**ITM 4700 Assignment 1**

**INTRODUCTION**

One of the first things we did for this assignment was create a new column labeled “Amount\_Category” which was created using the following IF statement:

**=IF(H2>99,"Low", IF(H2>499,"Medium", IF(H2>999,"High","Insignificant")))**.

We created this IF statement to sort donors into groups based on their level of contribution. Not only did this help us create Classification technique-based algorithms, it also revealed which data was not useful to us. For example, our IF statement sorted anyone who donated less than $100 into an “insignificant” category so when we generated the lists based on amount\_Sum, we knew which data to ignore and which to explore further.

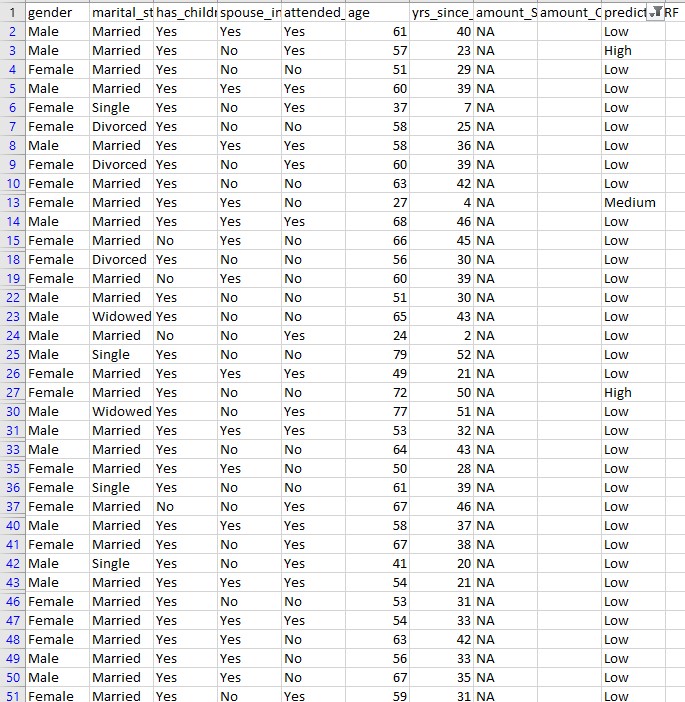
The discrepancy in accuracy between Decision Tree and Random Forest was large when comparing the two algorithms to each other. When using Decision Tree, we found the accuracy percentage to be far too low to put faith in the PredictedDT. When using Random Forest, we found that it did a *far* better job at generating a higher accuracy for our results which allowed us to be more certain about our predictions as to which donors will donate at each listed amount ($100, $500, and $1,000).

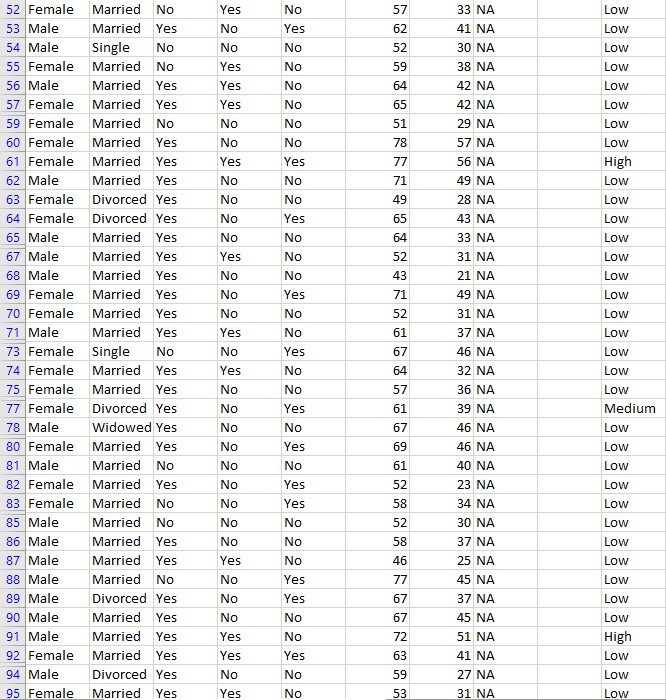
Comparing the two different techniques - Regression and Classification - we found Regression gives much lower accuracy percentages. We speculate this is because it is much harder to predict an exact dollar amount a person will contribute compared to the Classification technique which simply sorts donors into a Low, Medium, and High category based on which level the algorithm believes the donor will most likely fall into. The Random Forest Classification Technique gave us the highest accuracy percentages of 82.8% and 82.4% which means this technique gave the best predictions for our Predicted lists.

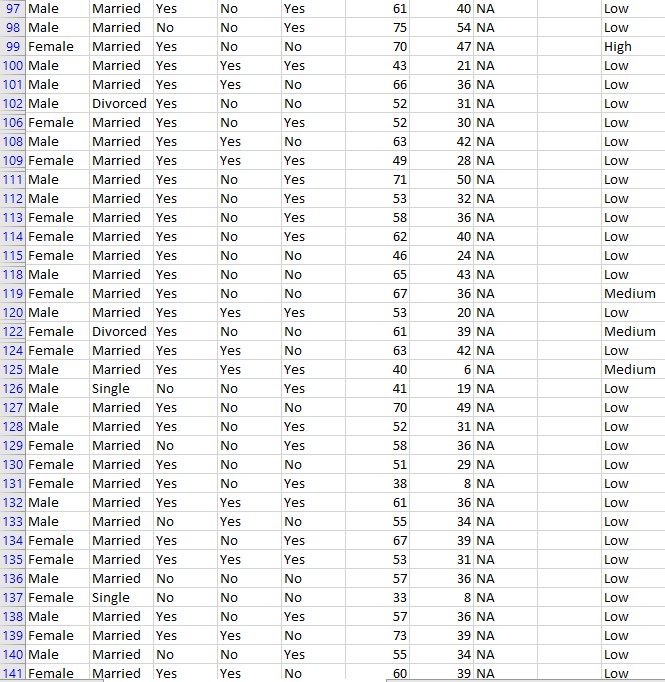
Finally, we wanted to explain which independent variables we found important, and how each list differed slightly based on level of contribution and which technique we employed. For the Classification technique, we found the variables *marital status, yrs since grad,* and *age* to be the 3 variables that were used in generating both the first and the third list. The first list relied more heavily on gender and the third list relied more on if the donors attended the event. For the regression technique, we included all variables but gender. We speculated that the Regression technique requires as much information as possible to generate more accurate numeric predictions. Comparing it to the Classification technique, it seems Regression is harder to make more accurate because the nature of specificity of predictions is harder for numeric than it is for categorical variables. Overall, Random Forest Classification Technique is by far the better technique.

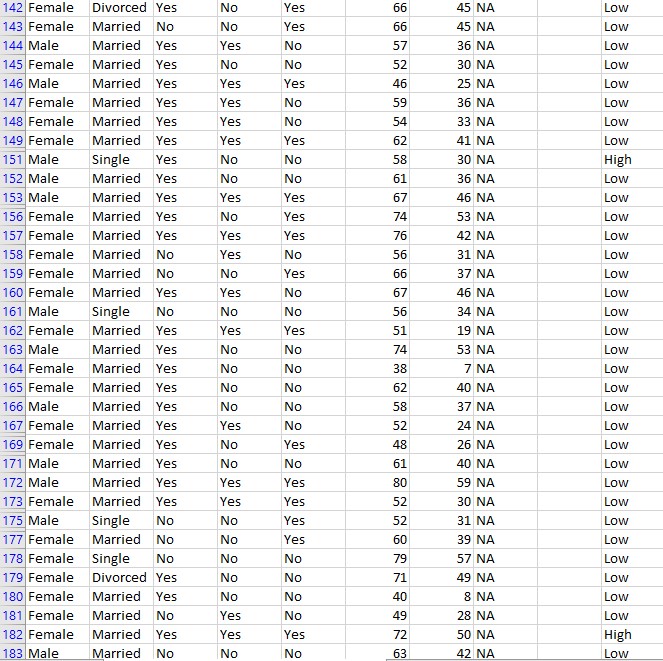
**3 LISTS OF DONORS**

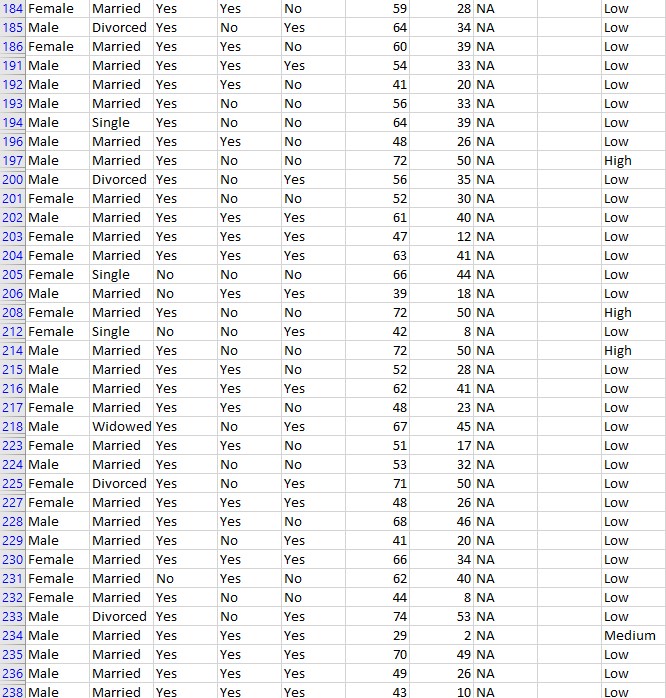
1. **Random Forest Classification Technique; Dependent Variable = Amount\_Category**

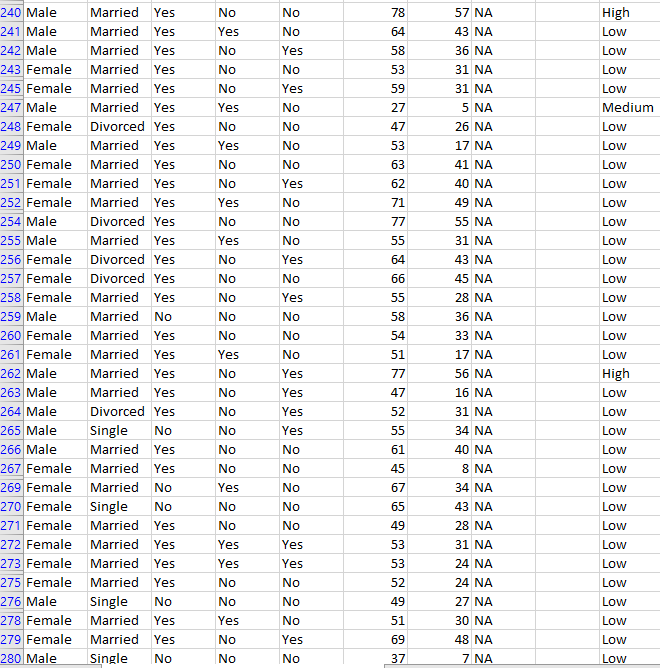




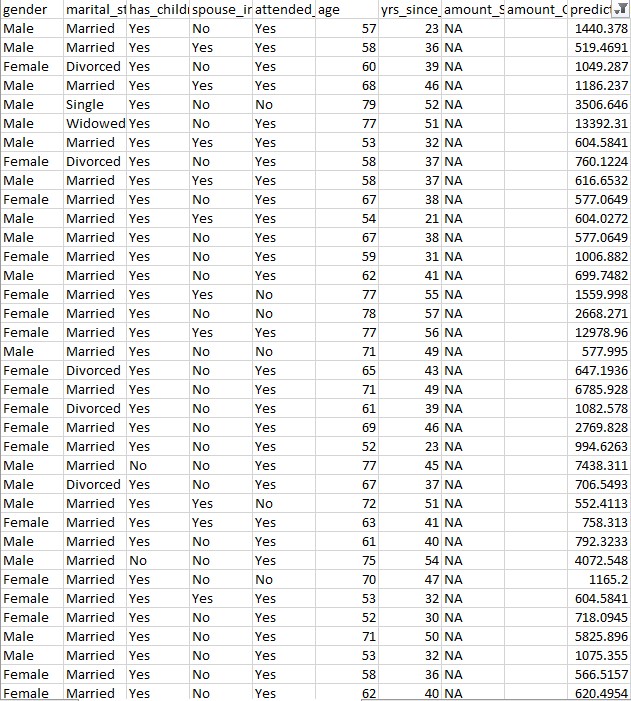


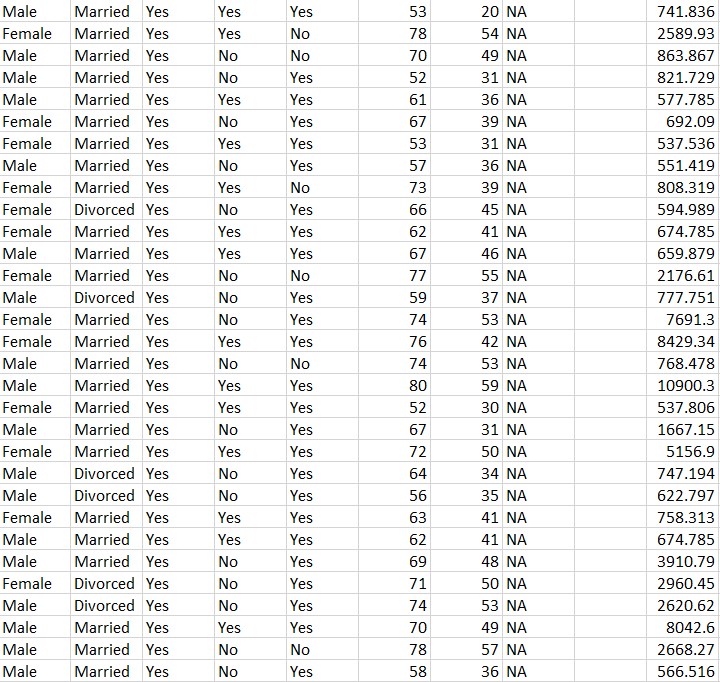


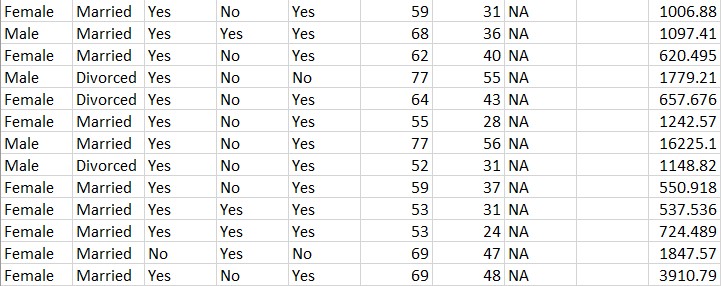


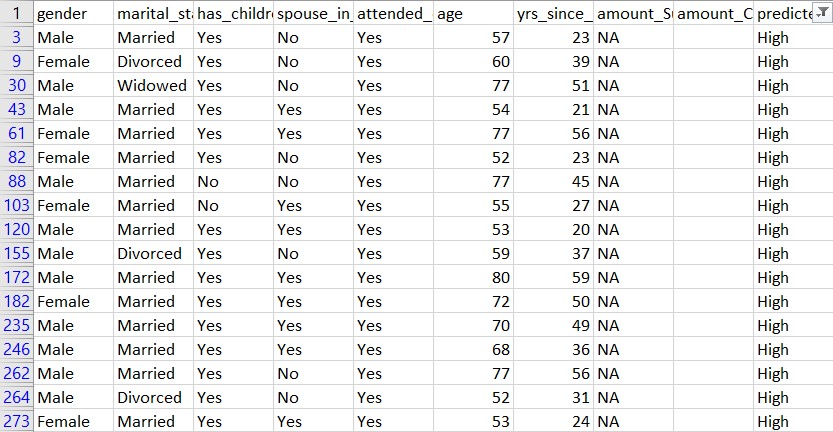


**2. Random Forest Regression Technique; Dependent Variable = Amount\_Sum**







**3. Random Forest Classification Technique; Dependent Variable = Amount\_Category**

**R Codes Used To Generate Our Tables**

**1. Random Forest Classification Technique; Accuracy = 82.4%**

dataset <- read.csv("Donor 2020 testing data included.csv")

nrow(dataset)

ncol(dataset)

head(dataset)

trainingsize <- as.integer((500)\*(1-0/100))

validationsize <- (500)-trainingsize

testingsize <- nrow(dataset)-(500)

training <- head(dataset,trainingsize)

validation <- tail(head(dataset,trainingsize+validationsize),validationsize)

testing <- tail(dataset,nrow(dataset)-(trainingsize+validationsize))

if(validationsize==0) {validation <- training}

nrow(training)

nrow(validation)

nrow(testing)

#install.packages("randomForest")

library(randomForest)

model <- randomForest(factor(amount\_Category)~gender+marital\_status+age+yrs\_since\_grad,data=training,ntrees=500)

importance(model)

validation$predictedRF <- predict(model,validation,type="class")

confusionmatrix <- table(validation$predictedRF,validation$amount\_Category)

confusionmatrix

accuracy <- (1 - mean(validation$predictedRF != validation$amount\_Category,na.rm=TRUE))\*100

paste(accuracy,"%",sep="")

predictedRF <- predict(model,testing,type="class")

testing <- cbind(testing,predictedRF)

write.csv(testing,"Donor 2020 testing data included\_Prediction.csv",row.names=FALSE)

**2. Random Forest Regression Technique R code; Accuracy = 69.80%**

dataset <- read.csv("Donor 2020 testing data included.csv")

nrow(dataset)

ncol(dataset)

head(dataset)

trainingsize <- as.integer((500)\*(1-0/100))

validationsize <- (500)-trainingsize

testingsize <- nrow(dataset)-(500)

training <- head(dataset,trainingsize)

validation <- tail(head(dataset,trainingsize+validationsize),validationsize)

testing <- tail(dataset,nrow(dataset)-(trainingsize+validationsize))

if(validationsize==0) {validation <- training}

nrow(training)

nrow(validation)

nrow(testing)

#install.packages("randomForest")

library(randomForest)

model <- randomForest(amount\_Sum~marital\_status+has\_children+spouse\_in\_db+attended\_event+age+yrs\_since\_grad,data=training,ntrees=500)

importance(model)

validation$predictedRF <- predict(model,validation,type="class")

accuracy <- 100 \* cor(validation$predictedRF,validation$amount\_Sum) ^ 2

paste(accuracy,"%",sep="")

predictedRF <- predict(model,testing,type="class")

testing <- cbind(testing,predictedRF)

write.csv(testing,"Donor 2020 testing data included\_Prediction.csv",row.names=FALSE)

**3. Random Forest Classification Technique R code; Accuracy =** **82.8%**

dataset <- read.csv("Donor 2020 testing data included.csv")

nrow(dataset)

ncol(dataset)

head(dataset)

trainingsize <- as.integer((500)\*(1-0/100))

validationsize <- (500)-trainingsize

testingsize <- nrow(dataset)-(500)

training <- head(dataset,trainingsize)

validation <- tail(head(dataset,trainingsize+validationsize),validationsize)

testing <- tail(dataset,nrow(dataset)-(trainingsize+validationsize))

if(validationsize==0) {validation <- training}

nrow(training)

nrow(validation)

nrow(testing)

#install.packages("randomForest")

library(randomForest)

model <- randomForest(factor(amount\_Category)~marital\_status+attended\_event+age+yrs\_since\_grad,data=training,ntrees=500)

importance(model)

validation$predictedRF <- predict(model,validation,type="class")

confusionmatrix <- table(validation$predictedRF,validation$amount\_Category)

confusionmatrix

accuracy <- (1 - mean(validation$predictedRF != validation$amount\_Category,na.rm=TRUE))\*100

paste(accuracy,"%",sep="")

predictedRF <- predict(model,testing,type="class")

testing <- cbind(testing,predictedRF)

write.csv(testing,"Donor 2020 testing data included\_Prediction.csv",row.names=FALSE)