rm(list=ls())

setwd("/Users/shikharmisra/Desktop/safedriver")

getwd()

train = read.csv("train.csv",header = T)

test = read.csv("test.csv",header = T)

#getting to know about the data first

str(train)

#Store Values in data frame(study)

MissingData = data.frame(varaibles = colnames(test), MissingInfo = apply(test,2,function(x)sum(is.na(x))))

#capping the values

##outlier analysis

fun <- function(x){

quantiles <- quantile( x, c(.05, .95 ) )

x[ x < quantiles[1] ] <- quantiles[1]

x[ x > quantiles[2] ] <- quantiles[2]

x

}

test$ps\_reg\_01=fun(test$ps\_reg\_01)

test$ps\_reg\_02=fun(test$ps\_reg\_02)

test$ps\_reg\_03=fun(test$ps\_reg\_03)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

test$ps\_car\_12=fun(test$ps\_car\_12)

test$ps\_car\_13=fun(test$ps\_car\_13)

test$ps\_car\_14=fun(test$ps\_car\_14)

test$ps\_car\_15=fun(test$ps\_car\_15)

test$ps\_calc\_01=fun(test$ps\_calc\_01)

test$ps\_calc\_02=fun(test$ps\_calc\_02)

test$ps\_calc\_03=fun(test$ps\_calc\_03)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#Exhaustive graph plotting

##not able to make graph by analysing that many rows

###taking subset of the dataset with similar traits i.e. the nature of subsetted data would be similar

#as that of original dataset

# sample without replacement

mysample <- train[sample(1:nrow(train), 5000,

replace=FALSE),]

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

hist(train$target)

hist(mysample$target)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

library(ggplot2)

graph1= ggplot(data=mysample,

aes(x=target, y=ps\_reg\_01)) +

geom\_jitter()

graph1

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph2= ggplot(data=mysample,

aes(x=target, y=ps\_reg\_02)) +

geom\_jitter()

graph2

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph3= ggplot(data=mysample,

aes(x=target, y=ps\_reg\_03)) +

geom\_jitter()

graph3

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph4= ggplot(data=mysample,

aes(x=target, y=ps\_car\_12)) +

geom\_jitter()

graph4

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph5= ggplot(data=mysample,

aes(x=target, y=ps\_car\_13)) +

geom\_jitter()

graph5

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph6= ggplot(data=mysample,

aes(x=target, y=ps\_car\_14)) +

geom\_jitter()

graph6

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph7= ggplot(data=mysample,

aes(x=target, y=ps\_car\_15)) +

geom\_jitter()

graph7

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph8= ggplot(data=mysample,

aes(x=target, y=ps\_calc\_01)) +

geom\_jitter()

graph8

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph9= ggplot(data=mysample,

aes(x=target, y=ps\_calc\_02)) +

geom\_jitter()

graph9

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

graph10= ggplot(data=mysample,

aes(x=target, y=ps\_calc\_03)) +

geom\_jitter()

graph10

#Feature Engineering (reading other sources what more features could be added if possible)

##data not totally explained cant think of one currently

### Saperating the variables which are really necessary for model building and leaving those aren't

# load the library

library(mlbench)

library(rpart)

library(caret)

train$id = NULL

mysample$id = NULL

fit = rpart(target ~ ., data = train, method = "anova")

#variable Importance

gbmImp <- varImp(fit, scale = FALSE)

gbmImp

#Divide into train and test

c.train <- train[1:nrow(train),]

c.test <- train[-(1:nrow(train)),]

library(h2o)

localH2O <- h2o.init(nthreads = -1)

h2o.init()

#data to h2o cluster

train.h2o <- as.h2o(c.train)

test.h2o <- as.h2o(c.test)

track.h2o <- as.h2o(test)

#dependent variable (target)

y.dep <- 1

#independent variables (dropping ID variables)

x.indep <- c(2:58)

#Random Forest

system.time(

rforest.model <- h2o.randomForest(y=y.dep, x=x.indep, training\_frame = train.h2o, ntrees = 1000, mtries = 3, max\_depth = 4, seed = 1122)

)

#making predictions on unseen data

system.time(predict.rforest <- as.data.frame(h2o.predict(rforest.model, track.h2o)))

#writing submission file

sub\_rf <- data.frame(id = test$id, target = predict.rforest$predict)

write.csv(sub\_rf, file = "trippy.csv", row.names = F) # around 4180 rank in kaggle with score of 0.24180

#GBM

system.time(

gbm.model <- h2o.gbm(y=y.dep, x=x.indep, training\_frame = train.h2o, ntrees = 1000, max\_depth = 4, learn\_rate = 0.01, seed = 1122)

)

h2o.performance (gbm.model)

#making prediction and writing submission file

predict.gbm <- as.data.frame(h2o.predict(gbm.model, track.h2o))

sub\_gbm <- data.frame(id = test$id, target = predict.gbm$predict)

write.csv(sub\_gbm, file = "sub\_gbm.csv", row.names = F)# around 3600s rank in kaggle with score of 0.27190