k-NN Donor Case Study

This is an [R Markdown](http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.2 ✓ purrr 0.3.4  
## ✓ tibble 3.0.3 ✓ dplyr 1.0.2  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0  
## ✓ readr 1.3.1 ✓ forcats 0.5.0

## ── Conflicts ────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

donors <- read\_csv("data/donors.csv", col\_types = "nnnnnnnnnnnnffffffffff")  
  
glimpse(donors)

## Rows: 95,412  
## Columns: 22  
## $ age <dbl> 60, 46, NA, 70, 78, NA, 38, NA, NA, 65, NA, 7…  
## $ numberChildren <dbl> NA, 1, NA, NA, 1, NA, 1, NA, NA, NA, NA, NA, …  
## $ incomeRating <dbl> NA, 6, 3, 1, 3, NA, 4, 2, 3, NA, 2, 1, 4, NA,…  
## $ wealthRating <dbl> NA, 9, 1, 4, 2, NA, 6, 9, 2, NA, 0, 5, 2, NA,…  
## $ mailOrderPurchases <dbl> 0, 16, 2, 2, 60, 0, 0, 1, 0, 0, 0, 3, 16, 0, …  
## $ totalGivingAmount <dbl> 240, 47, 202, 109, 254, 51, 107, 31, 199, 28,…  
## $ numberGifts <dbl> 31, 3, 27, 16, 37, 4, 14, 5, 11, 3, 1, 2, 9, …  
## $ smallestGiftAmount <dbl> 5, 10, 2, 2, 3, 10, 3, 5, 10, 3, 20, 10, 4, 5…  
## $ largestGiftAmount <dbl> 12, 25, 16, 11, 15, 16, 12, 11, 22, 15, 20, 1…  
## $ averageGiftAmount <dbl> 7.741935, 15.666667, 7.481481, 6.812500, 6.86…  
## $ yearsSinceFirstDonation <dbl> 8, 3, 7, 10, 11, 3, 10, 3, 9, 3, 1, 1, 8, 5, …  
## $ monthsSinceLastDonation <dbl> 14, 14, 14, 14, 13, 20, 22, 18, 19, 22, 12, 1…  
## $ inHouseDonor <fct> FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALS…  
## $ plannedGivingDonor <fct> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FAL…  
## $ sweepstakesDonor <fct> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FAL…  
## $ P3Donor <fct> FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALS…  
## $ state <fct> IL, CA, NC, CA, FL, AL, IN, LA, IA, TN, KS, I…  
## $ urbanicity <fct> town, suburb, rural, rural, suburb, town, tow…  
## $ socioEconomicStatus <fct> average, highest, average, average, average, …  
## $ isHomeowner <fct> NA, TRUE, NA, NA, TRUE, NA, TRUE, NA, NA, NA,…  
## $ gender <fct> female, male, male, female, female, NA, femal…  
## $ respondedMailing <fct> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FAL…

Use only numeric features in dataset as predictors for the response

donors <- donors %>%  
 select(  
 age,  
 numberChildren,  
 incomeRating,  
 wealthRating,  
 mailOrderPurchases,  
 totalGivingAmount,  
 numberGifts,  
 smallestGiftAmount,  
 largestGiftAmount,  
 averageGiftAmount,  
 yearsSinceFirstDonation,  
 monthsSinceLastDonation,  
 respondedMailing  
 )

Dealing with missing data

summary(donors)

## age numberChildren incomeRating wealthRating   
## Min. : 1.00 Min. :1.00 Min. :1.000 Min. :0.00   
## 1st Qu.:48.00 1st Qu.:1.00 1st Qu.:2.000 1st Qu.:3.00   
## Median :62.00 Median :1.00 Median :4.000 Median :6.00   
## Mean :61.61 Mean :1.53 Mean :3.886 Mean :5.35   
## 3rd Qu.:75.00 3rd Qu.:2.00 3rd Qu.:5.000 3rd Qu.:8.00   
## Max. :98.00 Max. :7.00 Max. :7.000 Max. :9.00   
## NA's :23665 NA's :83026 NA's :21286 NA's :44732   
## mailOrderPurchases totalGivingAmount numberGifts smallestGiftAmount  
## Min. : 0.000 Min. : 13.0 Min. : 1.000 Min. : 0.000   
## 1st Qu.: 0.000 1st Qu.: 40.0 1st Qu.: 3.000 1st Qu.: 3.000   
## Median : 0.000 Median : 78.0 Median : 7.000 Median : 5.000   
## Mean : 3.321 Mean : 104.5 Mean : 9.602 Mean : 7.934   
## 3rd Qu.: 3.000 3rd Qu.: 131.0 3rd Qu.: 13.000 3rd Qu.: 10.000   
## Max. :241.000 Max. :9485.0 Max. :237.000 Max. :1000.000   
##   
## largestGiftAmount averageGiftAmount yearsSinceFirstDonation  
## Min. : 5 Min. : 1.286 Min. : 0.000   
## 1st Qu.: 14 1st Qu.: 8.385 1st Qu.: 2.000   
## Median : 17 Median : 11.636 Median : 5.000   
## Mean : 20 Mean : 13.348 Mean : 5.596   
## 3rd Qu.: 23 3rd Qu.: 15.478 3rd Qu.: 9.000   
## Max. :5000 Max. :1000.000 Max. :13.000   
##   
## monthsSinceLastDonation respondedMailing  
## Min. : 0.00 FALSE:90569   
## 1st Qu.:12.00 TRUE : 4843   
## Median :14.00   
## Mean :14.36   
## 3rd Qu.:17.00   
## Max. :23.00   
##

Resolve missing age featue with mean imputation

donors <- donors %>%  
 mutate(age = ifelse(is.na(age), mean(age, na.rm = TRUE), age))  
  
summary(select(donors,age))

## age   
## Min. : 1.00   
## 1st Qu.:52.00   
## Median :61.61   
## Mean :61.61   
## 3rd Qu.:71.00   
## Max. :98.00

Resolve missing number of children with median imputation

donors <- donors %>%  
 mutate(numberChildren = ifelse(is.na(numberChildren), median(numberChildren, na.rm = TRUE), numberChildren))  
  
summary(select(donors, numberChildren))

## numberChildren   
## Min. :1.000   
## 1st Qu.:1.000   
## Median :1.000   
## Mean :1.069   
## 3rd Qu.:1.000   
## Max. :7.000

Exclude values for incomeRating and wealthRating

donors <- donors %>%  
 filter(!is.na(incomeRating) & !is.na(wealthRating) & wealthRating > 0)  
  
donors %>%  
 select(incomeRating, wealthRating) %>%  
 summary()

## incomeRating wealthRating   
## Min. :1.000 Min. :1.000   
## 1st Qu.:2.000 1st Qu.:4.000   
## Median :4.000 Median :6.000   
## Mean :3.979 Mean :5.613   
## 3rd Qu.:5.000 3rd Qu.:8.000   
## Max. :7.000 Max. :9.000

Normalize the Data

Build Normalization Function

normalize <- function(x) {  
 return((x - min(x)) / (max(x) - min(x)))  
}

Apply Normalizaiton Funciton to the dataset

donors <- donors %>%  
 mutate(age = normalize(age)) %>%  
 mutate(numberChildren = normalize(numberChildren)) %>%  
 mutate(incomeRating = normalize(incomeRating)) %>%  
 mutate(wealthRating = normalize(wealthRating)) %>%  
 mutate(mailOrderPurchases = normalize(mailOrderPurchases)) %>%  
 mutate(totalGivingAmount = normalize(totalGivingAmount)) %>%  
 mutate(numberGifts = normalize(numberGifts)) %>%  
 mutate(smallestGiftAmount = normalize(smallestGiftAmount)) %>%  
 mutate(largestGiftAmount = normalize(largestGiftAmount)) %>%  
 mutate(averageGiftAmount = normalize(averageGiftAmount)) %>%  
 mutate(yearsSinceFirstDonation = normalize(yearsSinceFirstDonation)) %>%  
 mutate(monthsSinceLastDonation = normalize(monthsSinceLastDonation))  
  
summary(donors)

## age numberChildren incomeRating wealthRating   
## Min. :0.0000 Min. :0.00000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.5155 1st Qu.:0.00000 1st Qu.:0.1667 1st Qu.:0.3750   
## Median :0.6249 Median :0.00000 Median :0.5000 Median :0.6250   
## Mean :0.6308 Mean :0.01483 Mean :0.4965 Mean :0.5766   
## 3rd Qu.:0.7526 3rd Qu.:0.00000 3rd Qu.:0.6667 3rd Qu.:0.8750   
## Max. :1.0000 Max. :1.00000 Max. :1.0000 Max. :1.0000   
## mailOrderPurchases totalGivingAmount numberGifts smallestGiftAmount  
## Min. :0.000000 Min. :0.000000 Min. :0.00000 Min. :0.00000   
## 1st Qu.:0.004149 1st Qu.:0.004945 1st Qu.:0.01271 1st Qu.:0.00600   
## Median :0.012448 Median :0.011834 Median :0.02966 Median :0.01000   
## Mean :0.025986 Mean :0.016236 Mean :0.03715 Mean :0.01538   
## 3rd Qu.:0.033195 3rd Qu.:0.021018 3rd Qu.:0.05508 3rd Qu.:0.02000   
## Max. :1.000000 Max. :1.000000 Max. :1.00000 Max. :1.00000   
## largestGiftAmount averageGiftAmount yearsSinceFirstDonation  
## Min. :0.000000 Min. :0.00000 Min. :0.0000   
## 1st Qu.:0.009045 1st Qu.:0.01405 1st Qu.:0.1818   
## Median :0.012060 Median :0.02034 Median :0.5455   
## Mean :0.014689 Mean :0.02362 Mean :0.5235   
## 3rd Qu.:0.017085 3rd Qu.:0.02750 3rd Qu.:0.8182   
## Max. :1.000000 Max. :1.00000 Max. :1.0000   
## monthsSinceLastDonation respondedMailing  
## Min. :0.0000 FALSE:45770   
## 1st Qu.:0.5217 TRUE : 2497   
## Median :0.6087   
## Mean :0.6208   
## 3rd Qu.:0.6957   
## Max. :1.0000

Splitting the data (train/test)

donors <- data.frame(donors)  
  
  
set.seed(1234)  
  
sample\_index <- sample(nrow(donors), round(nrow(donors) \* .75), replace = FALSE)  
donors\_train <- donors[sample\_index, ]  
donors\_test <- donors[-sample\_index, ]

Review dataset class distribution (imbalance)

Imbalance donors data set

round(prop.table(table(select(donors, respondedMailing), exclude = NULL)), 4)

##   
## FALSE TRUE   
## 0.9483 0.0517

donors %>%  
 select(respondedMailing) %>%  
 table(exclude = NULL) %>%  
 prop.table() %>%  
 round(4) \* 100

## .  
## FALSE TRUE   
## 94.83 5.17

Imbalance donors train data set

donors\_train %>%  
 select(respondedMailing) %>%  
 table(exclude = NULL) %>%  
 prop.table() %>%  
 round(4) \* 100

## .  
## FALSE TRUE   
## 94.88 5.12

Imbalance donors test data set

donors\_test %>%  
 select(respondedMailing) %>%  
 table(exclude = NULL) %>%  
 prop.table() %>%  
 round(4) \* 100

## .  
## FALSE TRUE   
## 94.68 5.32

Balance the training data set

library(DMwR)

## Loading required package: lattice

## Loading required package: grid

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

set.seed(1234)  
  
donors\_train <- SMOTE(respondedMailing ~ ., donors\_train, perc.over = 100, perc.under = 200)  
  
donors\_train %>%  
 select(respondedMailing) %>%  
 table(exclude = NULL) %>%  
 prop.table() %>%  
 round(4) \* 100

## .  
## FALSE TRUE   
## 50 50

Splitt off the class lables into seperate data sets with pull()

donors\_train\_labels <- as.factor(pull(donors\_train, respondedMailing))  
donors\_test\_labels <- as.factor(pull(donors\_test, respondedMailing))

Remove the class labels from our training and test datasets

donors\_train <- data.frame(select(donors\_train, -respondedMailing))  
donors\_test <- data.frame(select(donors\_test, -respondedMailing))

Build The Model

library(class)  
  
donors\_pred <-   
 knn(  
 train = donors\_train,  
 test = donors\_test,  
 cl = donors\_train\_labels,  
 k = 5  
 )  
  
head(donors\_pred)

## [1] FALSE FALSE TRUE TRUE FALSE FALSE  
## Levels: FALSE TRUE

Evaluating the Model

donors\_pred\_table <- table(donors\_test\_labels, donors\_pred)  
donors\_pred\_table

## donors\_pred  
## donors\_test\_labels FALSE TRUE  
## FALSE 6088 5337  
## TRUE 312 330

Model Accuracy as percentage

sum(diag(donors\_pred\_table)) / nrow(donors\_test) \* 100

## [1] 53.18638

Remodel including categorical variables - adjust value of k \* Approach to set the value of k as the square root of the number of training instances