setwd()

tele<-read.csv("telecomfinal.csv", header=TRUE, stringsAsFactors = T, na.strings=c(“”,NA))

options(scipen = 999)

library(dplyr)

names(tele)

str(tele)

summary(tele)

##---------Creating Data Quality Report(dqr)-----------##

#Extracting Variable names

Variables<-names(tele)

dqr<-as.data.frame(Variables)

rm(Variables)

#Recording Data Type for each Variable

dqr$DataType<-sapply(tele,class)

#No. of Records for each Variable

dqr$No.ofRecords<-nrow(tele)

#Counting No. of Unique Values for each variable

for(i in 1:ncol(tele))

{

dqr$UniqueRecords[i]<-length(unique(tele[,i]))

}

#No.of observations available for each variable and its percentage

dqr$DataAvailable<-colSums(!is.na(tele))

dqr$AvailablePercentage<-round(colMeans(!is.na(tele)),4)

#Total and Percentage of Missing Values for each Variable

dqr$Missing<-colSums(is.na(tele))

dqr$MissingPercentage<-round(colMeans(is.na(tele)),4)

#Minimum, Maximum, Mean, Quantile Values for each Variable

for(i in 1:ncol(tele))

{

dqr$Minimum[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",min(tele[,i],na.rm=T),0),2)

dqr$Maximum[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",max(tele[,i],na.rm=T),0),2)

dqr$Mean[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",mean(tele[,i],na.rm=T),0),2)

dqr$fifthPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.05,na.rm=T),0),2)

dqr$tenthPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.10,na.rm=T),0),2)

dqr$twentyfifthPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.25,na.rm=T),0),2)

dqr$fiftythPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.50,na.rm=T),0),2)

dqr$seventyfifthPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.75,na.rm=T),0),2)

dqr$ninetythPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.90,na.rm=T),0),2)

dqr$ninetyfifthPercentile[i]<-round(ifelse(class(tele[,i])=="integer"|class(tele[,i])=="numeric",quantile(tele[,i],p=0.95,na.rm=T),0),2)

}

str(dqr)

#Exporting Data Quality Report

write.csv(dqr,"Data Quality Report.csv",row.names = T)

#Deleting 181 records with missing values in many variables

index<-which(is.na(tele$mou\_mean))

tele<-tele[-index,]

#Missing Value treatment of var, "retdays" and Creating Dummy Variable, removing retdays

summary(tele$retdays)

sort(unique(tele$retdays), na.last = F)

tele$retdays\_1<-ifelse(is.na(tele$retdays)==TRUE, 0, 1)

str(tele$retdays\_1)

summary(tele$retdays\_1)

tele1<-select(tele,-retdays)

tele1$retdays\_1<-as.factor(tele1$retdays\_1)

#Making dummy variables for hnd\_webcap

tele1$webcap<-ifelse(tele1$hnd\_webcap==”WC” | tele1$hnd\_webcap==”WCMB”, 1, 0)

tele1<-select(tele1,-hnd\_webcap)

tele1$webcap<-as.factor(tele1$webcap)

#Identifying those records where number of actvsubs is same as uniqsubs

Tele1$actv\_uniq\_match<-ifelse(tele1$actvsubs==tele1$uniqsubs, 1,0)

#web capability dummy variable

tele1$webcap<-ifelse(tele1$hnd\_webcap==”WC” | tele1$hnd\_webcap ==”WCMB”, 1,0)

#Missing Value treatment for income variable using Similar Case Mean Method

table1<-table(tele1$income, tele1$churn)

table1

churn\_rate<-table1[,2]/rowSums(table1)

churn\_rate

ind1<-which(is.na(tele2$income)

tele1%>%mutate(quantile=ntile(income, 10))%>%group\_by(churn, quantile)%>%summarize(N=n())%>%filter(churn==1)->dat

tele1%>%mutate(quantile=ntile(income,10))%>%group\_by(quantile)%>%summarize(N=n())->dat1

dat$percentage<-dat$N/dat1$N

#churn rate in NA group is closest to churn rate in quantile 4. So replacing missing values in income with 4.

# Mean Imputation for missing values

tele1$avg6mou[is.na(tele1$avg6mou)]<-mean(tele1$avg6mou,na.rm = T)

tele1$avg6qty[is.na(tele1$avg6qty)]<-mean(tele1$avg6qty,na.rm = T)

tele1$hnd\_price[is.na(tele1$hnd\_price)]<-mean(tele1$hnd\_price,na.rm = T)

tele1$change\_mou[is.na(tele1$change\_mou)]<-mean(tele1$change\_mou,na.rm = T)

#finding variables with more than 15% missing values and removing them maybe because they are #not proving significant after cross-tabulations for churn or have many missing info values

#Removing variables: csa, children, car\_buy, cartype, mailresp, proptype, recv\_sms\_mean, #roam\_man, wrkwoman, truck, numbcars, mtrcycle, opk\_dat\_mean, occu1, dwllsize, mailordr, #forgntvl, dwlltype, age2, models, area, custcare\_mean, ccrndmou\_range, callwait\_range, #callwait\_mean, da\_mean, da\_range, prizm\_social\_once, div\_type

colMeans(is.na(tele1))

tele1<-tele1[,colMeans(is.na(tele1)<=0.15]

#Variable drop\_blk\_Mean is created by adding vars blck\_dat\_Mean + BLCK\_VCE\_MEAN + DROP\_DAT\_MEAN + DROP\_VCE\_MEAN

# So omitting Variable blck\_dat\_Mean

tele1<-select(tele1,-blck\_dat\_Mean)

#Data transformation

tele1$percentage<-(tele1$comp\_vce\_mean+tele1$comp\_dat\_mean)/(tele1$plcd\_vce\_mean+tele1$plcd\_dat\_mean)

tele1$actv\_uniq\_match<-ifelse(tele1$actvsubs==tele1$uniqsubs, 1,0)

tele1$actv\_uniq\_match<-as.factor(tele1$actv\_uniq\_match)

#Identifying Outliers for numerical variables and deleting them

quantile(tele1$mou\_mean, p=c(1:100)/100)

tele1%>%filter(mou\_mean<=2471.76)->tele1

quantile(tele1$totmrc\_Mean, p=c(1:100)/100)

tele1%>%filter(totmrc\_Mean<=219.99)->tele1

quantile(tele1$change\_mou, p=c(1:100)/100)

tele1%>%filter(change\_mou<=2777.5)->tele1

quantile(tele1$totcalls, p=c(1:100)/100)

tele1%>%filter(totcalls<=10089)->tele1

quantile(tele1$eqpdays, p=c(1:100)/100)

tele1%>%filter(eqpdays<=1823)->tele1

quantile(tele1$adjqty, p=c(1:100)/100)

tele1%>%filter(adjqty<=9575)->tele1

quantile(tele1$adjmou, p=c(1:100)/100)

tele1%>%filter(adjmou<=27389)->tele1

#Creating buckets for continuous variables according to percentiles

#Totmrc\_Mean

Tele1$cost\_group[tele1$totmrc\_Mean <=30] <- “Low”

Tele1$cost\_group[tele1$totmrc\_Mean >30 & tele1$totmrc\_Mean <=44.99 ] <- “Med”

Tele1$cost\_group[tele1$totmrc\_Mean >44.99 & tele1$totmrc\_Mean <=59.99 ] <- “super high”

Tele1$cost\_group[tele1$totmrc\_Mean > 59.99 ] <- “very high”

#Totrev

Tele1$totrev\_bkt[tele1$totrev <=502] <- “Low”

Tele1$totrev\_bkt[tele1$totrev >502 & tele1$totrev <=784 ] <- “Med”

Tele1$totrev\_bkt[tele1$totrev >784 & tele1$totrev <=1232 ] <- “super high”

Tele1$totrev\_bkt[tele1$totrev > 1232 ] <- “very high”

#Avgrev

Tele1$avgrev\_bkt[tele1$avgrev <=36] <- “Low”

Tele1$avgrev\_bkt[tele1$avgrev >36 & tele1$avgrev <=49 ] <- “Med”

Tele1$avgrev\_bkt[tele1$avgrev >49 & tele1$avgrev <=68 ] <- “super high”

Tele1$avgrev\_bkt[tele1$avgrev > 68 ] <- “very high”

#Adjrev

Tele1$adjrev\_bkt[tele1$adjrev <=429] <- “Low”

Tele1$adjrev\_bkt[tele1$adjrev >429 & tele1$adjrev <=706 ] <- “Med”

Tele1$adjrev\_bkt[tele1$adjrev >706 & tele1$adjrev <=1125 ] <- “super high”

Tele1$adjrev\_bkt[tele1$adjrev > 1125 ] <- “very high”

#Adjmou

Tele1$adjmou\_bkt[tele1$adjmou <=2215] <- “Low”

Tele1$adjmou\_bkt[tele1$adjmou >2215 & tele1$adjmou <=5756] <- “Med”

Tele1$adjmou\_bkt[tele1$adjmou >5756& tele1$adjmou <=8055] <- “super high”

Tele1$adjmou\_bkt[tele1$adjmou >8055]<- “very high”

#Avgqty

Tele1$avgqty\_bkt[tele1$avgqty <=61] <- “Low”

Tele1$avgqty\_bkt[tele1$avgqty >61 & tele1$avgqty <=119 ] <- “Med”

Tele1$avgqty\_bkt[tele1$avgqty >119 & tele1$avgqty <=208 ] <- “super high”

Tele1$avgqty\_bkt[tele1$avgqty > 208 ] <- “very high”

#Avg3qty

Tele1$avg3qty\_bkt[tele1$avg3qty <=53] <- “Low”

Tele1$avg3qty\_bkt[tele1$avg3qty >53 & tele1$avg3qty <=118 ] <- “Med”

Tele1$avg3qty\_bkt[tele1$avg3qty >118 & tele1$avg3qty <=215 ] <- “super high”

Tele1$avg3qty\_bkt[tele1$avg3qty > 215 ] <- “very high”

#Avgmou

Tele1$avgmou\_bkt[tele1$avgmou <=164] <- “Low”

Tele1$avgmou\_bkt[tele1$avgmou >164 & tele1$avgmou <=330 ] <- “Med”

Tele1$avgmou\_bkt[tele1$avgmou >330 & tele1$avgmou <=574 ] <- “super high”

Tele1$avgmou\_bkt[tele1$avgmou > 574 ] <- “very high”

#Avg3mou

Tele1$avg3mou\_bkt[tele1$avg3mou <=144] <- “Low”

Tele1$avg3mou\_bkt[tele1$avg3mou >144 & tele1$avg3mou <=329 ] <- “Med”

Tele1$avg3mou\_bkt[tele1$avg3mou >329 & tele1$avg3mou <=612 ] <- “super high”

Tele1$avg3mou\_bkt[tele1$avg3mou > 612 ] <- “very high”

#Ovrmou\_mean

#mean is 29

Tele3$ovr\_mou<-ifelse(tele3$ovrmou\_mean >0 & tele3$ovrmou\_mean <29, “avg\_ovr\_mou”, “high\_ovr\_mou)

Tele1$ovr\_mou<-as.factor(tele1$ovr\_mou)

#Rev\_mean

Tele1$rev\_mean\_bkt[tele1$rev\_Mean <=33] <- “Low”

Tele1$rev\_mean\_bkt[tele1$rev\_Mean >33 & tele1$rev\_Mean <=45 ] <- “Med”

Tele1$rev\_mean\_bkt[tele1$rev\_Mean >45 & tele1$rev\_Mean <=63 ] <- “super high”

Tele1$rev\_mean\_bkt[tele1$rev\_Mean > 63 ] <- “very high”

#Ovrrev\_mean

#mean is 29

Tele1$ovr\_rev<-ifelse(tele1$ovrrev\_mean >0 & tele1$ovrrev\_mean <=10, “avg\_ovr\_rev”, “high\_ovr\_rev)

Tele1$ovr\_rev<-as.factor(tele1$ovr\_rev)

#Adjqty

Tele1$adjqty\_bkt[tele1$adjqty <=773] <- “Low”

Tele1$adjqty\_bkt[tele1$adjqty >773 & tele1$adjqty <=2099 ] <- “Med”

Tele1$adjqty\_bkt[tele1$adjqty >2099 & tele1$adjqty <=2891 ] <- “super high”

Tele1$adjqty\_bkt[tele1$adjqty > 2891 ] <- “very high”

#Eqpdays

Tele1$eqpdays\_bkt[tele1$eqpdays <=365]<-“one\_year”

Tele1$eqpdays\_bkt[tele1$eqpdays >365 & tele1$eqpdays <=730]<-“two\_year”

Tele1$eqpdays\_bkt[tele1$eqpdays > 730]<-“old\_phone”

Tele1$eqpdays\_bkt<-as.factor(tele1$eqpdays\_bkt)

#Totcalls

Tele1$totcalls\_bkt[tele1$totcalls <=789] <- “Low”

Tele1$totcalls\_bkt[tele1$totcalls >789 & tele1$totcalls <=1590 ] <- “Med”

Tele1$totcalls\_bkt[tele1$totcalls >1590 & tele1$totcalls <=2931 ] <- “super high”

Tele1$totcalls\_bkt[tele1$totcalls > 2931 ] <- “very high”

#months

Tele1$months\_bkt[tele1$months <=365]<-“one\_year\_customer”

Tele1$months\_bkt[tele1$months >365 & tele1$months <=730]<-“two\_year\_customer”

Tele1$months\_bkt[tele1$months > 730]<-“very\_loyal\_customer”

Tele1$months\_bkt<- as.factor(tele1$months\_bkt)

#age1

Tele1$age\_group[tele1$age1 <=30]<-“young”

Tele1$age\_group[tele1$age1 >30 & tele1$age1 <=60]<-“middle\_age”

Tele1$age\_group[tele1$age1 > 60]<-“senior”

Tele1$age\_group<- as.factor(tele1$age\_group)

#income

tele1$income\_bkt[tele1$income <=4]<-“low\_income”

tele1$income\_bkt[tele1$income >4 & tele1$income <=5.4]<-“mid\_income”

tele1$income\_bkt[tele1$income >5.4 & tele1$income <=7]<-“super\_high\_income”

tele1$income\_bkt[tele1$income >7]<-“very\_high\_income”

tele1$income\_bkt <-as.factor(tele1$income\_bkt)

#preparing final data by excluding customer\_id and variables that have been transformed

Tele3<-tele1[ , -c(2,4,8:11,13:22,27:29,31:35)]

#Random sampling

set.seed(100)

index<-sample(nrow(tele3),0.70\*nrow(tele3),replace=F)

train<-tele3[index,]

test<-tele3[-index,]

#Checking Churn Rate; if sampling is done correctly, then we will find similar churn rate in #train and test dataset.

table(tele3$churn)/nrow(tele3)

table(train$churn)/nrow(train)

table(test$churn)/nrow(test)

# Building Logistic Regression Model after excluding var "Customer\_ID"

mod<-glm(churn~.,data=train,family="binomial")

summary(mod)

#Some variables are not significant. Running many iterations by removing insignificant #variables.

mod8<-glm(churn~mou\_Mean+change\_mou+drop\_vce\_Range+owylis\_vce\_Range+mou\_opkv\_Range+iwylis\_vce\_Mean+asl\_flag+refurb\_new+marital+webcap+percentage+actv\_uniq\_match+age\_group+cost\_group+adjmou\_bkt+avgmou\_bkt+avg3mou\_bkt+rev\_mean\_bkt+adjqty\_bkt+ovr\_rev, data=train, family=”binomial”)

#All variables are significant in mod8

# Checking For Multicollinearity

library(car)

vif(mod8)

## \*\*\*\*\* Model Testing \*\*\*\*\* ##

#Predicted Values ==> Predicting the probability of a customer churning.

predicted<-mod8$fitted.values

#Assuming cut-off probablity as per the churn rate in data set

table(tele3$churn)/nrow(tele3)

predbkt<-ifelse(predicted>0.2422636,1,0)

#Confusion Matrix

table(predbkt, train$churn)

#ROCR Curve

library(ROCR)

pred<-prediction(predicted,train$churn)

pref<-performance(pred,"tpr","fpr")

plot(pref,col="red")

auc<-performance(pred,"auc")

auc<-unlist(slot(auc,"y.values"))

auc

# The auc is 0. 6413305 which is more than 0.50.

# So the model seems to be better.

#Gains Chart

library(gains)

gains(test$churn,predict(mod8,type="response",newdata=test),groups = 10)

#the Gains Chart shows that the top 30% of the probabilities contain 41.3% customers that are likely to churn.

test$prob<-predict(mod8,type="response",newdata=test)

quantile(test$prob,prob=c(0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1))

#Top 40% of the probability scores lie between 0.2166985 and 0.2222167

#We can use this probablity to extract the data of customers who are highly likely to churn.

Q.1 What are the top five factors driving likelihood of churn at Mobicom?

head(sort(abs(mod8$coefficients),decreasing = T),10)

summary(mod8)

Q.2 Validation of survey findings.

# a) Whether "cost and billing" and "network and service quality" are important factors #influencing churn behaviour.

#Cost is analyzed by analyzing monthly recurring charge (MRC). The chart indicates that as #the monthly recurring charge of a customer increases, the churn rate drops steadily. One of #the reasons for this could be that higher MRC points to more number of family #members in the same plan which makes it more difficult to churn.

a<-table(tele3$cost\_group, tele3$churn)

0 1

Low 10203 4131

Med 17664 5343

Super high 11743 3605

Very high 9301 2327

round(a[,2]/rowSums(a),2)

Low Med Super high very high

0.29 0.23 0.23 0.20

#According to the logistic regression model, billing adjusted total minutes of use over #the life of the customer(Adjmou) is also significant factor influencing churn. Mean monthly #revenue (charge amount) (Rev\_mean ) is also significant in the model.

#Q.3 “network and service quality” as important factors influencing churn

#Comp\_vce\_mean, comp\_dat\_mean variable captures mean number of completed voice #and data calls. Plcd\_vce\_mean, plcd\_dat\_mean variable captures data on the mean #number of attempted voice and data calls placed.

#We can look at the ratio of comp\_vce\_Mean + comp\_dat\_mean and plcd\_vce\_Mean + #plcd\_dat\_mean.

#This ratio is negatively correlated to churn. A low ratio can be an indicator of network #issues. The beta coefficient for percentage variable in the model explains that if we keep #everything else constant and we look only at the change in this ratio, for every unit increase, #the probability of churn increases by 0.156734 units.

Q.4 Are data usage connectivity issues turning out to be costly? In other words, is it leading to churn?

#Data usage connectivity issues can be captured by looking into the mean number of #dropped or blocked calls. If the rate is high, then the customer will be unhappy resulting #into churn.

#Drop\_blk\_mean variable is significant in the model.

#Yet the data analysis done in the Data Quality Report shows that only 10% to 15% #customers are actually making data calls.

#Mobicom should focus on improving network connectivity and service to provide maximum #customer satisfaction in order to control the churn rate from the aspect of dropped data #calls.

Q.5 Would you recommend rate plan migration as a proactive retention strategy?

#Ovrrev\_mean\_log and data\_overage variables are significant in the model.

#The beta coefficient of ovrrev\_Mean\_log is 0.13 which means that a unit increase in #ovrrev\_mean will result is increase in churn rate by 13%.

#The beta coefficient of data\_overrage is 0.056 which means that with a unit increase in #data\_overage, the probability of churn rate increases by almost 6%.

#If the customer incurs overage charges consistently, either because of data overage or #minutes overage, then such customers should be migrated to a different rate plan as a #proactive retention strategy.

#Mean overage revenue is the sum of data and voice overage revenues.

#Q.6 What would be your recommendation on how to use this churn model for prioritization #of customers for a proactive retention campaigns in the future?

library(gains)

gains(test$churn,predict(mod8,type="response",newdata=test),groups = 10)

#the Gains Chart shows that the top 20% of the probabilities contain 29.3% customers that #are highly likely to churn.

# Selecting Customers with high churn rate

test$prob<-predict(mod8,type="response",newdata=test)

quantile(test$prob,prob=c(0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1))

# Top 20% of the probabilities lie between 0.3403818 and 0.7138657

# Applying cutoff value to predict customers who Will Churn

pred4<-predict(mod8, type="response", newdata=test)

pred4<-ifelse(pred4>=0.3403818 , 1, 0)

table(pred4,test$churn)

Targeted<-test[test$prob>0.3403818 & test$prob<=0.7138657 & test$churn=="1","Customer\_ID"]

Targeted<-as.data.frame(Targeted)

nrow(Targeted)

write.csv(Targeted,"Target\_Customers.csv",row.names = F)

#Q.7 What would be the target segments for proactive retention campaigns?

# Falling ARPU forecast is also a concern and therefore, Mobicom would like to save their #high revenue customers besides managing churn. Given a budget constraint of a contact #list of 20% of the subscriber pool, which subscribers should prioritized if "revenue saves" is #also a priority besides controlling churn. In other words, controlling churn is the primary #objective and revenue saves is the secondary objective.

pred5<-predict(mod8, type="response", newdata=test)

test$prob<-predict(mod8,type="response",newdata=test)

quantile(test$prob,prob=c(0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1))

pred6<-ifelse(pred5<0.20,"Low\_Score",ifelse(pred5>=0.20 & pred5<0.30,"Medium\_Score","High\_Score"))

table(pred6,test$churn)

str(test$totrev)

quantile(test$totrev,prob=c(0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1))

Revenue\_Levels<-ifelse(test$totrev<634,"Low\_Revenue",ifelse(test$totrev>=634 & test$totrev<1036,"Medium\_Revenue","High\_Revenue"))

table(Revenue\_Levels)

table(pred6,Revenue\_Levels)

## Thus this table can be used to select the levels of customers are to be targeted and the #Target list can be extracted as follows:

test$prob\_levels<-ifelse(pred5<0.20,"Low\_Score",ifelse(pred5>=0.20 & pred5<0.30,"Medium\_Score","High\_Score"))

test$Revenue\_Levels<-ifelse(test$totrev<670.660,"Low\_Revenue",ifelse(test$totrev>=634 & test$totrev<1036,"Medium\_Revenue","High\_Revenue"))

Targeted1<-test[test$prob\_levels=="High\_Score" & test$Revenue\_Levels=="High\_Revenue","Customer\_ID"]

Targeted1<-as.data.frame(Targeted1)

nrow(Targeted1)

write.csv(Targeted1,"High\_Revenue\_Target\_Customers.csv",row.names = F)