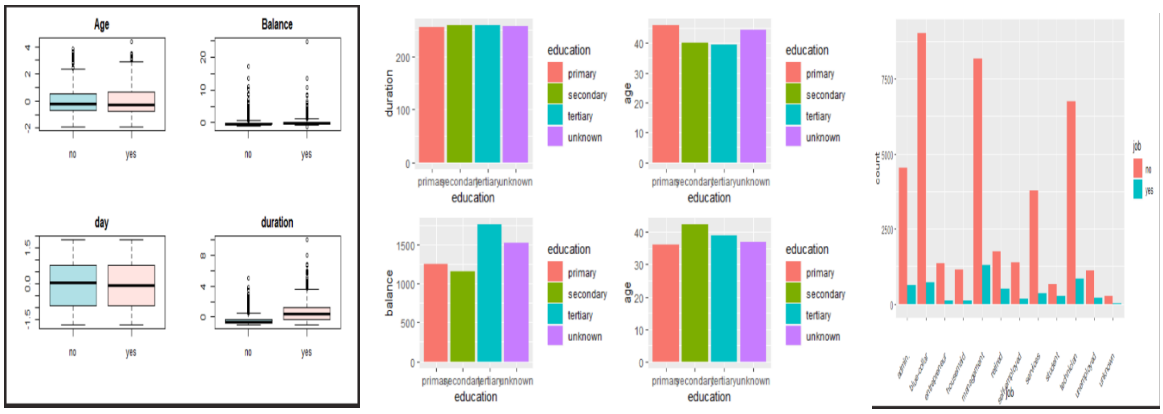
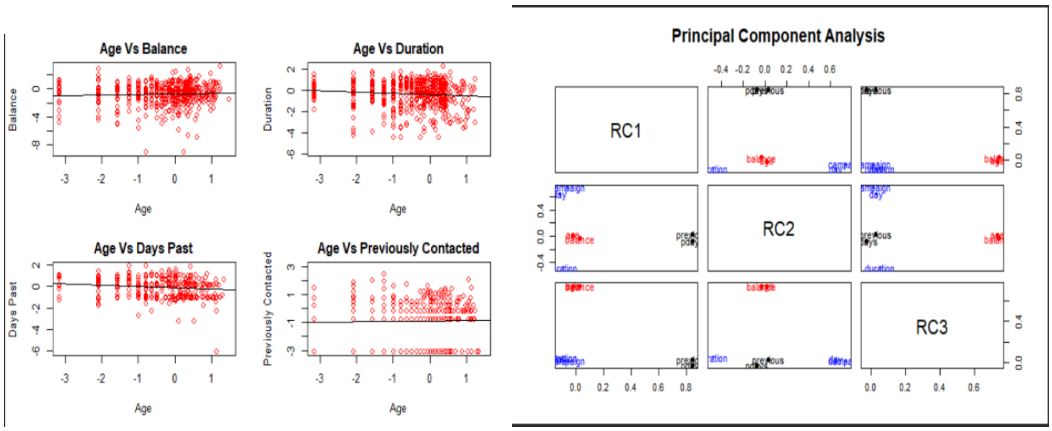
NAME – FAROOQUE AKHTAR **EMAIL : akhtar.f@husky.neu.edu**

Classification goal is to predict if client will subcribe to term deposit or not.EDA is as follows:

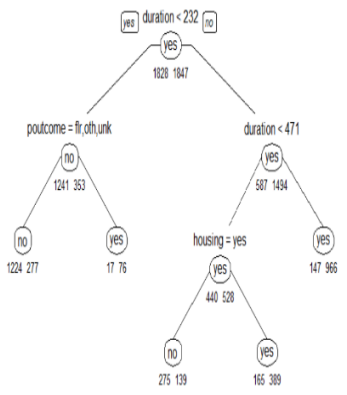


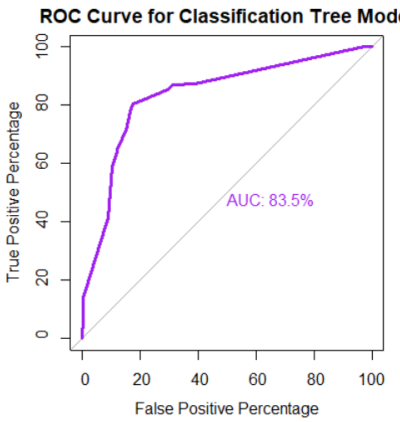
I have made Boxplot of factors vs response variable, barplots among factors to find correlation. Some of the prediction from above plots are: More duration of calls lead to better conversion rates.

After Exploratory Data Analysis, I have performed PCA to find the most important factors contributing to Class of interest.



AfterPCA, I have applied several models. I am showing the output of Classification Tree as well as its ROC Curve which means it was able to separate classes at a better rate.






```
          Accuracy : 0.7861
            95% CI : (0.7601, 0.8105)
   No Information Rate : 0.5057
   P-Value [Acc > NIR] : <2e-16

             Kappa : 0.5721
 Mcnemar's Test P-Value : 0.505

        Sensitivity : 0.7989
        Specificity : 0.7731
     Pos Pred Value : 0.7827
     Neg Pred Value : 0.7898
         Prevalence : 0.5057
     Detection Rate : 0.4040
   Detection Prevalence : 0.5162
      Balanced Accuracy : 0.7860

    'Positive' Class : no
```

CODE :

I am also attaching the html file which contains the entire code of the project.

```
# GGPLOT for Education with the other predictors
```{r}
education_duration<-summarise(group_by(bank.full,education),duration=mean(duratio
n))
education_duration
p1<-ggplot(education_duration,aes(x=education,y=duration,fill=education))+
geom_bar(stat='identity')

education_balance<-summarise(group_by(bank.full,education),balance=mean(balance))
education_balance
p2<-ggplot(education_balance,aes(x=education,y=balance,fill=education))+
geom_bar(stat='identity')

education_age<-summarise(group_by(bank.full,education),age=mean(age))
education_age
p3<-ggplot(education_age,aes(x=education,y=age,fill=education))+
geom_bar(stat='identity')

education_pdays<-summarise(group_by(bank.full,education),age=mean(pdays))
education_pdays
p4<-ggplot(education_pdays,aes(x=education,y=age,fill=education))+
geom_bar(stat='identity')

multiplot(p1, p2, p3, p4, cols=2)
```
```

```
ggplot(bank.full,aes(x=job,fill=job))+ geom_bar(stat='count',aes(fill =
factor(y)),position = position_dodge(width = 0.9))+theme(axis.text.x =
element_text(angle = 45, hjust = 1, vjust = 0.5))
```

```
# Boxplots (Comparing the Predictors with the Output Variable)
```{r}
par(mfrow=c(2,2))
boxplot(train.df$age ~ train.df$y,main="Age", col=c('powderblue', 'mistyrose'))
boxplot(train.df$balance ~ train.df$y,main="Balance", col=c('powderblue',
'mistyrose'))
boxplot(train.df$day ~ train.df$y,main="day", col=c('powderblue', 'mistyrose'))
boxplot(train.df$duration ~ train.df$y,main="duration", col=c('powderblue',
'mistyrose'))
boxplot(train.df$campaign ~ train.df$y,main="campaign", col=c('powderblue',
'mistyrose'))
boxplot(train.df$pdays ~ train.df$y,main="pdays", col=c('powderblue',
'mistyrose'))
boxplot(train.df$previous ~ train.df$y,main="previous", col=c('powderblue',
'mistyrose'))
```
```

Scatter Plot Between Age and other Continuous Variables

```{r}
par(mfrow=c(2,2))
plot(log(train.df$age), log(train.df$balance), main = "Age Vs Balance", xlab =
"Age", ylab = "Balance", col = 2)
abline(lm(log(train.df$balance) ~ log(train.df$age)))

plot(log(train.df$age), log(train.df$duration), main = "Age Vs Duration", xlab =
"Age", ylab = "Duration", col = 2)
abline(lm(log(train.df$duration) ~ log(train.df$age)))

plot(log(train.df$age), log(train.df$pdays), main = "Age Vs Days Past", xlab =
"Age", ylab = "Days Past", col = 2)
abline(lm(log(train.df$pdays) ~ log(train.df$age)))

plot(log(train.df$age), log(train.df$previous), main = "Age Vs Previously
Contacted", xlab = "Age", ylab = "Previously Contacted", col = 2)
abline(lm(log(train.df$previous) ~ log(train.df$age)))

plot(log(train.df$age), log(train.df$day), main = "Age Vs Day", xlab = "Age",
ylab = "Day", col = 2)
abline(lm(log(train.df$day) ~ log(train.df$age)))

plot(log(train.df$age), log(train.df$campaign), main = "Age Vs Campaign", xlab =
"Age", ylab = "Campaign", col = 2)
abline(lm(log(train.df$campaign) ~ log(train.df$age)))
```

```{r}
#install.packages("readxl")
#install.packages('psych')
normal_data <- train.df[,c(31,33,36:40)]
fa.parallel(normal_data, fm="pa", main = "Scree Plot With Parallel Analysis")
```

```{r}
pc <- principal(r = normal_data, nfactor = 3, rotate = "none")
pc
```

Performing Rotation

```{r}
pc_rotate <- principal(cont_data, nfactor = 3, rotate = "varimax")
pc_rotate
```

```{r}
pc_score <- principal(normal_data, nfactor = 3, scores = TRUE)
head(pc_score$scores)
```

```{r}
factor.plot(pc_rotate, labels = rownames(pc_rotate$loadings))
```

```
s_traindata <- t.train.df[c(7,8,12,16,17)]
s_validdata <- v.valid.df[c(7,8,12,16,17)]
s_testdata <- t.test.df[c(7,8,12,16,17)]
```

```{r}
class.tree <- rpart(y ~ .,data = s_traindata, control = rpart.control(maxdepth =
7), method = "class", minbucket = 50)
prp(class.tree, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10)
```