HW8

Lacey Gleason

4/14/2018

K-nearest neighbor

Let's try a variation on the NHANES data set again.

```
library(tidyverse)
library(class)
library(rpart)
library(NHANES)
library(RColorBrewer)
library(plot3D)
library(parallel)
library(randomForestSRC)
library(ggRandomForests)
library(mosaic)

# Create the NHANES dataset again

people <- NHANES %>% dplyr::select(Age, Gender, SleepTrouble, BMI, HHIncome, 
PhysActive)
#%>% na.omit()

glimpse(people)

## Observations: 10,000
## Variables: 6
## $ Age          <int> 34, 34, 34, 4, 49, 9, 8, 45, 45, 45, 66, 58, 54, ...
## $ Gender       <fct> male, male, male, male, female, male, male, femal...
## $ SleepTrouble <fct> Yes, Yes, Yes, NA, Yes, NA, NA, No, No, No, No, N...
## $ BMI          <dbl> 32.22, 32.22, 32.22, 15.30, 30.57, 16.82, 20.64, ...
## $ HHIncome     <fct> 25000-34999, 25000-34999, 25000-34999, 20000-2499...
## $ PhysActive   <fct> No, No, No, NA, No, NA, NA, Yes, Yes, Yes, Yes, Y...
```

Create the NHANES dataset again, just like we did in class, only using sleep trouble (variable name = `SleepTrouble`) as the dependent variable, instead of `Diabetes`.

Problem 1

What is the marginal distribution of sleep trouble (`SleepTrouble`)?

```
#What is the marginal distribution of sleep trouble in the NHANES dataset?

tally(~ SleepTrouble, data = people, format = "percent")
```

```
## SleepTrouble
##    No   Yes  <NA>
## 57.99 19.73 22.28
```

The marginal distribution of sleep trouble in the NHANES dataset is 19.73%.

Recall from our prior work, the packages work better if the dataset is a dataframe, and the variables are numeric.

```
class(people)

## [1] "tbl_df"     "tbl"        "data.frame"

# Convert back to dataframe
people <- as.data.frame(people)
glimpse(people)

## Observations: 10,000
## Variables: 6
## $ Age          <int> 34, 34, 34, 4, 49, 9, 8, 45, 45, 45, 66, 58, 54, ...
## $ Gender       <fct> male, male, male, male, female, male, male, femal...
## $ SleepTrouble <fct> Yes, Yes, Yes, NA, Yes, NA, NA, No, No, No, No, N...
## $ BMI          <dbl> 32.22, 32.22, 32.22, 15.30, 30.57, 16.82, 20.64, ...
## $ HHIncome     <fct> 25000-34999, 25000-34999, 25000-34999, 20000-2499...
## $ PhysActive   <fct> No, No, No, NA, No, NA, NA, Yes, Yes, Yes, Yes, Y...

# Convert factors to numeric - the packages just seem to work better that way
people$Gender <- as.numeric(people$Gender)
people$SleepTrouble <- as.numeric(people$SleepTrouble)
people$HHIncome <- as.numeric(people$HHIncome)
people$PhysActive <- as.numeric(people$PhysActive)

# remove missing values
people <- na.omit(people)

glimpse(people)

## Observations: 7,037
## Variables: 6
## $ Age          <int> 34, 34, 34, 49, 45, 45, 45, 66, 58, 54, 58, 50, 3...
## $ Gender       <dbl> 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 1, 2, 2, 2, 1, 1, 2...
## $ SleepTrouble <dbl> 2, 2, 2, 2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 2...
## $ BMI          <dbl> 32.22, 32.22, 32.22, 30.57, 27.24, 27.24, 27.24, ...
## $ HHIncome     <dbl> 6, 6, 6, 7, 11, 11, 11, 6, 12, 10, 11, 4, 6, 4, 1...
## $ PhysActive   <dbl> 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 2, 2, 2...
```

Apply the k-nearest neighbor procedure to predict `SleepTrouble` from the other covariates, as we did for `Diabetes`. Use k = 1, 3, 5, and 20.

Problem 2

```
#Apply k-nearest neighbor approach to predict SleepTrouble for k = 1, 3, 5, 2
0

# Let's try different values of k to see how that affects performance
knn.1 <- knn(train = people, test = people, cl = people$SleepTrouble, k = 1)
knn.3 <- knn(train = people, test = people, cl = people$SleepTrouble, k = 3)
knn.5 <- knn(train = people, test = people, cl = people$SleepTrouble, k = 5)
knn.20 <- knn(train = people, test = people, cl = people$SleepTrouble, k = 20
)
```

Now let’s see how well these classifiers work overall

Problem 3

```
# How well do these classifiers (k = 1, 3, 5, 20) work?
  # Calculate the percent predicted correctly

100*sum(people$SleepTrouble == knn.1)/length(knn.1)

## [1] 100

100*sum(people$SleepTrouble == knn.3)/length(knn.3)

## [1] 92.04206

100*sum(people$SleepTrouble == knn.5)/length(knn.5)

## [1] 88.70257

100*sum(people$SleepTrouble == knn.20)/length(knn.20)

## [1] 78.74094
```

Problem 4

What about success overall?

```
#Insert your code here to determine overall success for k = 1, 3, 5, 20
table(knn.1, people$SleepTrouble)

##
## knn.1    1    2
##     1 5239    0
##     2    0 1798

table(knn.3, people$SleepTrouble)

##
## knn.3    1    2
##     1 5062  383
##     2  177 1415

table(knn.5, people$SleepTrouble)
```

```
##
## knn.5    1    2
##     1 5032  588
##     2  207 1210
```

```r
table(knn.20, people$SleepTrouble)
```

```
##
## knn.20    1    2
##      1 5090 1347
##      2  149  451
```

We see that as k increases, the prediction for sleep trouble worsens.

Link to GitHub repository

https://github.com/lpgleason/2018week11.git