(train.df[,c("Age", "race", "sex")],
                          test.df[,c("Age", "race", "sex")],
                          train.df[,c("X..50k")], k = kval) # build the KNN model
  response.predict # print the testing results

  test.df <- mutate(test.df, class.pred = response.predict)
  # how did we do? calculate the error rate
  means[i] <- mean(test.df[,c("X1")] != test.df["class.pred"])
}
```

The main loop, which builds `maxKVal` models and calculates the error for those models.

Analysis

```
min(means) # at k = 78
```

```
## [1] 0.2404029
```

Find the minimum error, in this case that occurs around `k = 40`.

Regression

Setup

```
# get the dataset
data <- read.csv("winequality-white.csv")
degrees <- 20 # the number of degrees to test
results.df <- data.frame("degree" = integer(), "train" = double(), "test" = double())
```

Main Loop

```
for(i in 1:degrees) {
  n <- nrow(data) # get the number of rows
  train <- sample(1:n, n/2, rep = F) # get subset of the data for training
  test <- setdiff(1:n, train) # and one for testing

  train.df <- data[train, c("pH", "alcohol")] # set up the data frame for training
  test.df <- data[test, c("pH", "alcohol")] # set up the data frame for testing

  mod <- lm(pH~poly(alcohol, i), data = train.df) # build the linear model with degree i
  pred.train <- predict(mod, newdata = train.df) # traing the model
  train.df <- mutate(train.df, pred = pred.train) # add the training data to the data frame

  pred.test <- predict(mod, newdata = test.df) # test the model
  test.df <- mutate(test.df, pred = pred.test) # add the test data to the data frame

  mse.train <- with(train.df, mean((pH - pred)^2)) # calculate information about the model
  mse.test <- with(test.df, mean((pH - pred)^2))
  c(mse.train, mse.test) # print the information about the model to the console
  results.df <- rbind(results.df, c(i, mse.train, mse.test))
}
```

Calculate the models, then add information about that degree to the data frame.

Results

```
head(results.df)
```

```
##   X1 X0.0221162734705131 X0.0228162739161177
## 1  1          0.02211627          0.02281627
## 2  2          0.02255372          0.02163271
## 3  3          0.02168366          0.02194662
## 4  4          0.02195079          0.02165768
## 5  5          0.02164993          0.02171383
## 6  6          0.02191057          0.02126057
```