## BAN 502

## Mandy Hawkins

## Random Forest Assignment

library(tidyverse)

## -- Attaching packages -------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.7  
## v tidyr 0.8.2 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

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

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(ranger)  
  
library(mice)

##   
## Attaching package: 'mice'

## The following object is masked from 'package:tidyr':  
##   
## complete

## The following objects are masked from 'package:base':  
##   
## cbind, rbind

Blood <- read\_csv("~/BAN502/Module 4/Random Forests/Random Forests Assignment/Blood.csv")

## Parsed with column specification:  
## cols(  
## Mnths\_Since\_Last = col\_integer(),  
## TotalDonations = col\_integer(),  
## Total\_Donated = col\_integer(),  
## Mnths\_Since\_First = col\_integer(),  
## DonatedMarch = col\_integer()  
## )

View(Blood)

Blood =Blood %>% mutate(DonatedMarch =as.factor(as.character(DonatedMarch))) %>% mutate(DonatedMarch = fct\_recode(DonatedMarch, "No" = "0" , "Yes" = "1"))

Blood = Blood %>% drop\_na() #delete any row with an NA value   
str(Blood) #check structure after the drop

## Classes 'tbl\_df', 'tbl' and 'data.frame': 748 obs. of 5 variables:  
## $ Mnths\_Since\_Last : int 2 0 1 2 1 4 2 1 2 5 ...  
## $ TotalDonations : int 50 13 16 20 24 4 7 12 9 46 ...  
## $ Total\_Donated : int 12500 3250 4000 5000 6000 1000 1750 3000 2250 11500 ...  
## $ Mnths\_Since\_First: int 98 28 35 45 77 4 14 35 22 98 ...  
## $ DonatedMarch : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 1 2 1 2 2 ...

## Task 1

Split the dataset into training (70%) and testing (30%) sets. Use set.seed of 1234.

set.seed(1234)  
train.rows = createDataPartition(y = Blood$DonatedMarch, p=0.7, list = FALSE) #70% in training  
train = Blood[train.rows,]   
test = Blood[-train.rows,]

## Task 2

Create a random forest model on the training set to predict DonatedMarch using all of the variables in the dataset. Use caret’s trainControl function to set up 10 fold cross-validation. Use a random number seed of 123. Use 100 trees.

Random forest

fit\_control = trainControl(method = "cv", number = 10) #set up 10 fold cross-validation  
set.seed(123)

### Task 3

Using varImp, what is the most important variable in the model, what is the least important? TotalDonations is the most important and Total\_Donated is the least important.

set.seed(123)   
rf\_fit = train(DonatedMarch ~.,   
 data = Blood,   
 method = "ranger",   
 importance = "permutation",   
 num.trees = 100,  
 trControl = fit\_control)

Check out random forest details

varImp(rf\_fit)

## ranger variable importance  
##   
## Overall  
## TotalDonations 100.00  
## Mnths\_Since\_First 78.15  
## Mnths\_Since\_Last 18.51  
## Total\_Donated 0.00

## Task 4

Develop predictions on the training set. Use the “head” function to display the ﬁrst six predictions.

predRF = predict(rf\_fit, train)  
head(predRF)

## [1] Yes Yes No No Yes Yes  
## Levels: No Yes

## Task 5

Use the model to create a confusion using caret’s confusionMatrix function for the training set. What is the accuracy, sensitivity, and speciﬁcity of the model? The accuracy is 89%, sensitivity is 0.58 and specificity is 0.98

## Task 6

How does the accuracy of the model compare to a naive model that assumes that all observations are in the majority class?

confusionMatrix(predRF, train$DonatedMarch, positive = "Yes")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 394 54  
## Yes 5 71  
##   
## Accuracy : 0.8874   
## 95% CI : (0.8572, 0.9132)  
## No Information Rate : 0.7615   
## P-Value [Acc > NIR] : 1.933e-13   
##   
## Kappa : 0.6419   
## Mcnemar's Test P-Value : 4.129e-10   
##   
## Sensitivity : 0.5680   
## Specificity : 0.9875   
## Pos Pred Value : 0.9342   
## Neg Pred Value : 0.8795   
## Prevalence : 0.2385   
## Detection Rate : 0.1355   
## Detection Prevalence : 0.1450   
## Balanced Accuracy : 0.7777   
##   
## 'Positive' Class : Yes   
##

## Task 7

Use the model to develop predictions on the test set. Develop a confusion matrix. How does the model perform on the testing set? The accuracy is not as good on the testing set.

predRF\_test = predict(rf\_fit, newdata = test)

Confusion matrix

confusionMatrix(predRF\_test, test$DonatedMarch, positive = "Yes")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 171 24  
## Yes 0 29  
##   
## Accuracy : 0.8929   
## 95% CI : (0.8448, 0.9301)  
## No Information Rate : 0.7634   
## P-Value [Acc > NIR] : 6.149e-07   
##   
## Kappa : 0.6485   
## Mcnemar's Test P-Value : 2.668e-06   
##   
## Sensitivity : 0.5472   
## Specificity : 1.0000   
## Pos Pred Value : 1.0000   
## Neg Pred Value : 0.8769   
## Prevalence : 0.2366   
## Detection Rate : 0.1295   
## Detection Prevalence : 0.1295   
## Balanced Accuracy : 0.7736   
##   
## 'Positive' Class : Yes   
##