Project : Retail Store prediction using predictive analytics in R

Problem statement: This data set is related with retail domain and challenge is to predict whether a store should get opened or not based on certain factors such as sales, population,area etc.

We have given you two datasets , store_train.csv and store_test.csv . You need to use data store_train to build predictive model for response variable ‘store’. store_test data contains all other factors except ‘store’, you need to predict that using the model that you developed and submit your predicted values in a csv files.

Solution:

```
train <- read.csv('store_train.csv',stringsAsFactors = F)
test <- read.csv('store_test.csv',stringsAsFactors = F)
train$population[is.na(train$population)] <- mean(train$population, na.rm = TRUE)
train <- train[, !(names(train) %in% c("Id"))]

test$population[is.na(test$population)] <- mean(test$population, na.rm = TRUE)
test$country[is.na(test$country)] <- mean(test$country, na.rm = TRUE)
test <- test[, !(names(test) %in% c("Id"))]

test$store=NA
train$data='train'
test$data='test'
all <- rbind(train, test)

CreateDummies=function(data, var, freq_cutoff=100){
  t=table(data[,var])
  t=t[t>freq_cutoff]
  t=sort(t)
  categories=names(t)[-1]
  for (cat in categories) {
    name=paste(var, cat, sep="_")
    name=gsub(" ","", name)
    name=gsub("-","_",name)
    name=gsub("\\?", "Q", name)
    name=gsub("<","LT_",name)
    name=gsub("\\+", "", name)
    name=gsub(">", "GT_",name)
    name=gsub("=", "EQ_",name)
    name=gsub(",","", name)
    name=gsub("/","_",name)
    data[,name]=as.numeric(data[,var]==cat)
  }
  data[,var]=NULL
  return(data)
}
library("dplyr")
all<- all %>%
  select(-countyname, -storecode, -Areaname, -countytownname, -state_alpha)
```

```
for_dummy_vars=c('country', 'State', 'CouSub', 'store_Type')
for (var in for_dummy_vars){
  all=CreateDummies (all, var,50)
}

for (col in names (all)){
  if(sum(is.na(all[,col]))>0 & !(col %in% c("data","store"))){
    all[is.na(all[,col]), col]=mean(all[all$data == 'train', col],na.rm=T)
  }
}
train <- all %>% filter (data == 'train') %>% select(-data)
test <- all %>% filter (data == 'test') %>% select (-data, -store)
model_lm=lm(store~.,data=train)
library(car)
sort(vif(model_lm), decreasing = T)[1:3]
model_lm=lm(store~.-sales0, data=train)
sort(vif(model_lm), decreasing = T)[1:3]
model_lm=lm(store~.-sales0-sales2, data=train)
sort(vif(model_lm), decreasing=T)[1:3]
model_lm=lm(store~.-sales0-sales2-sales3, data=train)
sort(vif(model_lm), decreasing=T)[1:3]
train$store <- as.factor(train$store)
library(randomForest)
rf_fit=randomForest (store~.,data=train,ntree=20)
store_predictions=predict(rf_fit,newdata= test, type="prob")[,2]
write.csv(store_predictions, file = "store_predictions.csv", row.names = FALSE)
```