## Module 4 – Random Forests Assignment

# Lea Shipley

options(tidyverse.quiet = TRUE)  
library(tidyverse)  
library(caret)

## Warning: package 'caret' was built under R version 3.6.2

## Loading required package: lattice

##   
## Attaching package: 'caret'

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

library(ranger)

## Warning: package 'ranger' was built under R version 3.6.2

blood <- read\_csv("Blood.csv")

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

blood = blood %>% mutate(DonatedMarch = as\_factor(as.numeric(DonatedMarch))) %>%  
mutate(DonatedMarch = fct\_recode(DonatedMarch,"Yes" = "1","No" = "0"))

**Task 1**

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**

fit\_control = trainControl(method = "cv", number = 10)   
  
  
set.seed(123)   
rf\_fit = train(x=as.matrix(train[,-5]), y=as.matrix(train$DonatedMarch),  
 method = "ranger",   
 importance = "permutation",  
 trControl = fit\_control,  
 num.trees = 100)  
#notice exclusion of "data = " line in block of code above. Not needed as data is explicitly indicated via non-formula interface

**Task 3**

varImp(rf\_fit)

## ranger variable importance  
##   
## Overall  
## TotalDonations 100.00  
## Total\_Donated 57.72  
## Mnths\_Since\_First 35.32  
## Mnths\_Since\_Last 0.00

rf\_fit

## Random Forest   
##   
## 524 samples  
## 4 predictor  
## 2 classes: 'No', 'Yes'   
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 472, 472, 471, 471, 471, 472, ...   
## Resampling results across tuning parameters:  
##   
## mtry splitrule Accuracy Kappa   
## 2 gini 0.7519956 0.2410225  
## 2 extratrees 0.7767779 0.2904529  
## 3 gini 0.7406386 0.2326286  
## 3 extratrees 0.7577649 0.2447921  
## 4 gini 0.7482946 0.2498497  
## 4 extratrees 0.7424528 0.2099090  
##   
## Tuning parameter 'min.node.size' was held constant at a value of 1  
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were mtry = 2, splitrule =  
## extratrees and min.node.size = 1.

“TotalDonations” is the most important variable in the dataset. “Mnths\_Since\_Last” is the least important variable.

**Task 4**

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

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

**Task 5**

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 391 43  
## Yes 8 82  
##   
## Accuracy : 0.9027   
## 95% CI : (0.874, 0.9267)  
## No Information Rate : 0.7615   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.7036   
##   
## Mcnemar's Test P-Value : 1.927e-06   
##   
## Sensitivity : 0.6560   
## Specificity : 0.9799   
## Pos Pred Value : 0.9111   
## Neg Pred Value : 0.9009   
## Prevalence : 0.2385   
## Detection Rate : 0.1565   
## Detection Prevalence : 0.1718   
## Balanced Accuracy : 0.8180   
##   
## 'Positive' Class : Yes   
##

The accuracy of the model on the training set is 0.9027, with sensitivity of 0.6560, and specificity of 0.9799.

**Task 6** The accuracy of the model (0.9027) is significantly better than a naive model (0.7615).

**Task 7**

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

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 154 34  
## Yes 17 19  
##   
## Accuracy : 0.7723   
## 95% CI : (0.7118, 0.8255)  
## No Information Rate : 0.7634   
## P-Value [Acc > NIR] : 0.41192   
##   
## Kappa : 0.2913   
##   
## Mcnemar's Test P-Value : 0.02506   
##   
## Sensitivity : 0.35849   
## Specificity : 0.90058   
## Pos Pred Value : 0.52778   
## Neg Pred Value : 0.81915   
## Prevalence : 0.23661   
## Detection Rate : 0.08482   
## Detection Prevalence : 0.16071   
## Balanced Accuracy : 0.62954   
##   
## 'Positive' Class : Yes   
##

The model does not perform quite as well on the testing set. Accuracy decreases from 0.9027 to 0.7723. Sensitivity and specificity also decrease compared to the model on the training set.

**Task 8**

The model could possibly be used for a targeted recruitment of blood donors. I would be hesitant to recommend this model for real-world use. Concerns for this model would be that the data could be overfit with the significant decrease in accuracy that occurred on the testing set.