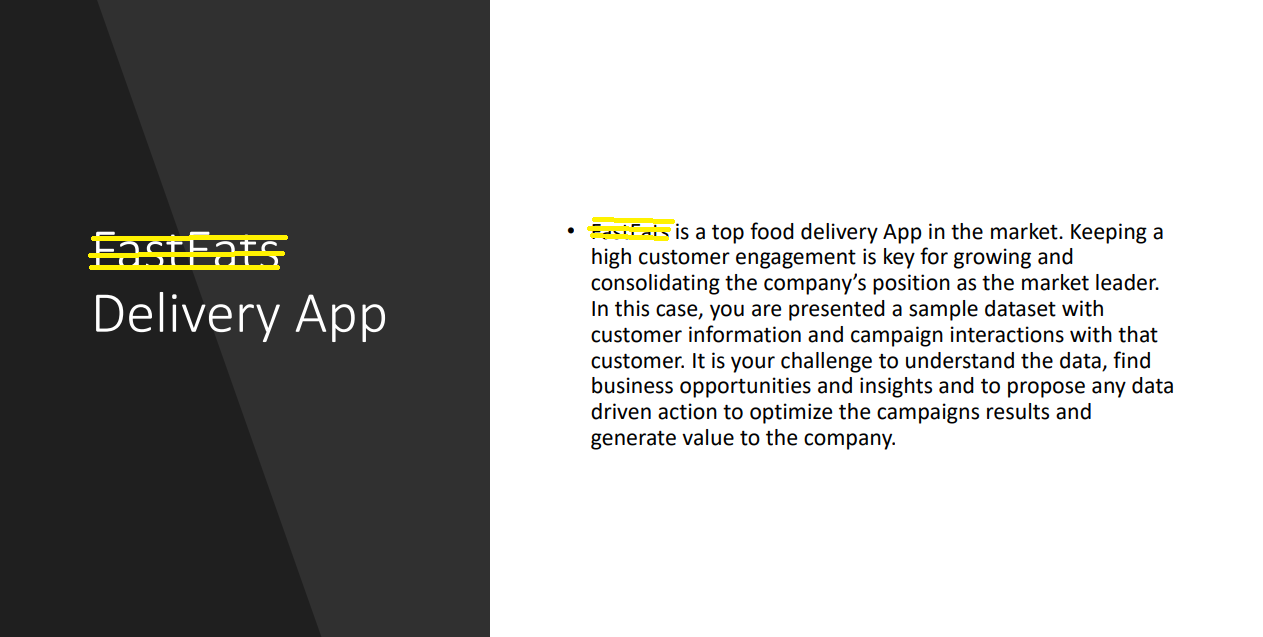
Problem: Predicting Customer Response based on Demographics & Buying for Food Delivery Client

Solution: Supervised Classification with XgBoost using Feature Engineering for testing with most AUC and Analyzing with SHAP plots.



library(caret)

library(lattice)

library(ggplot2)

library(tidyverse)

library(dplyr)

library(forcats)

fasteats = read.csv(file=file.choose(),header=TRUE)

str(fasteats)

library(skimr)

skim(fasteats)

summary(fasteats)

sum(is.na(fasteats$Income))

fasteats <- fasteats %>%

mutate(Year\_Birth = ifelse(Year\_Birth < 1940, median(fasteats$Year\_Birth), Year\_Birth))

library(lubridate)

fasteats$Dt\_Customer <- as.Date(fasteats$Dt\_Customer, format = "%m/%d/%y")

max\_length <- max(nchar(fasteats$ID))

fasteats$ID <- as.numeric(sprintf("%0\*d", max\_length, as.integer(fasteats$ID)))

# Removing the ID col

fasteats <- select(fasteats, -ID)

fasteats$Marital\_Status <- factor(fasteats$Marital\_Status)

fasteats$Marital\_Status <- recode(fasteats$Marital\_Status, "Alone" = "Single")

fasteats$Marital\_Status <- fct\_lump(fasteats$Marital\_Status, n = 5, other\_level = "Other")

#fasteats$Response<-as.factor(fasteats$Response)

#fasteats$Response<-fct\_recode(fasteats$Response, No = "0", Yes = "1")

fasteats$Education <- as.factor(fasteats$Education)

levels(fasteats$Education)

preProcess\_missingdata\_model <- preProcess(fasteats, method='medianImpute')

preProcess\_missingdata\_model

fasteats <- predict(preProcess\_missingdata\_model,newdata=fasteats)

skim(fasteats)

fasteats\_predictors\_dummy <- model.matrix(Response~ ., data = fasteats)

fasteats\_predictors\_dummy <- data.frame(fasteats\_predictors\_dummy[,-1])

fasteats <- cbind(Response = fasteats$Response, fasteats\_predictors\_dummy)

fasteats$Response <- as.factor(fasteats$Response)

fasteats$Response <- fct\_recode(fasteats$Response, noresponse = "0",response = "1")

fasteats$Response<- relevel(fasteats$Response, ref = "response")#this makes the first level "response"

set.seed(99)

index <- createDataPartition(fasteats$Response, p = .8,list = FALSE)

fasteats\_train <- fasteats[index,]

fasteats\_test <- fasteats[-index,]

##if we don't replace my missing values before converting them to dummy values I am missing some rows.

##preProcess\_missingdata\_model <- preProcess(fasteats, method='medianImpute')

##preProcess\_missingdata\_model

##fasteats\_train <-predict(preProcess\_missingdata\_model,newdata=fasteats\_train)

##fasteats\_test <-predict(preProcess\_missingdata\_model,newdata=fasteats\_test)

set.seed(10)

model\_fasteats<-train(Response~.,

data= fasteats\_train,

method="xgbTree",

tuneGrid= expand.grid(nrounds = c(50,200),

eta = c(0.025, 0.05, 0.1),

max\_depth = c(2, 3, 4, 5),

gamma = 0,

colsample\_bytree = 1,

min\_child\_weight = 1,

subsample = 1),

trControl=trainControl(method = "cv",

number = 5)

)

model\_fasteats

model\_fasteats$bestTune

plot(varImp(model\_fasteats))

fasteats\_predictions<- predict(model\_fasteats, fasteats\_test, type = "prob")

library(ROCR)

pred\_tree<-prediction(fasteats\_predictions$response,fasteats\_test$Response,

label.ordering = c("noresponse","response"))

perf\_tree <- performance(pred\_tree, "tpr", "fpr")

plot(perf\_tree, colorize=TRUE)

auc\_tree<-unlist(slot(performance(pred\_tree, "auc"), "y.values"))

auc\_tree

install.packages("SHAPforxgboost")

library(SHAPforxgboost)

Xdata<-as.matrix(select(fasteats\_train,-c(Response))) # change data to matrix for plots

# Crunch SHAP values

shap <- shap.prep(model\_fasteats$finalModel, X\_train = Xdata)

# SHAP importance plot

shap.plot.summary(shap)