KNN

Elijah Verdoorn

September 24, 2016

Setup

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

Setup the number of k values that we'll use, and create a matrix of means that we can use to analyze the number of k valuse that'll work best.

Main Loop

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

  colleges.df <- read.csv("collegeScoreboard2015Scaled2.csv") # read in the training data
  test.df <- read.csv("collegeScoreboard2015Scaled1.csv") # read in the testing data

  response.predict <- knn(colleges.df[2:4],
                          test.df[2:4],
                          colleges.df[,c("Type")],
                          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["Type"] != test.df["class.pred"])
}
```

Main loop. Create a bunch of KNN models, calculate the error rate for each of the models, then store that value in a matrix for later analysis.

Optimal K Value

```
min(means) # at k = 38
```

```
## [1] 0.1366906
```

Get the minimum error in the matrix, then look for that value in the matrix. Turns out that k=38 is the best for our purposes.