# XGB Test Attempt

JJ Goh

12/8/2020

## Check WD

```
getwd()
## [1] "C:/Users/JJ/Documents/410final"
```

## Load dependencies

```
library(xgboost)
library(knitr)
library(Matrix)
library(dplyr)
library(ggplot2)
library(GGally)
library(data.table)
```

## Import data

```
df = read.csv ("xgb_1.csv")
df2 = read.csv("MasterJshan_v2.csv")
head(df)
##   Subscription.Type Subscription.Event.Type Purchase.Store Demo.User
## 1           Limited        INITIAL_PURCHASE            App       Yes
## 2           Limited        INITIAL_PURCHASE            Web        No
## 3           Limited        INITIAL_PURCHASE            Web        No
## 4           Limited        INITIAL_PURCHASE            App       Yes
## 5           Limited        INITIAL_PURCHASE            App        No
## 6           Limited        INITIAL_PURCHASE            Web       Yes
##   Free.Trial.User Auto.Renew   Country User.Type Email.Subscriber
## 1              No        Off US/Canada  Consumer              Yes
## 2              No        Off     Other  Consumer               No
## 3              No        Off US/Canada  Consumer              Yes
## 4              No        Off US/Canada  Consumer              Yes
## 5              No        Off US/Canada  Consumer              Yes
## 6              No        Off US/Canada  Consumer              Yes
```

```
##   champion_binary
## 1              No
## 2              No
## 3              No
## 4              No
## 5              No
## 6              No
```

## Test/Train Split (80/20 split)

```
samp.size = floor(0.8 * nrow(df)) #80% of the sample size
train.ind <- sample(seq_len(nrow(df)), size = samp.size)
train_df= df[train.ind, ]
test_df= df[-train.ind, ]
```

## Data Transformation

```
train_1 = data.table(train_df)
matrix_train_1 = sparse.model.matrix(champion_binary~., data = train_1)
prediction = train_1[,champion_binary] == "Yes"
```

```
head(matrix_train_1)
## 6 x 11 sparse Matrix of class "dgCMatrix"
##
## 1 1 1 1 1 . . 1 . 1 . 1
## 2 1 1 . 1 . . 1 1 . 1 .
## 3 1 1 . 1 . . . 1 . . .
## 4 1 1 1 . . 1 1 . 1 . 1
## 5 1 1 1 1 . 1 . . 1 . 1
## 6 1 1 . . 1 . 1 . 1 . 1
```

## Build Model

```
bst <- xgboost(data = matrix_train_1, label = prediction, max_depth = 4,
               eta = 1, nthread = 2, nrounds = 8,objective = "binary:logistic")
## [1]  train-error:0.053874
## [2]  train-error:0.053906
## [3]  train-error:0.053874
## [4]  train-error:0.053906
## [5]  train-error:0.053874
## [6]  train-error:0.053874
## [7]  train-error:0.053874
## [8]  train-error:0.053874
```

```r
importance <- xgb.importance(feature_names = colnames(matrix_train_1), model = bst)
head(importance)
```

```
##                            Feature       Gain      Cover  Frequency
## 1:                 User.TypeOther 0.52104006 0.26760410 0.07608696
## 2:       Subscription.TypeLimited 0.14046544 0.15908023 0.11956522
## 3:                    Auto.RenewOn 0.06598849 0.10344227 0.07608696
## 4:              Purchase.StoreWeb 0.05852964 0.10994994 0.10869565
## 5: Subscription.Event.TypeRENEWAL 0.05137734 0.05145595 0.07608696
## 6:             Free.Trial.UserYes 0.04598744 0.08054663 0.07608696
```

```r
xgb.plot.importance(importance_matrix = importance)
```