MDDS 6372 Group Project2 : Bank Term Deposit Classification

#Title: "MSDS 6372 Group Project 2: Bank Project - Predicting if a customer will subscribe to a term deposit."  
#Group Members: Nicole Norelli,Nneanna Okpara,Sowmya Mani  
#Date: March 25 2021  
  
#Introduction: This Project is about the Bank Market Analysis to predict if a customer will subscribe to a term deposit  
  
#The data set used for this analysis consists of Bank Full Data:  
#Citation:  
#This dataset is public available for research. The details are described in [Moro et al., 2011].   
#[Moro et al., 2011] S. Moro, R. Laureano and P. Cortez. Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology.   
#In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimarães, Portugal, October, 2011. EUROSIS.  
#Available at: [pdf] http://hdl.handle.net/1822/14838  
# [bib] http://www3.dsi.uminho.pt/pcortez/bib/2011-esm-1.txt  
#Bank Full.csv and Bank.csv  
  
##### Bank client data #####  
#1 - age (numeric)  
#2 - job : type of job (categorical:"admin.","unknown","unemployed","management","housemaid","entrepreneur","student",  
# "blue-collar","self-employed","retired","technician","services")   
#3 - marital : marital status (categorical: "married","divorced","single"; note: "divorced" means divorced or widowed)  
#4 - education (categorical: "unknown","secondary","primary","tertiary")  
#5 - default: has credit in default? (binary: "yes","no")  
#6 - balance: average yearly balance, in euros (numeric)   
#7 - housing: has housing loan? (binary: "yes","no")  
#8 - loan: has personal loan? (binary: "yes","no")  
##### related with the last contact of the current campaign #####  
#9 - contact: contact communication type (categorical: "unknown","telephone","cellular")   
#10 - day: last contact day of the month (numeric)  
#11 - month: last contact month of year (categorical: "jan", "feb", "mar", ..., "nov", "dec")  
#12 - duration: last contact duration, in seconds (numeric)  
##### other attributes #####  
#13 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)  
#14 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means client was not previously contacted)  
#15 - previous: number of contacts performed before this campaign and for this client (numeric)  
#16 - poutcome: outcome of the previous marketing campaign (categorical: "unknown","other","failure","success")  
  
##### Output variable (desired target) #####  
#17 - y - has the client subscribed a term deposit? (binary: "yes","no")  
  
#plot(model.name)

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

#Libraries loaded for the ANalysis  
library(XML)

## Warning: package 'XML' was built under R version 4.0.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.0.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(RCurl)

## Warning: package 'RCurl' was built under R version 4.0.3

library(httr)

## Warning: package 'httr' was built under R version 4.0.3

library(jsonlite)

## Warning: package 'jsonlite' was built under R version 4.0.3

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.0.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.4 v stringr 1.4.0  
## v tidyr 1.1.2 v forcats 0.5.0  
## v readr 1.4.0

## Warning: package 'ggplot2' was built under R version 4.0.4

## Warning: package 'tibble' was built under R version 4.0.3

## Warning: package 'tidyr' was built under R version 4.0.3

## Warning: package 'readr' was built under R version 4.0.3

## Warning: package 'purrr' was built under R version 4.0.3

## Warning: package 'stringr' was built under R version 4.0.3

## Warning: package 'forcats' was built under R version 4.0.3

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x tidyr::complete() masks RCurl::complete()  
## x dplyr::filter() masks stats::filter()  
## x purrr::flatten() masks jsonlite::flatten()  
## x dplyr::lag() masks stats::lag()

library(naniar)

## Warning: package 'naniar' was built under R version 4.0.3

library(GGally)

## Warning: package 'GGally' was built under R version 4.0.3

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(ggplot2)  
library(class)  
library(caret)

## Warning: package 'caret' was built under R version 4.0.3

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

## The following object is masked from 'package:httr':  
##   
## progress

library(knnp)

## Warning: package 'knnp' was built under R version 4.0.3

##   
## Attaching package: 'knnp'

## The following object is masked from 'package:class':  
##   
## knn

library(e1071)

## Warning: package 'e1071' was built under R version 4.0.3

library(ggplot2)  
library(maps)

## Warning: package 'maps' was built under R version 4.0.3

##   
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':  
##   
## map

library(dplyr)  
library(mapproj)

## Warning: package 'mapproj' was built under R version 4.0.3

library(ggplot2)  
library(dplyr)  
library(ggcorrplot)

## Warning: package 'ggcorrplot' was built under R version 4.0.3

library(GGally)  
library(viridis)

## Warning: package 'viridis' was built under R version 4.0.3

## Loading required package: viridisLite

library(gplots)

## Warning: package 'gplots' was built under R version 4.0.4

##   
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':  
##   
## lowess

library(leaps)

## Warning: package 'leaps' was built under R version 4.0.3

library(matrixStats)

##   
## Attaching package: 'matrixStats'

## The following object is masked from 'package:dplyr':  
##   
## count

library(ResourceSelection)

## Warning: package 'ResourceSelection' was built under R version 4.0.4

## ResourceSelection 0.3-5 2019-07-22

library(MASS)

## Warning: package 'MASS' was built under R version 4.0.3

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(glmnet)

## Warning: package 'glmnet' was built under R version 4.0.3

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.0.3

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

## Loaded glmnet 4.1

library(ROCR)

## Warning: package 'ROCR' was built under R version 4.0.4

library(randomForest)

## Warning: package 'randomForest' was built under R version 4.0.4

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':  
##   
## margin

## The following object is masked from 'package:dplyr':  
##   
## combine

library(tidyverse)  
library(dplyr)  
library(naniar)  
library(GGally)  
library(ggplot2)  
library(class)  
library(caret)  
library(e1071)  
library(ggcorrplot)  
library(magrittr)

## Warning: package 'magrittr' was built under R version 4.0.3

##   
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':  
##   
## set\_names

## The following object is masked from 'package:tidyr':  
##   
## extract

library(dplyr)  
library(tidyr)  
library(naniar)  
library(ggplot2)  
library(plotly)

## Warning: package 'plotly' was built under R version 4.0.4

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:MASS':  
##   
## select

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:httr':  
##   
## config

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library(forcats)  
library(GGally)  
library(ggcorrplot)  
library(car)

## Warning: package 'car' was built under R version 4.0.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.0.3

##   
## Attaching package: 'car'

## The following object is masked from 'package:purrr':  
##   
## some

## The following object is masked from 'package:dplyr':  
##   
## recode

library(ISLR)

## Warning: package 'ISLR' was built under R version 4.0.3

library(leaps)  
library(matrixStats)  
library(leaps)  
library(olsrr)

## Warning: package 'olsrr' was built under R version 4.0.3

##   
## Attaching package: 'olsrr'

## The following object is masked from 'package:MASS':  
##   
## cement

## The following object is masked from 'package:datasets':  
##   
## rivers

library(OLScurve)

## Warning: package 'OLScurve' was built under R version 4.0.3

library(glmnet)  
  
#Import the Bank Full Data  
Bank\_Full<-read.csv('C:/Sowmya/SMU/03\_Applied Stats/Group Project 2/bank-full.csv' ,sep=";")  
  
#Quick Peek at the SUmmary data of the available dataset  
summary(Bank\_Full)

## age job marital education   
## Min. :18.00 Length:45211 Length:45211 Length:45211   
## 1st Qu.:33.00 Class :character Class :character Class :character   
## Median :39.00 Mode :character Mode :character Mode :character   
## Mean :40.94   
## 3rd Qu.:48.00   
## Max. :95.00   
## default balance housing loan   
## Length:45211 Min. : -8019 Length:45211 Length:45211   
## Class :character 1st Qu.: 72 Class :character Class :character   
## Mode :character Median : 448 Mode :character Mode :character   
## Mean : 1362   
## 3rd Qu.: 1428   
## Max. :102127   
## contact day month duration   
## Length:45211 Min. : 1.00 Length:45211 Min. : 0.0   
## Class :character 1st Qu.: 8.00 Class :character 1st Qu.: 103.0   
## Mode :character Median :16.00 Mode :character Median : 180.0   
## Mean :15.81 Mean : 258.2   
## 3rd Qu.:21.00 3rd Qu.: 319.0   
## Max. :31.00 Max. :4918.0   
## campaign pdays previous poutcome   
## Min. : 1.000 Min. : -1.0 Min. : 0.0000 Length:45211   
## 1st Qu.: 1.000 1st Qu.: -1.0 1st Qu.: 0.0000 Class :character   
## Median : 2.000 Median : -1.0 Median : 0.0000 Mode :character   
## Mean : 2.764 Mean : 40.2 Mean : 0.5803   
## 3rd Qu.: 3.000 3rd Qu.: -1.0 3rd Qu.: 0.0000   
## Max. :63.000 Max. :871.0 Max. :275.0000   
## y   
## Length:45211   
## Class :character   
## Mode :character   
##   
##   
##

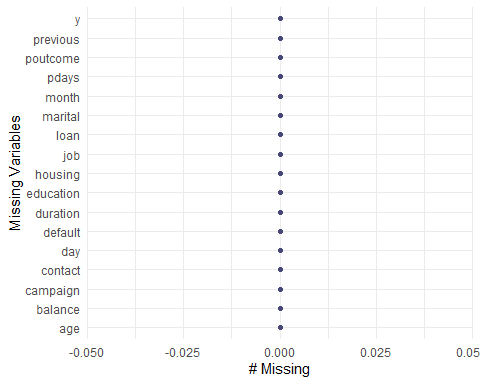
str(Bank\_Full)

## 'data.frame': 45211 obs. of 17 variables:  
## $ age : int 58 44 33 47 33 35 28 42 58 43 ...  
## $ job : chr "management" "technician" "entrepreneur" "blue-collar" ...  
## $ marital : chr "married" "single" "married" "married" ...  
## $ education: chr "tertiary" "secondary" "secondary" "unknown" ...  
## $ default : chr "no" "no" "no" "no" ...  
## $ balance : int 2143 29 2 1506 1 231 447 2 121 593 ...  
## $ housing : chr "yes" "yes" "yes" "yes" ...  
## $ loan : chr "no" "no" "yes" "no" ...  
## $ contact : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ day : int 5 5 5 5 5 5 5 5 5 5 ...  
## $ month : chr "may" "may" "may" "may" ...  
## $ duration : int 261 151 76 92 198 139 217 380 50 55 ...  
## $ campaign : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ pdays : int -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...  
## $ previous : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ poutcome : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ y : chr "no" "no" "no" "no" ...

#Checking for Missing Data  
sapply(Bank\_Full,function(x) sum(is.na(x)))

## age job marital education default balance housing loan   
## 0 0 0 0 0 0 0 0   
## contact day month duration campaign pdays previous poutcome   
## 0 0 0 0 0 0 0 0   
## y   
## 0

gg\_miss\_var(Bank\_Full)+xlab("Missing Variables")



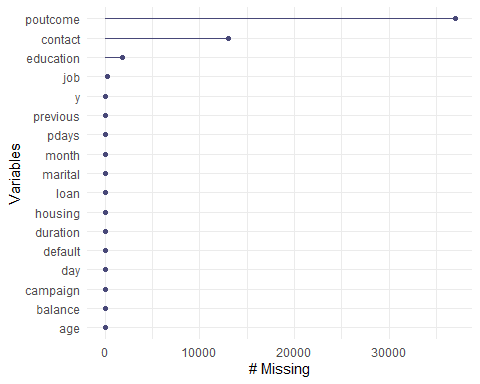
#The Bank dataset has 45,211 observations with 17 variables providing more information on the Bank Clients.There is no missing data in the data set  
  
# missing values using other than NA (Find the Unknowns in the data)  
sapply(Bank\_Full, function(x) sum(x %in% 'unknown'))

## age job marital education default balance housing loan   
## 0 288 0 1857 0 0 0 0   
## contact day month duration campaign pdays previous poutcome   
## 13020 0 0 0 0 0 0 36959   
## y   
## 0

#We find 288 unknowns in the job category, 1,857 in education, 13,020 in contact, 36,959 in poutcome  
  
Bank\_Full\_NA = Bank\_Full %>%  
 dplyr::na\_if('unknown')  
sapply(Bank\_Full\_NA, function(x) sum(is.na(x)))

## age job marital education default balance housing loan   
## 0 288 0 1857 0 0 0 0   
## contact day month duration campaign pdays previous poutcome   
## 13020 0 0 0 0 0 0 36959   
## y   
## 0

#Change unknown to NA   
gg\_miss\_var(Bank\_Full\_NA)



# This is interesting:   
# Contact (contact communication type) is missing in a block from May 5 to partway through July 4, so it's not random  
# Similar issue with poutcome (outcome of previous marketing campaign). Values start to fill in around late October. Of course there's a lot more missing in this one.  
# Another thing to note: Because this data spans 2+ years (it seems to be in calendar order), It would be better if we should put a year variable in to keep track...  
summary(Bank\_Full\_NA)

## age job marital education   
## Min. :18.00 Length:45211 Length:45211 Length:45211   
## 1st Qu.:33.00 Class :character Class :character Class :character   
## Median :39.00 Mode :character Mode :character Mode :character   
## Mean :40.94   
## 3rd Qu.:48.00   
## Max. :95.00   
## default balance housing loan   
## Length:45211 Min. : -8019 Length:45211 Length:45211   
## Class :character 1st Qu.: 72 Class :character Class :character   
## Mode :character Median : 448 Mode :character Mode :character   
## Mean : 1362   
## 3rd Qu.: 1428   
## Max. :102127   
## contact day month duration   
## Length:45211 Min. : 1.00 Length:45211 Min. : 0.0   
## Class :character 1st Qu.: 8.00 Class :character 1st Qu.: 103.0   
## Mode :character Median :16.00 Mode :character Median : 180.0   
## Mean :15.81 Mean : 258.2   
## 3rd Qu.:21.00 3rd Qu.: 319.0   
## Max. :31.00 Max. :4918.0   
## campaign pdays previous poutcome   
## Min. : 1.000 Min. : -1.0 Min. : 0.0000 Length:45211   
## 1st Qu.: 1.000 1st Qu.: -1.0 1st Qu.: 0.0000 Class :character   
## Median : 2.000 Median : -1.0 Median : 0.0000 Mode :character   
## Mean : 2.764 Mean : 40.2 Mean : 0.5803   
## 3rd Qu.: 3.000 3rd Qu.: -1.0 3rd Qu.: 0.0000   
## Max. :63.000 Max. :871.0 Max. :275.0000   
## y   
## Length:45211   
## Class :character   
## Mode :character   
##   
##   
##

# Note: pdays: -1 means not previously contacted.Need to see how this can be handled

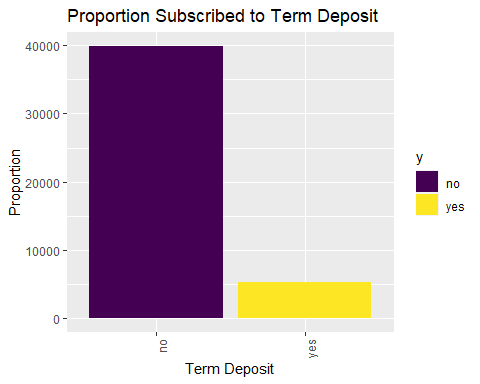
#The data seem to be sequentially ordered, adding an ID number in order  
Bank\_Full = Bank\_Full %>%  
 mutate(id = row\_number())  
# Adding a new variable to include year which could add more insights to the data  
Bank\_Full = Bank\_Full %>%  
 mutate(year = case\_when(id>=1 & id<=27729 ~ '2008',  
 id>=27730 & id<=42591 ~ '2009',  
 id>=42592 ~ '2010'))  
Bank\_Full$year = factor(Bank\_Full$year)  
  
# Let's reorder month so it makes sense on graphs  
Bank\_Full$month = factor(Bank\_Full$month, levels=c("jan","feb","mar","apr","may","jun","jul","aug","sep","oct","nov","dec"))  
  
# A look at categorical  
ggpairs(Bank\_Full,columns=c(2:5,7:9,11,16:17),aes(colour=y)) # that's a little busy, let's break it down



#Analyzing the data with plots  
str(Bank\_Full)

## 'data.frame': 45211 obs. of 19 variables:  
## $ age : int 58 44 33 47 33 35 28 42 58 43 ...  
## $ job : chr "management" "technician" "entrepreneur" "blue-collar" ...  
## $ marital : chr "married" "single" "married" "married" ...  
## $ education: chr "tertiary" "secondary" "secondary" "unknown" ...  
## $ default : chr "no" "no" "no" "no" ...  
## $ balance : int 2143 29 2 1506 1 231 447 2 121 593 ...  
## $ housing : chr "yes" "yes" "yes" "yes" ...  
## $ loan : chr "no" "no" "yes" "no" ...  
## $ contact : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ day : int 5 5 5 5 5 5 5 5 5 5 ...  
## $ month : Factor w/ 12 levels "jan","feb","mar",..: 5 5 5 5 5 5 5 5 5 5 ...  
## $ duration : int 261 151 76 92 198 139 217 380 50 55 ...  
## $ campaign : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ pdays : int -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...  
## $ previous : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ poutcome : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ y : chr "no" "no" "no" "no" ...  
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ year : Factor w/ 3 levels "2008","2009",..: 1 1 1 1 1 1 1 1 1 1 ...

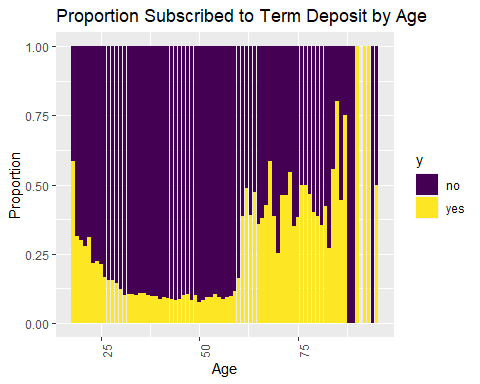
#attach(Bank\_Full)  
#Response  
Bank\_Full %>%   
 ggplot(aes(x=y, fill=y)) +   
 geom\_bar(position="stack") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Term Deposit")+  
 ggtitle("Proportion Subscribed to Term Deposit")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



#The count of Yes is 11% and No is 89% on term deposit  
  
#Bank\_Full %>% count(y)  
#No s count 39,922; Yes count 5,289   
  
# Age vs response  
t(aggregate(age~y,data=Bank\_Full,summary))

## [,1] [,2]   
## y "no" "yes"   
## age.Min. "18.00000" "18.00000"  
## age.1st Qu. "33.00000" "31.00000"  
## age.Median "39.00000" "38.00000"  
## age.Mean "40.83899" "41.67007"  
## age.3rd Qu. "48.00000" "50.00000"  
## age.Max. "95.00000" "95.00000"

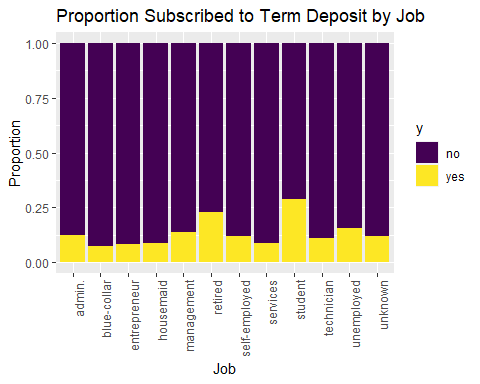
Bank\_Full %>%   
 ggplot(aes(x=age, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Age")+  
 ggtitle("Proportion Subscribed to Term Deposit by Age")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



#The older the age the higher proportions of Yes to a term deposit  
#Age and term deposit seems to be correlated.  
  
# Job vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$job),2)

##   
## admin. blue-collar entrepreneur housemaid management retired  
## no 0.87797331 0.92725031 0.91728312 0.91209677 0.86244449 0.77208481  
## yes 0.12202669 0.07274969 0.08271688 0.08790323 0.13755551 0.22791519  
##   
## self-employed services student technician unemployed unknown  
## no 0.88157061 0.91116996 0.71321962 0.88943004 0.84497314 0.88194444  
## yes 0.11842939 0.08883004 0.28678038 0.11056996 0.15502686 0.11805556

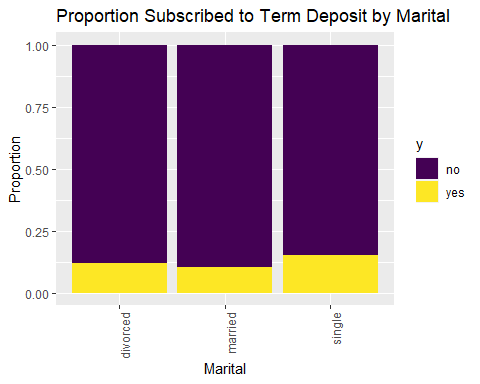
Bank\_Full %>%   
 ggplot(aes(x=job, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Job")+  
 ggtitle("Proportion Subscribed to Term Deposit by Job")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



#Looks like retired and students are on the higher proportions of Yes to a term deposit  
  
# Marital vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$marital),2)

##   
## divorced married single  
## no 0.8805454 0.8987653 0.8505082  
## yes 0.1194546 0.1012347 0.1494918

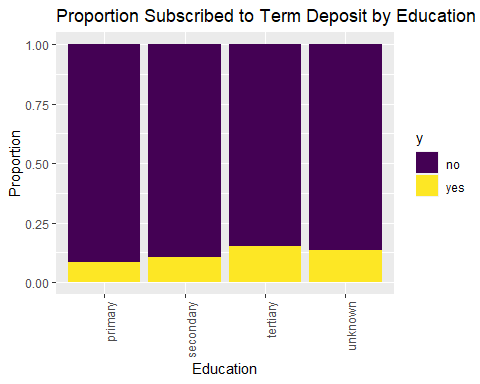
Bank\_Full %>%   
 ggplot(aes(x=marital, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") + xlab("Marital")+  
 ggtitle("Proportion Subscribed to Term Deposit by Marital")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# Slightly higher for singles  
  
# Education vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$education),2)

##   
## primary secondary tertiary unknown  
## no 0.91373522 0.89440565 0.84993610 0.86429725  
## yes 0.08626478 0.10559435 0.15006390 0.13570275

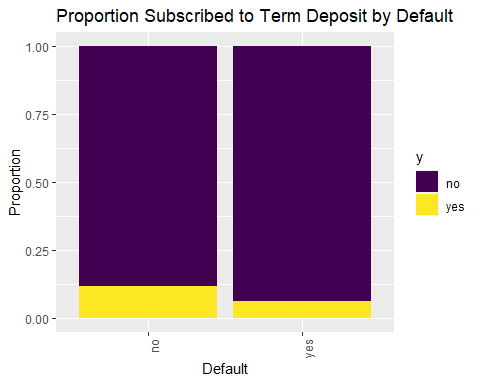
Bank\_Full %>%   
 ggplot(aes(x=education, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Education")+  
 ggtitle("Proportion Subscribed to Term Deposit by Education")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# Looks like it increases slightly as education increases  
  
# default vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$default),2)

##   
## no yes  
## no 0.88203892 0.93619632  
## yes 0.11796108 0.06380368

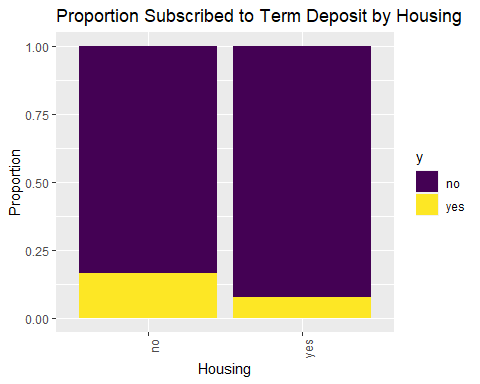
Bank\_Full %>%   
 ggplot(aes(x=default, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Default")+  
 ggtitle("Proportion Subscribed to Term Deposit by Default")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# Higher for those who have less credit, but there are very few defaults relatively, so it might not be that useful  
  
# housing vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$housing),2)

##   
## no yes  
## no 0.8329764 0.9230004  
## yes 0.1670236 0.0769996

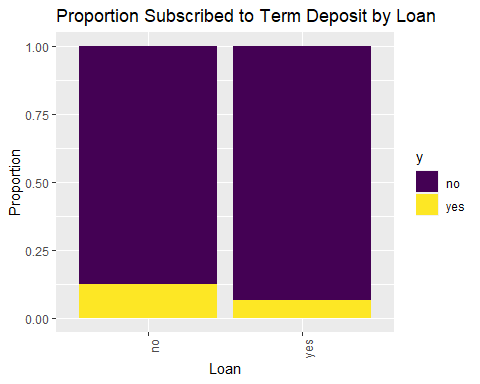
Bank\_Full %>%   
 ggplot(aes(x=housing, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Housing")+  
 ggtitle("Proportion Subscribed to Term Deposit by Housing")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# Higher for clients with no housing loans, this looks like a good variable   
  
# loan vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$loan),2)

##   
## no yes  
## no 0.87344273 0.93318609  
## yes 0.12655727 0.06681391

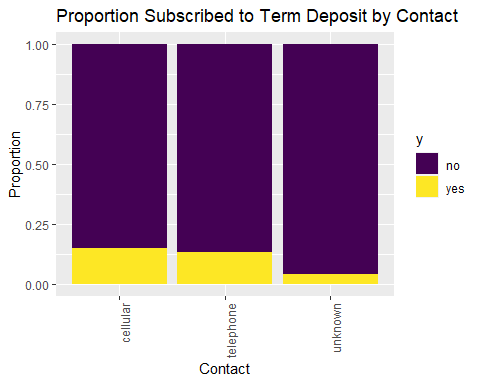
Bank\_Full %>%   
 ggplot(aes(x=loan, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Loan")+  
 ggtitle("Proportion Subscribed to Term Deposit by Loan")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# Higher for clients with no loans, although most people are in the no category for loan anyway  
  
# contact vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$contact),2)

##   
## cellular telephone unknown  
## no 0.85081100 0.86579491 0.95929339  
## yes 0.14918900 0.13420509 0.04070661

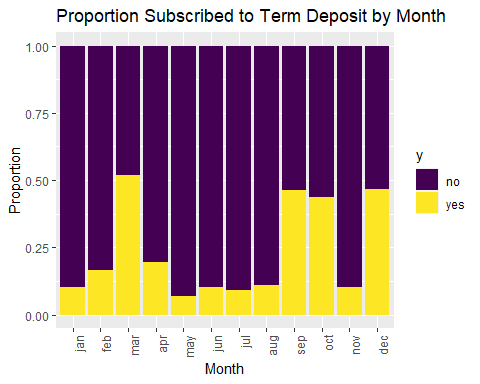
Bank\_Full %>%   
 ggplot(aes(x=contact, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Contact")+  
 ggtitle("Proportion Subscribed to Term Deposit by Contact")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



# This doesn't seem like it's going to be a good variable anyway, so maybe all the NAs won't be an issue.The clients contacted through cellular seemed to have opened a term deposit.  
  
# month vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$month),2)

##   
## jan feb mar apr may jun  
## no 0.89878831 0.83352208 0.48008386 0.80320600 0.93280546 0.89777195  
## yes 0.10121169 0.16647792 0.51991614 0.19679400 0.06719454 0.10222805  
##   
## jul aug sep oct nov dec  
## no 0.90906454 0.88986714 0.53540587 0.56233062 0.89848866 0.53271028  
## yes 0.09093546 0.11013286 0.46459413 0.43766938 0.10151134 0.46728972

Bank\_Full %>%   
 ggplot(aes(x=month, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Month")+  
 ggtitle("Proportion Subscribed to Term Deposit by Month")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



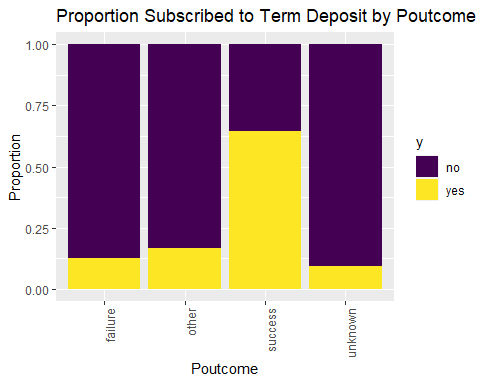
# This is interesting. Higher proportions in Dec, Mar, Oct, Sept; however, those seem to be the months with less data...  
summary(Bank\_Full$month)

## jan feb mar apr may jun jul aug sep oct nov dec   
## 1403 2649 477 2932 13766 5341 6895 6247 579 738 3970 214

# Dec only has 214, Mar has 477, Oct has 738, Sept has 579. All the other months have 1403-13766  
# also see breakdown by year  
  
# outcome vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$poutcome),2)

##   
## failure other success unknown  
## no 0.87390329 0.83315217 0.35274653 0.90838497  
## yes 0.12609671 0.16684783 0.64725347 0.09161503

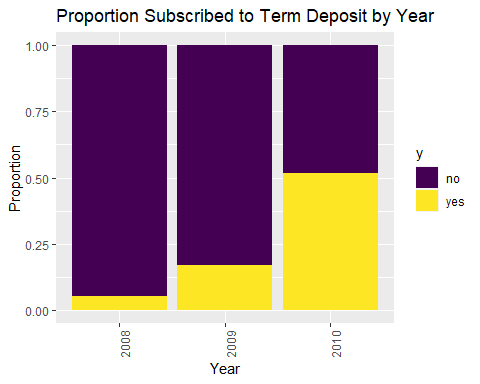
Bank\_Full %>%   
 ggplot(aes(x=poutcome, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Poutcome")+  
 ggtitle("Proportion Subscribed to Term Deposit by Poutcome")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



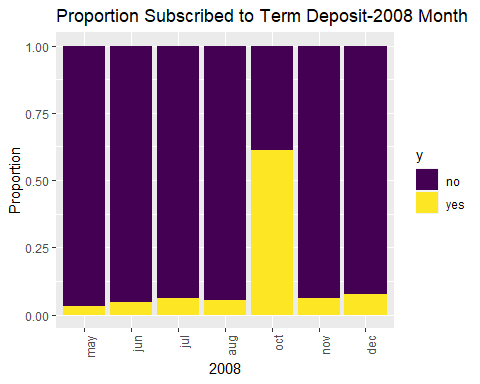
# Well, even though we have a lot of NAs, success is a really strong predictor of our outcome  
# We probably need to find a way to incorporate that.  
  
# year vs response  
prop.table(table(Bank\_Full$y,Bank\_Full$year),2)

##   
## 2008 2009 2010  
## no 0.94947528 0.82936348 0.48396947  
## yes 0.05052472 0.17063652 0.51603053

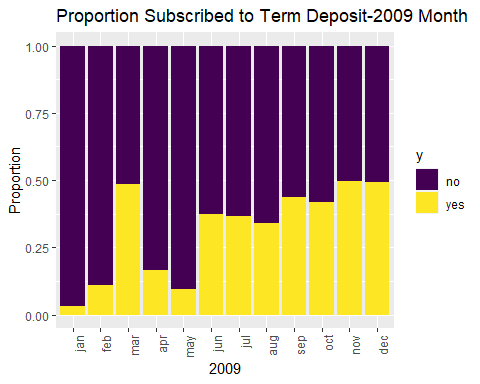
Bank\_Full %>%   
 ggplot(aes(x=year, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("Year")+  
 ggtitle("Proportion Subscribed to Term Deposit by Year")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



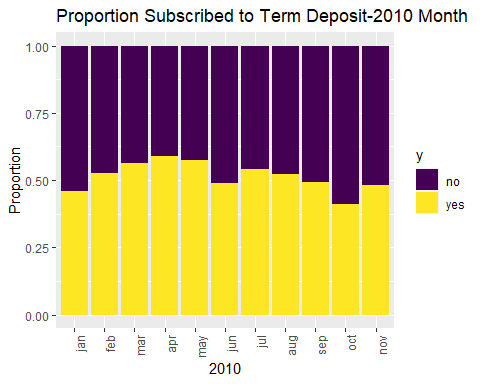
# Well this seems important.Slightly higher for 2010. There is a lot of missing values (unknowns) in 2008. This was also the financial crisis. We see lot of yes and no for 2010 are can be considered a great dataset for prediction. We will use 2008 and 2009 to understand the data but the data seems to be more clean and normal in 2010.  
  
# Break down months by year  
#Year=2008  
Bank\_Full %>%   
 filter(year == 2008) %>%  
 ggplot(aes(x=month, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("2008")+  
 ggtitle("Proportion Subscribed to Term Deposit-2008 Month")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



#Oct month shows higher on the Yes side  
  
#Year=2009  
Bank\_Full %>%   
 filter(year == 2009) %>%  
 ggplot(aes(x=month, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("2009")+  
 ggtitle("Proportion Subscribed to Term Deposit-2009 Month")+ theme(axis.text.x = element\_text(angle=90, hjust=1))



#Higher from Oct until dec  
  
#Year=2010  
Bank\_Full %>%   
 filter(year == 2010) %>%  
 ggplot(aes(x=month, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +xlab("2010")+  
 ggtitle("Proportion Subscribed to Term Deposit-2010 Month")+ theme(axis.text.x = element\_text(angle=90, hjust=1))

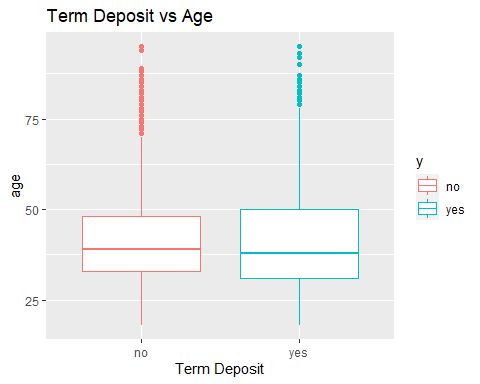


#Year 2010 seema to have had more cleints with term deposits than year 2008 and 2009. The yes and no are almost equal.The 2010 data looks better for prediction compared to 2008 and 2009.  
#The Yes on term deposit seems to be increasing with year.

# continuous variables  
# Age vs Response  
t(aggregate(age~y,data=Bank\_Full,summary))

## [,1] [,2]   
## y "no" "yes"   
## age.Min. "18.00000" "18.00000"  
## age.1st Qu. "33.00000" "31.00000"  
## age.Median "39.00000" "38.00000"  
## age.Mean "40.83899" "41.67007"  
## age.3rd Qu. "48.00000" "50.00000"  
## age.Max. "95.00000" "95.00000"

Bank\_Full %>%  
 ggplot(aes(x=y, y=age, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Age") +  
 xlab("Term Deposit")



# clients between age 35 to 50 seems to be more likely to open a term deposit  
  
# Balance vs Response  
t(aggregate(balance~y,data=Bank\_Full,summary))

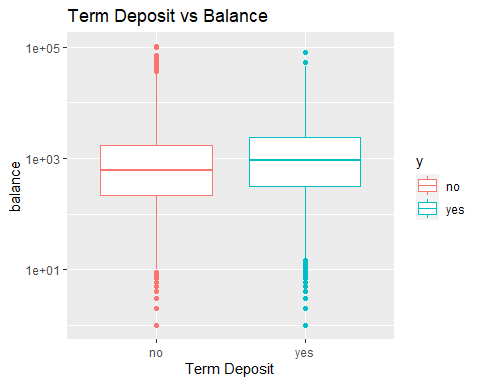
## [,1] [,2]   
## y "no" "yes"   
## balance.Min. " -8019.000" " -3058.000"  
## balance.1st Qu. " 58.000" " 210.000"  
## balance.Median " 417.000" " 733.000"  
## balance.Mean " 1303.715" " 1804.268"  
## balance.3rd Qu. " 1345.000" " 2159.000"  
## balance.Max. "102127.000" " 81204.000"

Bank\_Full %>%  
 ggplot(aes(x=y, y=balance, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Balance") +  
 xlab("Term Deposit")+scale\_y\_log10()

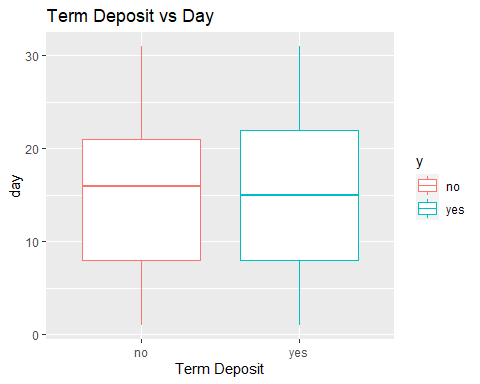
## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

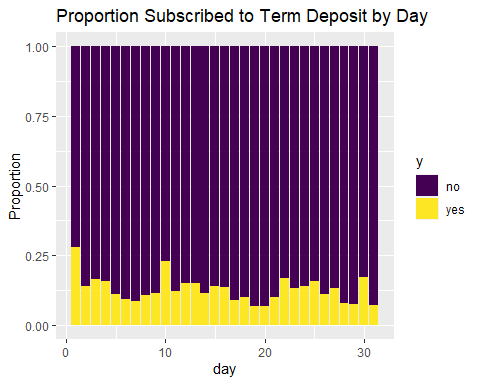
## Warning: Removed 7280 rows containing non-finite values (stat\_boxplot).



# looks like it could benefit from a transformation  
# The clients with more balance seems to be having a term deposit Yes. There seems to be a leverage point on Nos.The clients who opened a term deposit had a more balance than the ones who did not have a term deposit  
  
# Day vs Response (not sure this is the best way to look at this)  
Bank\_Full %>%  
 ggplot(aes(x=y, y=day, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Day") +  
 xlab("Term Deposit")



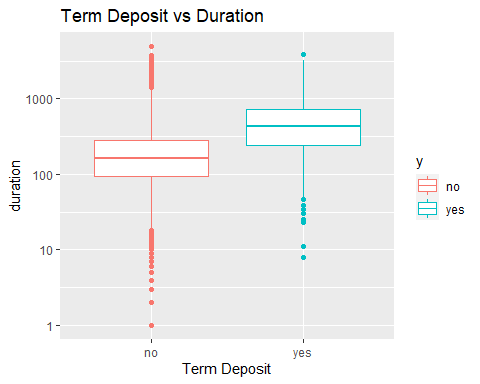
#Last Day contacted of the month seems to be almost same for those who had a opened term deposit or not.  
  
# Probably better to do this:  
Bank\_Full %>%   
 ggplot(aes(x=day, fill=y)) +   
 geom\_bar(position="fill") +   
 scale\_fill\_viridis\_d() +  
 ylab("Proportion") +  
 ggtitle("Proportion Subscribed to Term Deposit by Day")



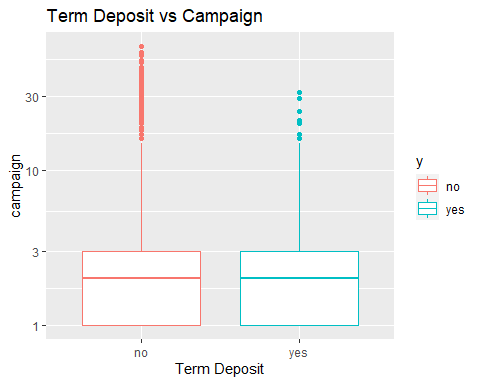
# maybe more likely on days 1, 10, 22, 30? Maybe something to do with paydays  
  
# Duration vs Response  
Bank\_Full %>%  
 ggplot(aes(x=y, y=duration, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Duration") +  
 xlab("Term Deposit")+scale\_y\_log10()

## Warning: Transformation introduced infinite values in continuous y-axis

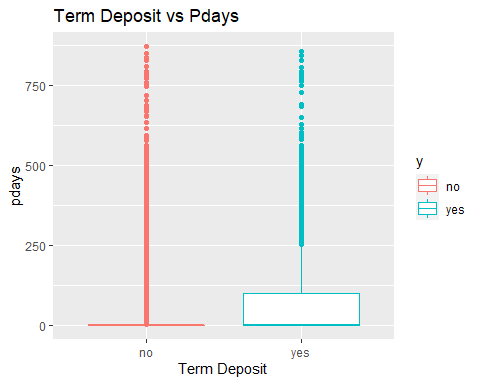
## Warning: Removed 3 rows containing non-finite values (stat\_boxplot).



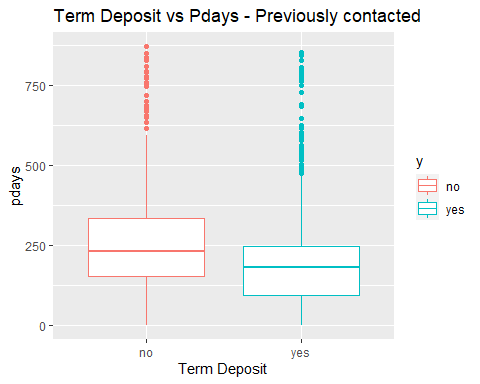
# Definitely a difference there - yes tends to have a longer duration  
# might want to look at that outlier in the high 4000s of duration  
  
# Campaign vs Response  
Bank\_Full %>%  
 ggplot(aes(x=y, y=campaign, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Campaign") +  
 xlab("Term Deposit")+scale\_y\_log10()



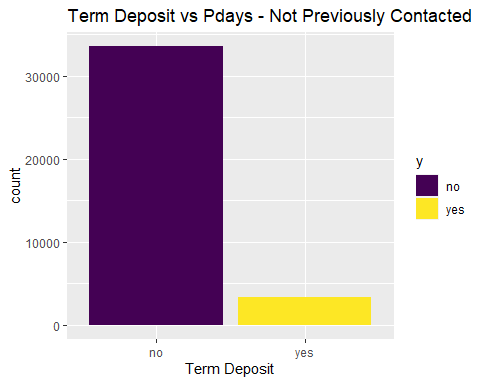
# The number of contacts performed during this campaign days is almost the same as for having term deposit and not having a term deposit   
# There might be a cut-off above which there are only No's (like >35 or 40)  
  
# Pdays vs Response (NOTE the meaning of -1)  
Bank\_Full %>%  
 ggplot(aes(x=y, y=pdays, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Pdays") +  
 xlab("Term Deposit")



# let's see what this looks like when we take out the -1:  
#The number of days that passed by after the client was last contacted from a previous campaign had more yes than nos.  
Bank\_Full %>%  
 filter(pdays != -1) %>%  
 ggplot(aes(x=y, y=pdays, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Pdays - Previously contacted") +  
 xlab("Term Deposit")



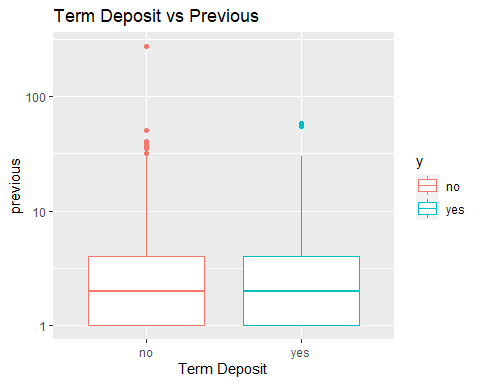
# So for those previously contacted: less days are associated with more yes in response  
# let's look at those not previously contacted:  
  
Bank\_Full %>%  
 filter(pdays == -1) %>%  
 ggplot(aes(x=y, fill=y)) +  
 geom\_bar() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Pdays - Not Previously Contacted") +  
 xlab("Term Deposit")



# look at count for not previously contacted. They are more than for no s than the yes s.  
  
#Bank\_Full %>% filter(pdays == -1) %>% count(y)  
# no:33570, yes:3384 (1/9 are yes)  
# look at count for previously contacted  
  
#Bank\_Full %>% filter(pdays != -1) %>% count(y)  
# No:6352, yes:1905 (1/3 are yes)  
# that's a big difference  
  
# Previous vs Response  
Bank\_Full %>%  
 ggplot(aes(x=y, y=previous, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Previous") +  
 xlab("Term Deposit")+scale\_y\_log10()

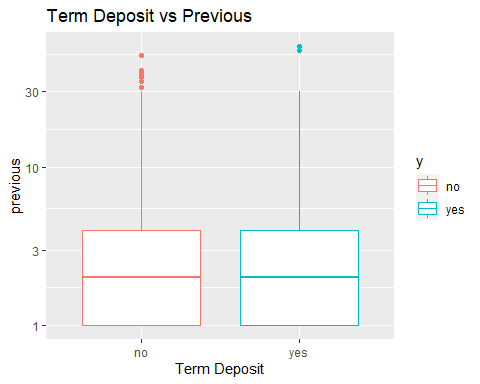
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 36954 rows containing non-finite values (stat\_boxplot).



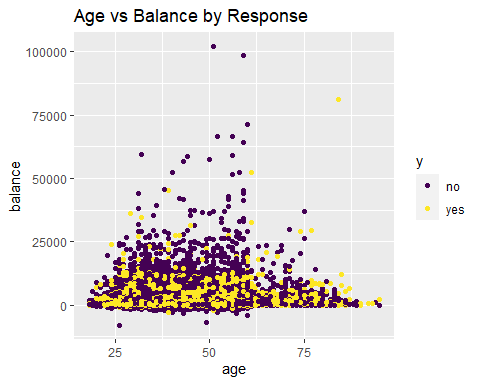
#The number of contacts performed before this campaign and for this client shows on an average the same for yes and no for having a term deposit.# outlier over 250  
# let's look at it w/o  
  
Bank\_Full %>%  
 filter(previous < 100) %>%  
 ggplot(aes(x=y, y=previous, color=y)) +  
 geom\_boxplot() +  
 scale\_fill\_viridis\_d() +  
 ggtitle("Term Deposit vs Previous") +  
 xlab("Term Deposit")+scale\_y\_log10()

## Warning: Transformation introduced infinite values in continuous y-axis  
  
## Warning: Removed 36954 rows containing non-finite values (stat\_boxplot).

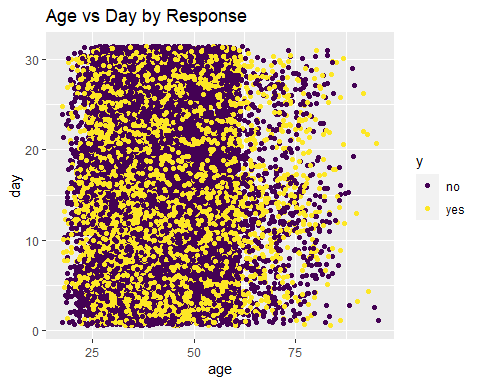


# hard to visualize. in the table, the mean for yes is higher though.

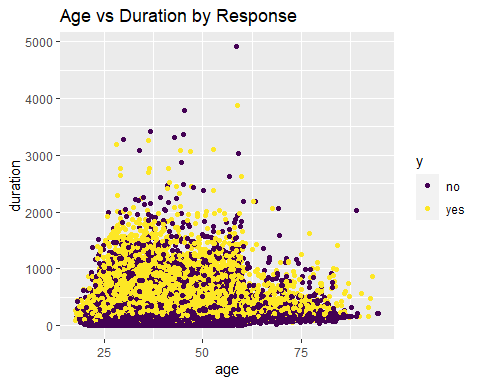
# paired scatterplots coded by response: AGE  
# (looking for possible interactions)  
# Age vs Balance  
Bank\_Full %>%  
 ggplot(aes(x=age, y=balance, color=y)) +  
 geom\_point()+  
#+scale\_y\_log10()+scale\_x\_log10()  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Balance by Response")



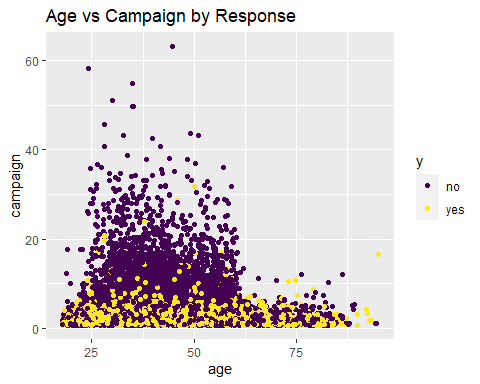
#Looks like they are linearly correlated to each other. We also see few outlier that are high leverage points for no (2 observations) and yes (1 observation).This could be a valid interaction as age and balance are linearly correlated.  
  
# Age vs Day  
Bank\_Full %>%  
 ggplot(aes(x=age, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Day by Response")



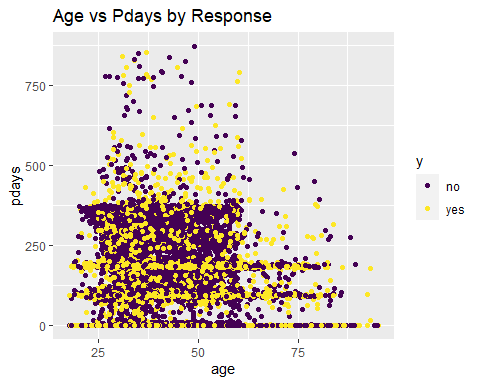
#Age and last contact day of the month dont seem to be valid interactions  
  
# Age vs Duration  
Bank\_Full %>%  
 ggplot(aes(x=age, y=duration, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Duration by Response")



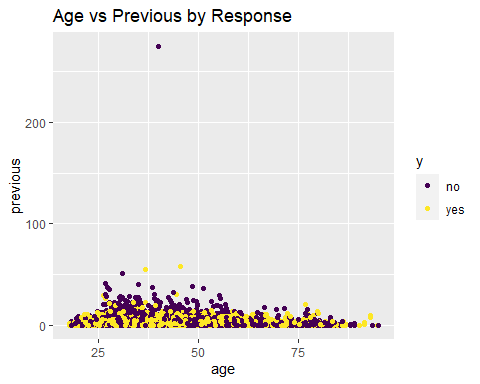
#Age and last contact duration dont seem to be valid interactions  
  
# Age vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=age, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Campaign by Response")



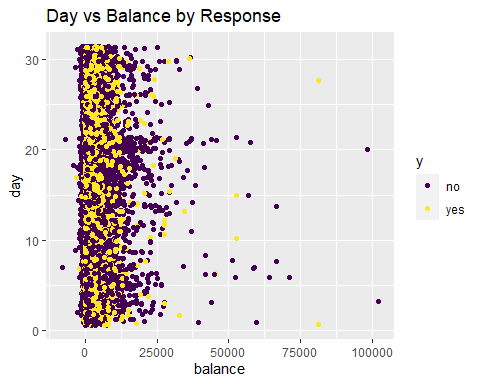
# There's a cut-off around 60 where people are no longer contacted by the campaign a ton of times.Age and campaign could play an import role in identifying the right candidate eligible for a term deposit.  
  
# Age vs Pdays  
Bank\_Full %>%  
 ggplot(aes(x=age, y=pdays, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Pdays by Response")



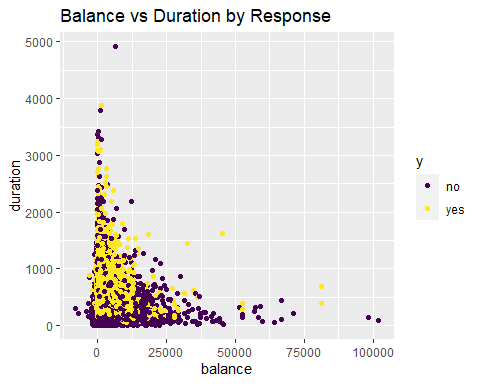
#Age and number of contacts performed during this campaign could be correlated and eligible for an interaction.  
  
# Age vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=age, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Previous by Response")



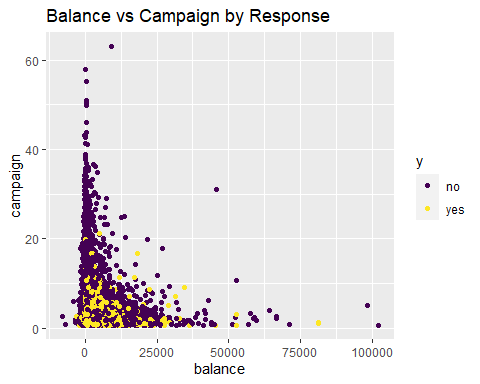
# note the outlier: previous = 275. Next closest is 58.  
#Age and number of contacts performed before this campaign could be potential for interactions.  
  
# paired scatterplots coded by response: BALANCE  
# (looking for possible interactions)  
# Balance vs Day  
Bank\_Full %>%  
 ggplot(aes(x=balance, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Day vs Balance by Response")



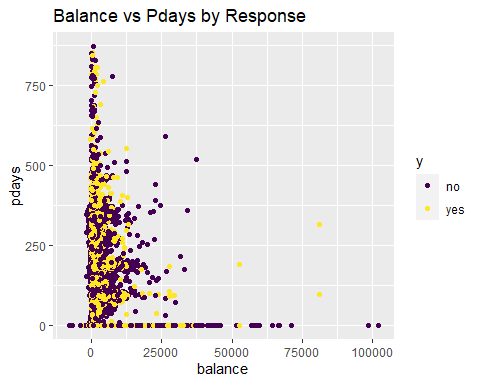
#Balance and last contact of the month could be potential for interaction.  
  
# Balance vs Duration  
Bank\_Full %>%  
 ggplot(aes(x=balance, y=duration, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Balance vs Duration by Response")



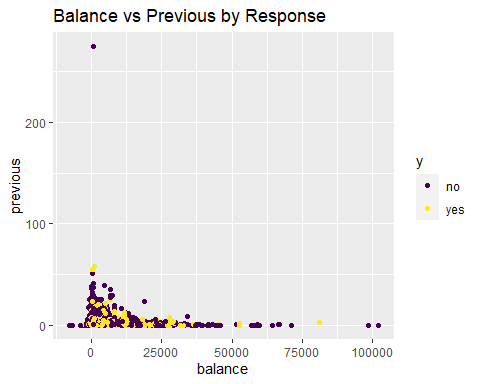
#Balance vs last contact duration dont se to be potential for interactionsS  
  
# Balance vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=balance, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Balance vs Campaign by Response")



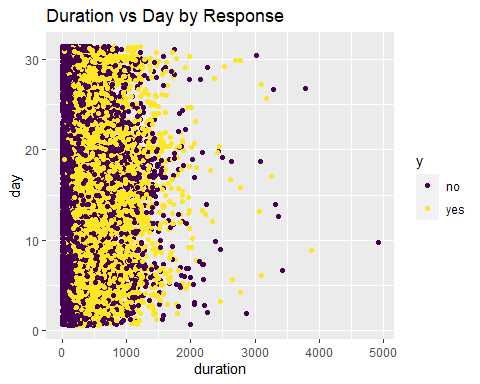
# As noted previously, there aren't any Yes's above a certain # in campaign  
  
# Balance vs Pdays  
Bank\_Full %>%  
 ggplot(aes(x=balance, y=pdays, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Balance vs Pdays by Response")



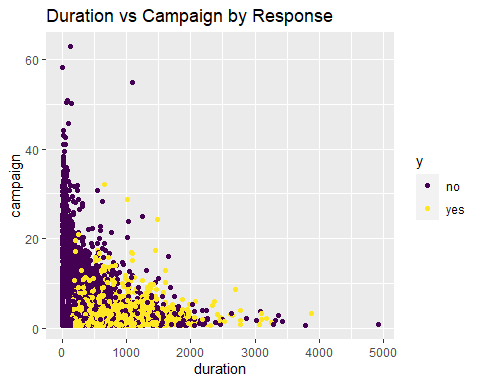
# note the outlier.Not potential for interactionsS  
  
# Balance vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=balance, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Balance vs Previous by Response")



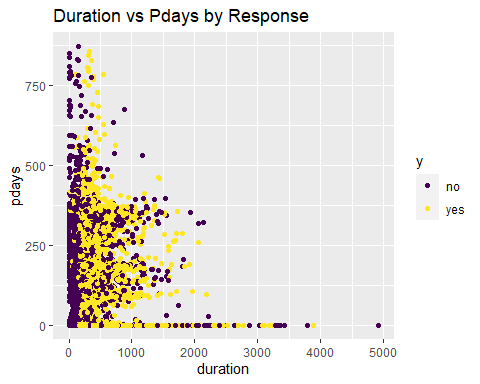
# note the outlier: previous = 275. Next closest is 58.  
  
# paired scatterplots coded by response: DURATION  
# (looking for possible interactions)  
# Duration vs Day  
Bank\_Full %>%  
 ggplot(aes(x=duration, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Duration vs Day by Response")



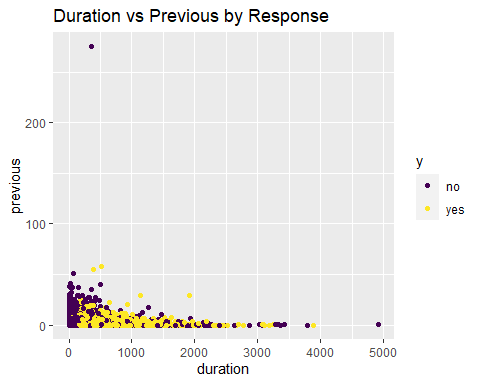
# note the outlier:  
  
# Duration vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=duration, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Duration vs Campaign by Response")



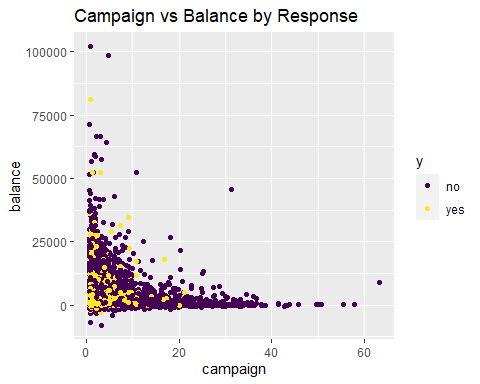
#Last contact duration and campain contacts dont seem to have an interaction.There are 1 outliers on Duration and 2 on campaign  
  
# Duration vs Pdays  
Bank\_Full %>%  
 ggplot(aes(x=duration, y=pdays, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Duration vs Pdays by Response")



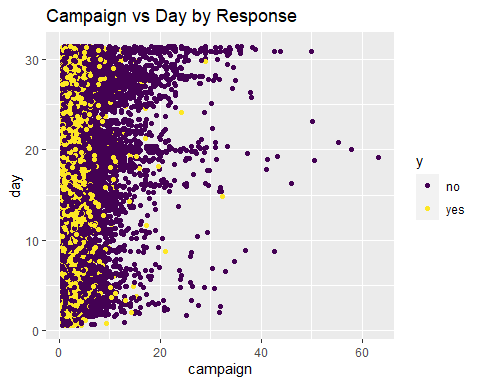
#Last contacted duration and pdays passed after client was last contacted could have an interaction.  
  
# Duration vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=duration, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Duration vs Previous by Response")



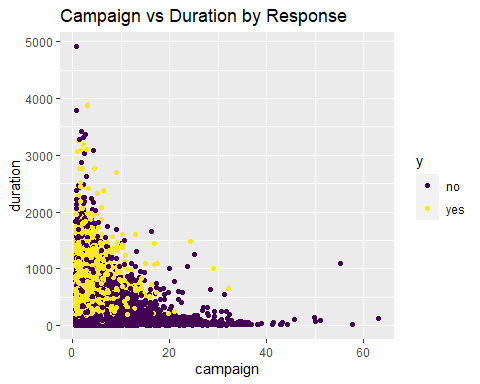
# note the outlier: previous = 275. Next closest is 58.  
  
# paired scatterplots coded by response: Campaign  
# (looking for possible interactions)  
# Campaign vs Balance  
Bank\_Full %>%  
 ggplot(aes(x=campaign, y=balance, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Campaign vs Balance by Response")



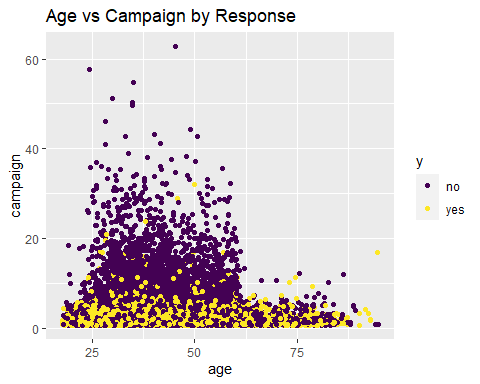
#Look at the outlier  
  
# Campaign vs Day  
Bank\_Full %>%  
 ggplot(aes(x=campaign, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Campaign vs Day by Response")



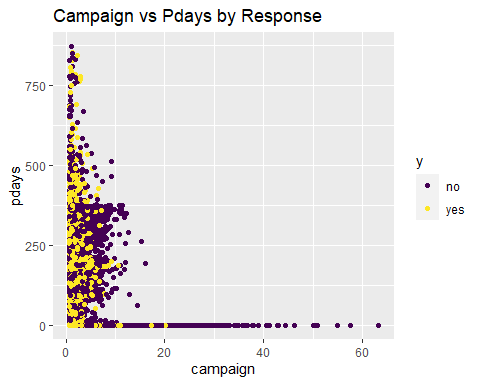
# Campaign vs Duration  
Bank\_Full %>%  
 ggplot(aes(x=campaign, y=duration, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Campaign vs Duration by Response")



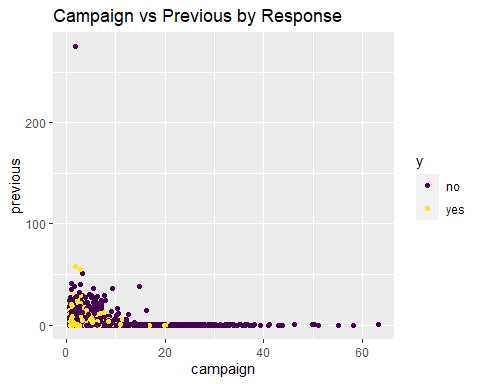
# Age vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=age, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Campaign by Response")



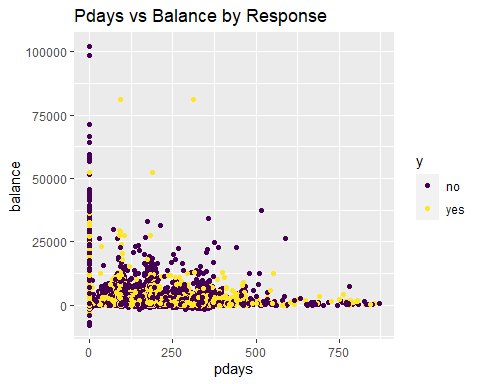
# There's a cut-off around 60 where people are no longer contacted by the campaign a ton of times.  
  
# Campaign vs Pdays  
Bank\_Full %>%  
 ggplot(aes(x=campaign, y=pdays, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Campaign vs Pdays by Response")



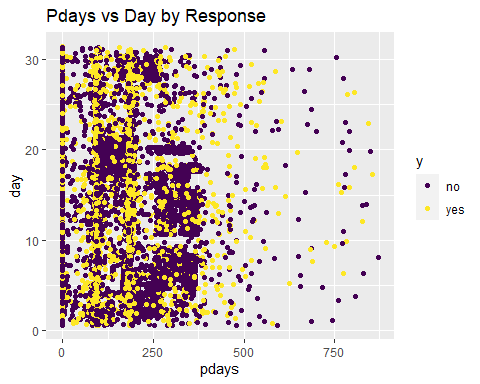
# Campaign vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=campaign, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Campaign vs Previous by Response")



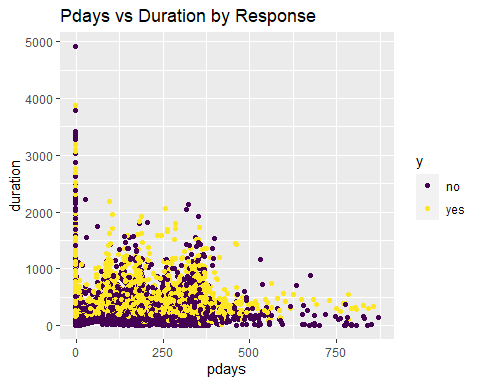
# So the people who were contacted a ton of times by this campaign were all people who had not been previously contacted before  
  
# paired scatterplots coded by response: PDAYS  
# (looking for possible interactions)  
# Pdays vs Balance  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=balance, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Balance by Response")



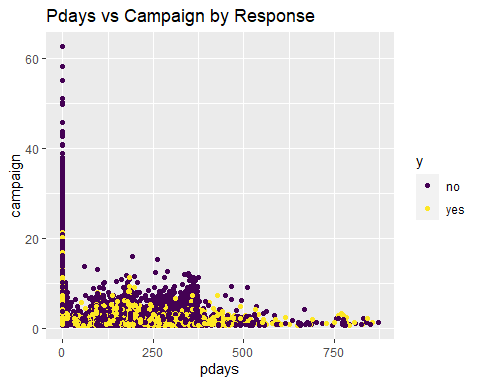
# Pdays vs Day  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Day by Response")



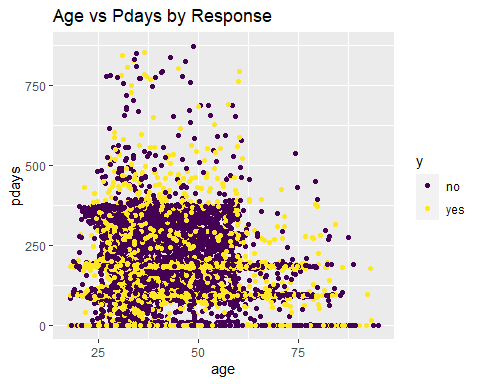
# Pdays vs Duration  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=duration, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Duration by Response")



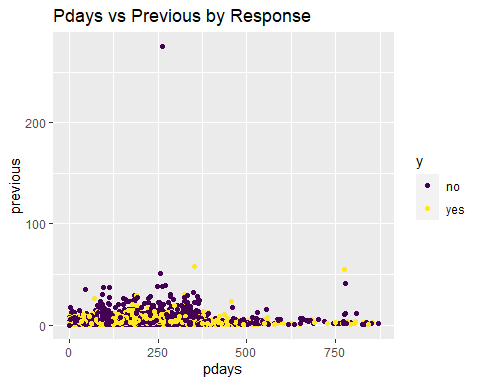
# Pdays vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Campaign by Response")



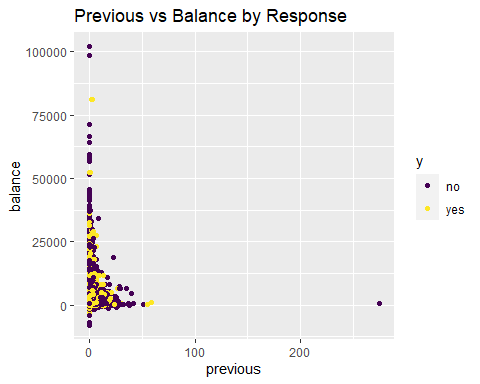
# Age vs Pdays  
Bank\_Full %>%  
 ggplot(aes(x=age, y=pdays, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Pdays by Response")



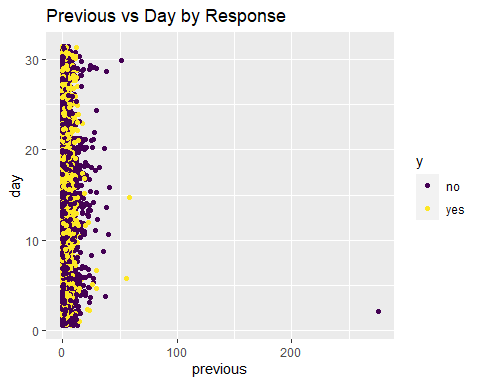
# Pdays vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Previous by Response")



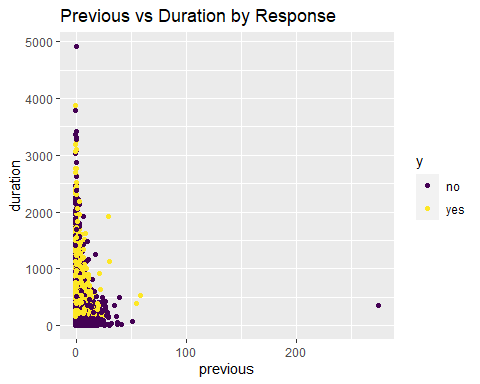
# note the outlier: previous = 275. Next closest is 58.  
  
# paired scatterplots coded by response: PREVIOUS  
# (looking for possible interactions)  
# Previous vs Balance  
Bank\_Full %>%  
 ggplot(aes(x=previous, y=balance, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Previous vs Balance by Response")



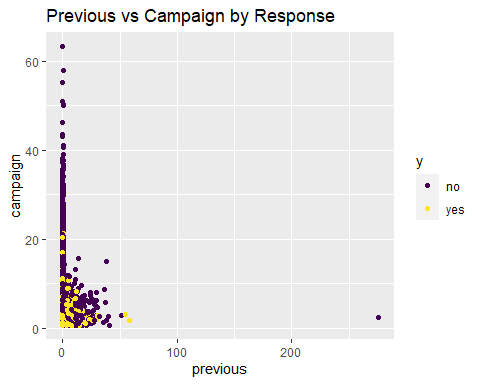
#There is the outlier.  
  
# Previous vs Day  
Bank\_Full %>%  
 ggplot(aes(x=previous, y=day, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Previous vs Day by Response")



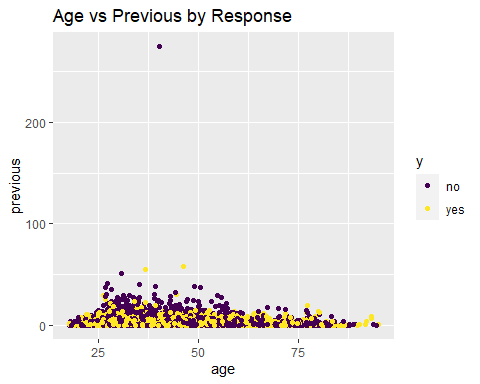
#The outlier is seen  
  
# Previous vs Duration  
Bank\_Full %>%  
 ggplot(aes(x=previous, y=duration, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Previous vs Duration by Response")



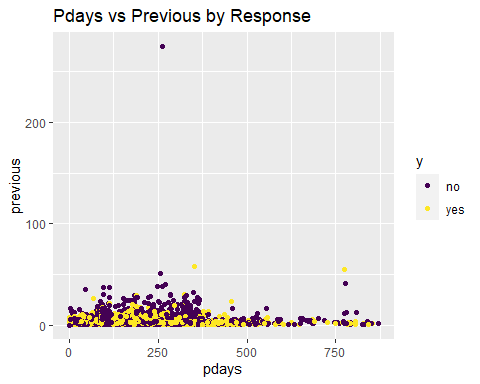
#The outlier is seen  
  
# Previous vs Campaign  
Bank\_Full %>%  
 ggplot(aes(x=previous, y=campaign, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Previous vs Campaign by Response")



#The outlier is seen  
  
# Age vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=age, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Age vs Previous by Response")

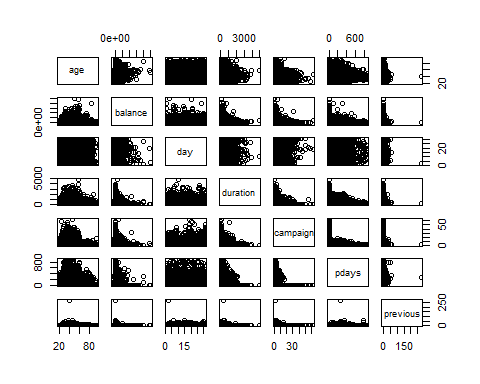


# note the outlier: previous = 275. Next closest is 58.  
  
# Pdays vs Previous  
Bank\_Full %>%  
 ggplot(aes(x=pdays, y=previous, color=y)) +  
 geom\_point(position="jitter") +  
 scale\_color\_viridis\_d() +  
 ggtitle("Pdays vs Previous by Response")



# note the outlier: previous = 275. Next closest is 58.

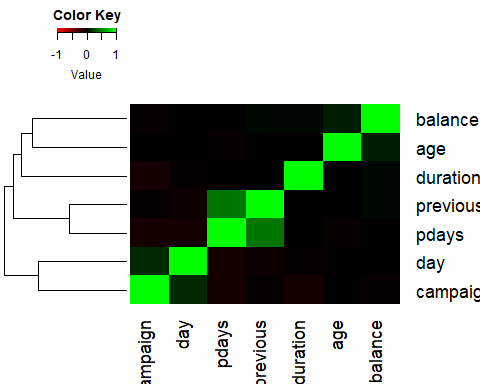
# Correlations between continuous variable  
# Exploring multicollinearity  
pairs(Bank\_Full[,c(1,6,10,12,13,14,15)])



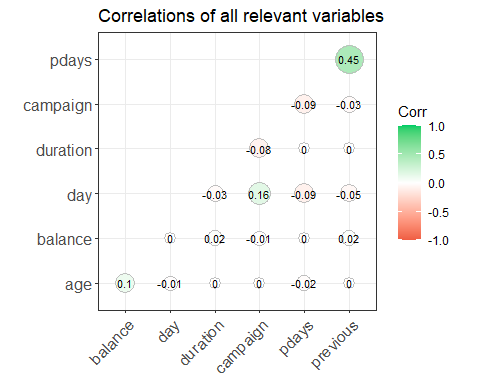
my.cor<-cor(Bank\_Full[,c(1,6,10,12,13,14,15)])  
my.cor

## age balance day duration campaign  
## age 1.000000000 0.097782739 -0.009120046 -0.004648428 0.004760312  
## balance 0.097782739 1.000000000 0.004502585 0.021560380 -0.014578279  
## day -0.009120046 0.004502585 1.000000000 -0.030206341 0.162490216  
## duration -0.004648428 0.021560380 -0.030206341 1.000000000 -0.084569503  
## campaign 0.004760312 -0.014578279 0.162490216 -0.084569503 1.000000000  
## pdays -0.023758014 0.003435322 -0.093044074 -0.001564770 -0.088627668  
## previous 0.001288319 0.016673637 -0.051710497 0.001203057 -0.032855290  
## pdays previous  
## age -0.023758014 0.001288319  
## balance 0.003435322 0.016673637  
## day -0.093044074 -0.051710497  
## duration -0.001564770 0.001203057  
## campaign -0.088627668 -0.032855290  
## pdays 1.000000000 0.454819635  
## previous 0.454819635 1.000000000

#pairs(Bank\_Full[,c(1,6,10,12,13,14,15)],col=Bank\_Full$y)  
# Heatmap  
heatmap.2(my.cor,col=redgreen(75),   
 density.info="none", trace="none", dendrogram=c("row"),   
 symm=F,symkey=T,symbreaks=T, scale="none")



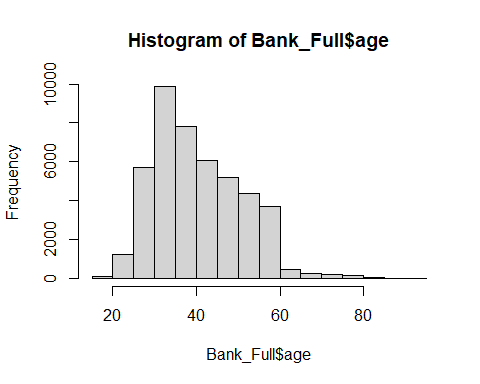
corr <- Bank\_Full %>% dplyr::select(age,balance,day,duration,campaign,pdays,previous)   
corr <- round(cor(corr), 2)  
ggcorrplot(corr, type = "lower",  
 lab = TRUE, lab\_size = 3, method = "circle",  
 colors = c("tomato2", "white", "springgreen3"),  
 title = "Correlations of all relevant variables",  
 ggtheme = theme\_bw())



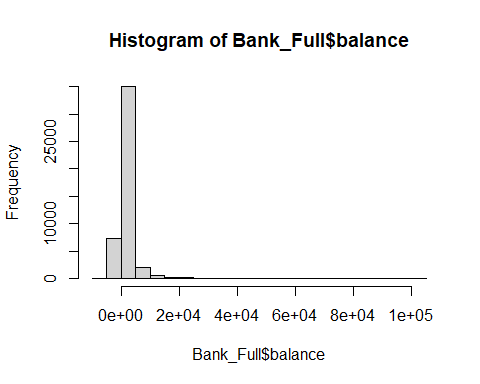
# previous and pdays correlation = 0.45. Will also be correlated with the dummy variable we build  
  
# multicollinearity w/ categorical speculation  
# Hyp: older people would be more likely to have landline  
#plot(Bank\_Full$age~Bank\_Full$contact,col=c("red","blue"))  
# Yes, but there's overlap  
# Hyp: balance and education   
#plot(Bank\_Full$balance~Bank\_Full$education,col=c("red","blue"))  
#plot(log(Bank\_Full$balance)~Bank\_Full$education,col=c("red","blue"))  
# note I know there's zeros, just looking   
# slight but not dramatic  
# age and marital  
#plot(Bank\_Full$age~Bank\_Full$marital,col=c("red","blue"))  
# single people tend to be younger, as you'd expect  
# housing and loan  
#plot(Bank\_Full$housing~Bank\_Full$loan,col=c("red","blue")) # similar ratios

#model.main<-glm(age~., data=Bank\_Full)  
#vif(model.main)  
#Using this tool, GVIF is the same as VIF for continuous predictors only  
#For categorical predictors, the value GVIG^(1/(2\*df)) should be squared and interpreted  
#as a usual vif type metric.The following code can be used to interpret VIFs like we   
#discussed in class.  
#(vif(model.main)[,3])^2  
#VIF look good.Nothing seems to be greater than 10

# Normality will be a concern for LDA/QDA:  
hist(Bank\_Full$age) #fine



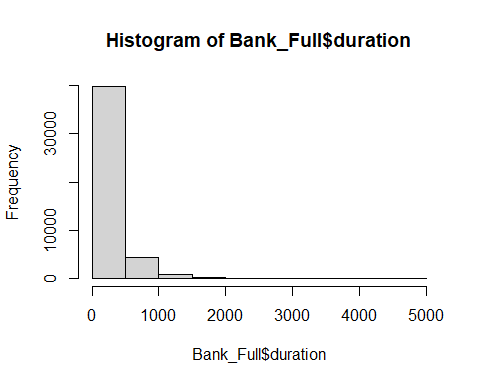
hist(Bank\_Full$balance) #skew



range(Bank\_Full$balance) # This is skewed but has negative numbers

## [1] -8019 102127

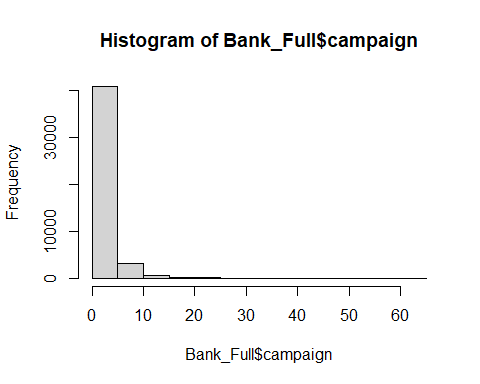
hist(Bank\_Full$duration) #skew



range(Bank\_Full$duration) # There are 3 zeros...

## [1] 0 4918

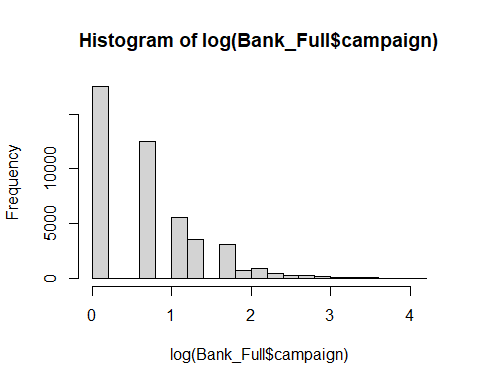
hist(Bank\_Full$campaign)#skew



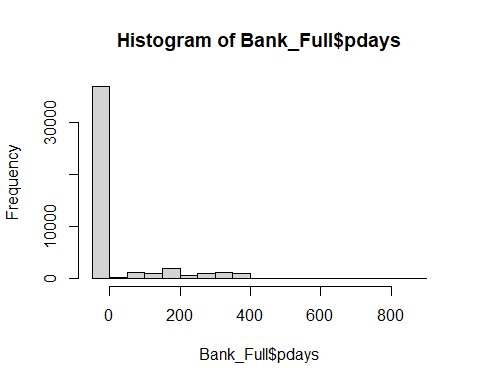
range(Bank\_Full$campaign) # 1-3

## [1] 1 63

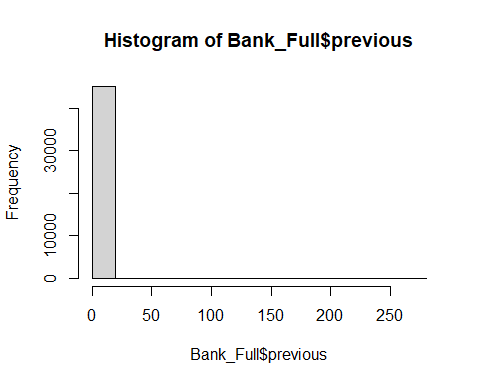
hist(log(Bank\_Full$campaign)) # still doesn't look great



hist(Bank\_Full$pdays) #skewed, but there's negative -1's that we will likely change to 0



hist(Bank\_Full$previous)

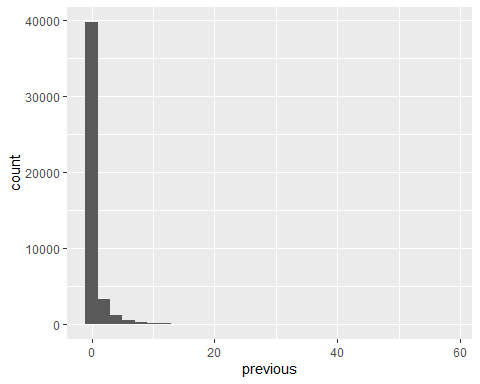


range(Bank\_Full$previous) #0-275 (but next lowest is 58)

## [1] 0 275

Bank\_Full %>%  
 filter(previous <60) %>%  
 ggplot(aes(x=previous)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# look at the identified possible outliers  
# Duration  
summary(Bank\_Full$duration)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0 103.0 180.0 258.2 319.0 4918.0

Bank\_Full %>%   
 filter(Bank\_Full$duration == 4918)

## age job marital education default balance housing loan contact day  
## 1 59 technician married tertiary no 6573 yes no telephone 10  
## month duration campaign pdays previous poutcome y id year  
## 1 nov 4918 1 -1 0 unknown no 24149 2008

# 4918 is max (response is no), next value following it is 3881  
# 4918/60 = 81.9 minutes. 3881/60 = 64.68 minutes. Median = 3 min, Mean = 4.3 min, 3rd Q = 5.31 min  
# Can we justify throwing this out by saying 81.9 minutes is not a good use of time? That's 19 average phone calls...  
# Also this is Nov 2008. Depending on what we do, it may be a non-issue  
# Previous  
summary(Bank\_Full$previous)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 0.0000 0.0000 0.5803 0.0000 275.0000

# Might make more sense to look at it w/o the 0s:  
Bank\_Full %>%  
 filter(previous >0) %>%  
 summarise(mean = mean(previous),   
 median = median(previous),   
 min = min(previous),  
 max = max(previous),  
 quantile = quantile(previous))

## mean median min max quantile  
## 1 3.177546 2 1 275 1  
## 2 3.177546 2 1 275 1  
## 3 3.177546 2 1 275 2  
## 4 3.177546 2 1 275 4  
## 5 3.177546 2 1 275 275

# 275 (response is no), next value is 58   
Bank\_Full %>%  
 filter(Bank\_Full$previous == 275)

## age job marital education default balance housing loan contact day  
## 1 40 management married tertiary no 543 yes no cellular 2  
## month duration campaign pdays previous poutcome y id year  
## 1 feb 349 2 262 275 other no 29183 2009

# I suppose we could say that we don't have enough data in this date range to be useful  
# So we can restrict the range down to <100 or something.  
# Removing the 2 outliers:  
# double checking my work  
nrow(Bank\_Full) # 45211

## [1] 45211

Bank\_Full = subset(Bank\_Full, duration != 4918)  
nrow(Bank\_Full) #45210

## [1] 45210

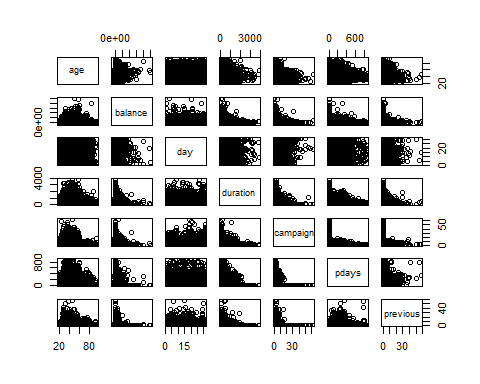
Bank\_Full = subset(Bank\_Full, previous != 275)  
nrow(Bank\_Full) #45209

## [1] 45209

#PCA as part of the EDA  
# Continuous predictors  
#For the 3 Years (2008,2009,2010)  
# This includes: age, balance, day, duration, campaign, pdays, previous  
str(Bank\_Full)

## 'data.frame': 45209 obs. of 19 variables:  
## $ age : int 58 44 33 47 33 35 28 42 58 43 ...  
## $ job : chr "management" "technician" "entrepreneur" "blue-collar" ...  
## $ marital : chr "married" "single" "married" "married" ...  
## $ education: chr "tertiary" "secondary" "secondary" "unknown" ...  
## $ default : chr "no" "no" "no" "no" ...  
## $ balance : int 2143 29 2 1506 1 231 447 2 121 593 ...  
## $ housing : chr "yes" "yes" "yes" "yes" ...  
## $ loan : chr "no" "no" "yes" "no" ...  
## $ contact : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ day : int 5 5 5 5 5 5 5 5 5 5 ...  
## $ month : Factor w/ 12 levels "jan","feb","mar",..: 5 5 5 5 5 5 5 5 5 5 ...  
## $ duration : int 261 151 76 92 198 139 217 380 50 55 ...  
## $ campaign : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ pdays : int -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...  
## $ previous : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ poutcome : chr "unknown" "unknown" "unknown" "unknown" ...  
## $ y : chr "no" "no" "no" "no" ...  
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ year : Factor w/ 3 levels "2008","2009",..: 1 1 1 1 1 1 1 1 1 1 ...

reduced<-Bank\_Full[,c(1,6,10,12,13,14,15)]  
pairs(reduced)



#Let's take a quick look at the summary statistics and in particular lets calculate the variance of each variable and add them up to obtain the total variance.  
apply(reduced,2,summary)

## age balance day duration campaign pdays previous  
## Min. 18.00000 -8019.000 1.00000 0.000 1.000000 -1.00000 0.0000000  
## 1st Qu. 33.00000 72.000 8.00000 103.000 1.000000 -1.00000 0.0000000  
## Median 39.00000 448.000 16.00000 180.000 2.000000 -1.00000 0.0000000  
## Mean 40.93583 1362.175 15.80685 258.058 2.763897 40.19383 0.5742662  
## 3rd Qu. 48.00000 1428.000 21.00000 319.000 3.000000 -1.00000 0.0000000  
## Max. 95.00000 102127.000 31.00000 3881.000 63.000000 871.00000 58.0000000

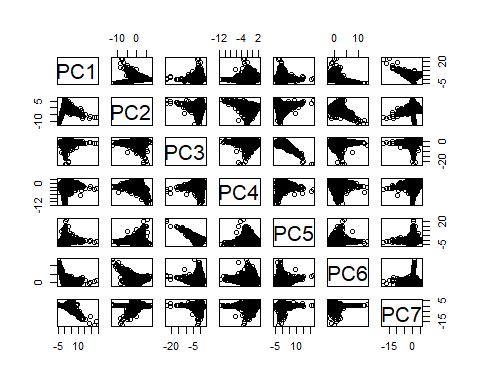
var.raw<-apply(reduced,2,var)  
var.raw

## age balance day duration campaign pdays   
## 1.127559e+02 9.270394e+06 6.926171e+01 6.584300e+04 9.598076e+00 1.002508e+04   
## previous   
## 3.640261e+00

#Total variance  
sum(var.raw)

## [1] 9346457

#Running PCA is relatively straight forward. The following script conducts a PCA using the covariance matrix (nonstandardarized #variables) and stores the results in an object. This object contains the eigenvectors, eigenvalue, and the new principle #component vectors. Lets start by producing a correlation matrix to verify that new principle component variables are #uncorrelated.  
pc.result<-prcomp(Bank\_Full[,c(1,6,10,12,13,14,15)],scale.=TRUE)  
pc.scores<-pc.result$x  
pairs(pc.scores)



cor(pc.scores)

## PC1 PC2 PC3 PC4 PC5  
## PC1 1.000000e+00 1.646821e-15 7.240806e-16 7.513190e-16 1.061386e-15  
## PC2 1.646821e-15 1.000000e+00 -4.098392e-15 5.588093e-15 -2.452901e-15  
## PC3 7.240806e-16 -4.098392e-15 1.000000e+00 -1.011659e-14 3.121267e-14  
## PC4 7.513190e-16 5.588093e-15 -1.011659e-14 1.000000e+00 -1.358345e-14  
## PC5 1.061386e-15 -2.452901e-15 3.121267e-14 -1.358345e-14 1.000000e+00  
## PC6 4.651381e-16 -1.817970e-15 5.957578e-15 1.881012e-15 6.553160e-15  
## PC7 -1.156057e-15 -9.565370e-17 -1.828315e-15 1.509032e-15 -2.007877e-15  
## PC6 PC7  
## PC1 4.651381e-16 -1.156057e-15  
## PC2 -1.817970e-15 -9.565370e-17  
## PC3 5.957578e-15 -1.828315e-15  
## PC4 1.881012e-15 1.509032e-15  
## PC5 6.553160e-15 -2.007877e-15  
## PC6 1.000000e+00 6.575485e-16  
## PC7 6.575485e-16 1.000000e+00

#We can again verify that the total variance in the new PC variables is exactly the same as the original data. The eigenvectors are stored inside of "pc.result" as well in the "rotation" object.  
var.pca<-apply(pc.scores,2,var)  
var.pca

## PC1 PC2 PC3 PC4 PC5 PC6 PC7   
## 1.5895869 1.1600390 1.0974117 0.9754226 0.8972295 0.8260487 0.4542615

#Total Variance of PC's  
sum(var.pca)

## [1] 7

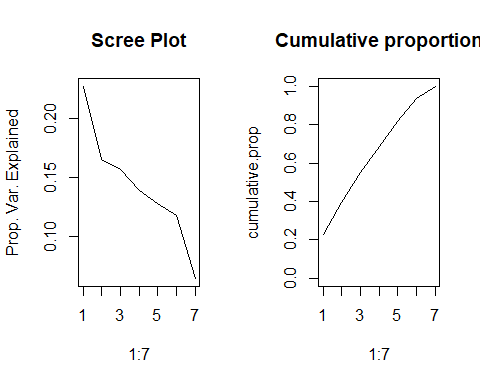
#Total Variance of Original Variables.  
sum(var.raw)

## [1] 9346457

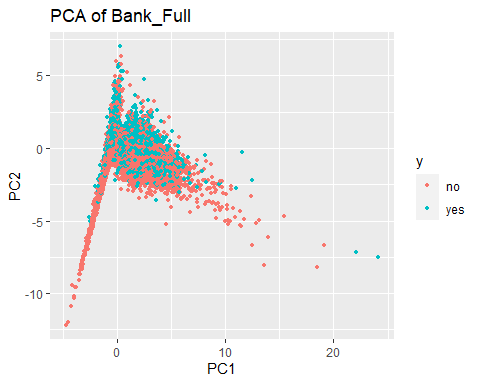
#List of eigenvectors  
pc.result$rotation

## PC1 PC2 PC3 PC4 PC5  
## age -0.01889173 0.09584663 -0.696026622 0.23882876 -0.659588776  
## balance 0.02948460 0.12361649 -0.697442213 -0.15340804 0.678698184  
## day -0.23229130 -0.53523432 -0.114209259 -0.47634785 -0.009841468  
## duration 0.04300790 0.44775037 0.017235867 -0.81694076 -0.287173990  
## campaign -0.21653157 -0.63044363 -0.108307865 -0.12096282 -0.135356238  
## pdays 0.67768911 -0.17892257 -0.001868909 -0.04901571 -0.024472952  
## previous 0.66092086 -0.24311265 -0.063611200 -0.08995985 -0.053154775  
## PC6 PC7  
## age -0.113984599 0.03023984  
## balance 0.113764723 0.01778535  
## day -0.646934159 0.03151640  
## duration 0.217593428 0.01283948  
## campaign 0.712431968 0.05840970  
## pdays -0.023886592 0.71073810  
## previous 0.008032136 -0.69932164

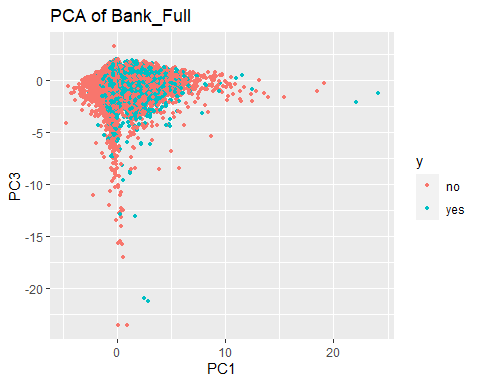
#A scree plot of the eigenvalues used to determine how many pc's to keep can be plotted in the following way:  
par(mfrow=c(1,2))  
eigenvals<-(pc.result$sdev)^2  
plot(1:7,eigenvals/sum(eigenvals),type="l",main="Scree Plot",ylab="Prop. Var. Explained")  
cumulative.prop<-cumsum(eigenvals/sum(eigenvals))  
plot(1:7,cumulative.prop,type="l",main="Cumulative proportion",ylim=c(0,1))



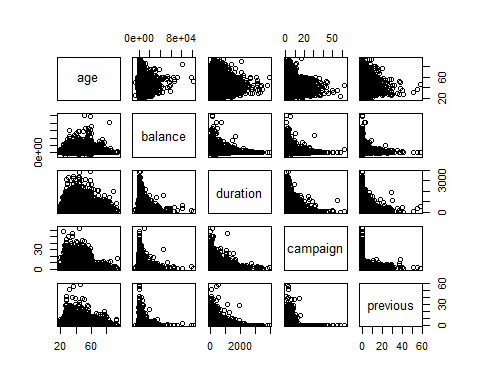
par(mfrow=c(1,1))  
#The scree plots show the elbow around 2.The variance explained for .9 is 6.   
#Use ggplot2 to plot the first few pc's  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC2)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



#Looking at PC1 and PC3  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC3)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



# Not great. Some separation. Still debating using day in there.  
# PCA w/o day:  
# This includes: age, balance, duration, campaign, pdays, previous  
# We might want to do this again when we address normality for LDA/QDA.  
# Not necessary for PCA but slides indicate it may improve the results.  
# PCA without day and pdays  
reduced<-Bank\_Full[,c(1,6,12,13,15)]  
pairs(reduced)



apply(reduced,2,summary)

## age balance duration campaign previous  
## Min. 18.00000 -8019.000 0.000 1.000000 0.0000000  
## 1st Qu. 33.00000 72.000 103.000 1.000000 0.0000000  
## Median 39.00000 448.000 180.000 2.000000 0.0000000  
## Mean 40.93583 1362.175 258.058 2.763897 0.5742662  
## 3rd Qu. 48.00000 1428.000 319.000 3.000000 0.0000000  
## Max. 95.00000 102127.000 3881.000 63.000000 58.0000000

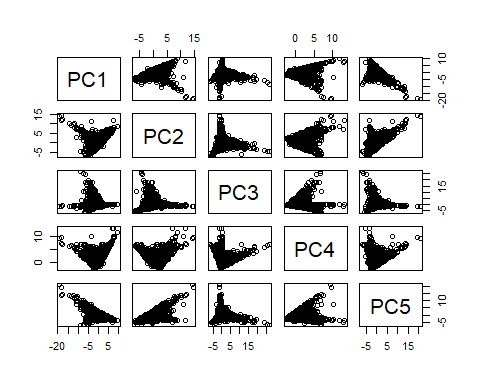
var.raw<-apply(reduced,2,var)  
var.raw

## age balance duration campaign previous   
## 1.127559e+02 9.270394e+06 6.584300e+04 9.598076e+00 3.640261e+00

#Total variance  
sum(var.raw)

## [1] 9336363

pc.result<-prcomp(Bank\_Full[,c(1,6,12,13,15)],scale.=TRUE)  
pc.scores<-pc.result$x  
pairs(pc.scores)



cor(pc.scores)

## PC1 PC2 PC3 PC4 PC5  
## PC1 1.000000e+00 -1.892153e-14 3.215478e-15 1.076578e-14 1.790066e-14  
## PC2 -1.892153e-14 1.000000e+00 -4.064115e-15 -1.361522e-14 -2.300901e-14  
## PC3 3.215478e-15 -4.064115e-15 1.000000e+00 2.339893e-15 4.466491e-15  
## PC4 1.076578e-14 -1.361522e-14 2.339893e-15 1.000000e+00 1.848995e-14  
## PC5 1.790066e-14 -2.300901e-14 4.466491e-15 1.848995e-14 1.000000e+00

var.pca<-apply(pc.scores,2,var)  
var.pca

## PC1 PC2 PC3 PC4 PC5   
## 1.1145237 1.0811957 0.9988439 0.9101864 0.8952504

#Total Variance of PC's  
sum(var.pca)

## [1] 5

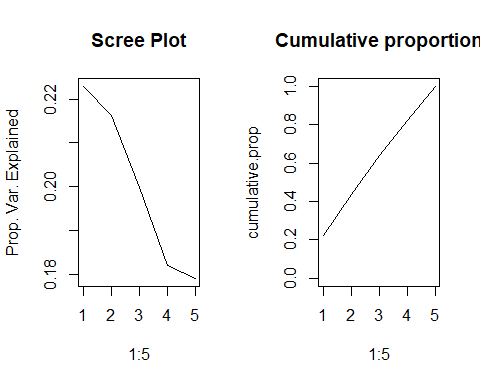
#Total Variance of Original Variables.  
sum(var.raw)

## [1] 9336363

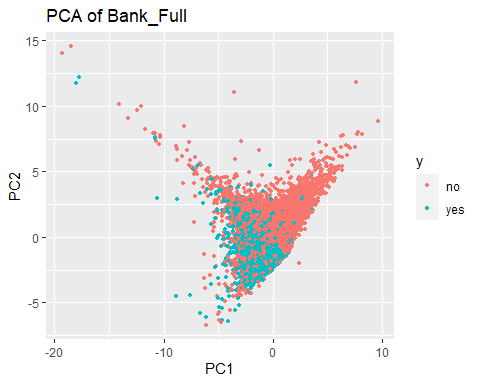
#List of eigenvectors  
pc.result$rotation

## PC1 PC2 PC3 PC4 PC5  
## age -0.4549616 0.5505030 -0.105325051 -0.3530449 -0.5951658  
## balance -0.5737513 0.4077508 -0.038322792 0.3168371 0.6345820  
## duration -0.4242188 -0.4824403 -0.414327159 0.5319689 -0.3641867  
## campaign 0.4592187 0.5281887 -0.003664299 0.6663462 -0.2571074  
## previous -0.2701408 -0.1376724 0.903192989 0.2190102 -0.2105878

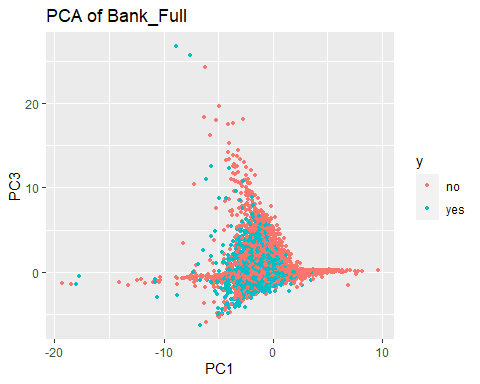
#A scree plot of the eigenvalues used to determine how many pc's to keep can be plotted in the following way:  
par(mfrow=c(1,2))  
eigenvals<-(pc.result$sdev)^2  
plot(1:5,eigenvals/sum(eigenvals),type="l",main="Scree Plot",ylab="Prop. Var. Explained")  
cumulative.prop<-cumsum(eigenvals/sum(eigenvals))  
plot(1:5,cumulative.prop,type="l",main="Cumulative proportion",ylim=c(0,1))



par(mfrow=c(1,1))  
#Removing day and pdays The scree plots show the elbow around 4 predictors.The variance explained for 80 of the variance is 4.   
#Use ggplot2 to plot the first few pc's  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC2)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



#Looking at PC1 and PC3  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC3)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



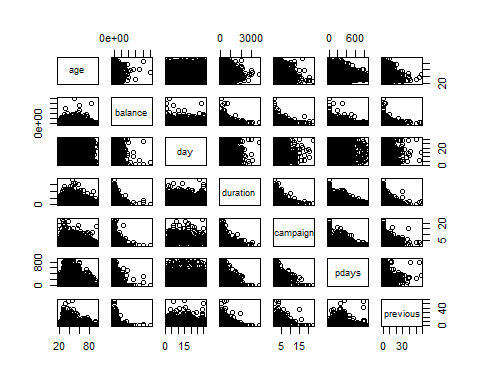
#Without day and pday we find that the model requires first 4 principle components or combinations of these to explain about 80% of the variability.  
  
#Removing year 2008  
# Explore what the data looks like if we remove 2008  
prop.table(table(Bank\_Full$y,Bank\_Full$year),2)

##   
## 2008 2009 2010  
## no 0.94947346 0.82935200 0.48396947  
## yes 0.05052654 0.17064800 0.51603053

#plot(Bank\_Full$y~Bank\_Full$year,col=c("red","blue"))  
Bank\_Full.2008 = Bank\_Full %>%  
 filter(year != 2008)  
summary(Bank\_Full.2008)

## age job marital education   
## Min. :18.00 Length:17481 Length:17481 Length:17481   
## 1st Qu.:31.00 Class :character Class :character Class :character   
## Median :37.00 Mode :character Mode :character Mode :character   
## Mean :40.44   
## 3rd Qu.:48.00   
## Max. :95.00   
##   
## default balance housing loan   
## Length:17481 Min. : -4057 Length:17481 Length:17481   
## Class :character 1st Qu.: 144 Class :character Class :character   
## Mode :character Median : 539 Mode :character Mode :character   
## Mean : 1437   
## 3rd Qu.: 1536   
## Max. :102127   
##   
## contact day month duration   
## Length:17481 Min. : 1.00 may :5809 Min. : 0.0   
## Class :character 1st Qu.: 6.00 apr :2932 1st Qu.: 112.0   
## Mode :character Median :13.00 feb :2648 Median : 197.0   
## Mean :13.73 jan :1403 Mean : 267.6   
## 3rd Qu.:19.00 aug :1032 3rd Qu.: 336.0   
## Max. :31.00 jun : 855 Max. :3785.0   
## (Other):2802   
## campaign pdays previous poutcome   
## Min. : 1.000 Min. : -1.00 Min. : 0.000 Length:17481   
## 1st Qu.: 1.000 1st Qu.: -1.00 1st Qu.: 0.000 Class :character   
## Median : 2.000 Median : -1.00 Median : 0.000 Mode :character   
## Mean : 2.094 Mean : 97.58 Mean : 1.333   
## 3rd Qu.: 2.000 3rd Qu.:185.00 3rd Qu.: 2.000   
## Max. :23.000 Max. :871.00 Max. :58.000   
##   
## y id year   
## Length:17481 Min. :27730 2008: 0   
## Class :character 1st Qu.:32101 2009:14861   
## Mode :character Median :36471 2010: 2620   
## Mean :36471   
## 3rd Qu.:40841   
## Max. :45211   
##

# 17481 rows. 13593 no to 3888 yes. (3.5:1) Much less unbalanced.  
#For the 3 Years  
# This includes: age, balance, day, duration, campaign, pdays, previous  
reduced<-Bank\_Full.2008[,c(1,6,10,12,13,14,15)]  
pairs(reduced)



#Let's take a quick look at the summary statistics and in particular lets calculate the variance of each variable and add them up #to obtain the total variance.  
apply(reduced,2,summary)

## age balance day duration campaign pdays previous  
## Min. 18.00000 -4057.000 1.00000 0.000 1.00000 -1.00000 0.000000  
## 1st Qu. 31.00000 144.000 6.00000 112.000 1.00000 -1.00000 0.000000  
## Median 37.00000 539.000 13.00000 197.000 2.00000 -1.00000 0.000000  
## Mean 40.43602 1436.968 13.73222 267.627 2.09353 97.58469 1.333505  
## 3rd Qu. 48.00000 1536.000 19.00000 336.000 2.00000 185.00000 2.000000  
## Max. 95.00000 102127.000 31.00000 3785.000 23.00000 871.00000 58.000000

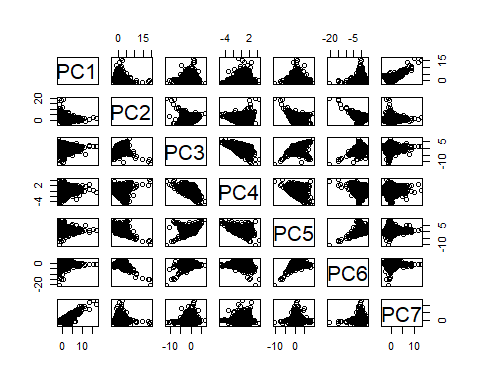
var.raw<-apply(reduced,2,var)  
var.raw

## age balance day duration campaign pdays   
## 1.512222e+02 9.346700e+06 7.194780e+01 6.357241e+04 2.762934e+00 1.932927e+04   
## previous   
## 7.568860e+00

#Total variance  
sum(var.raw)

## [1] 9429835

#Running PCA is relatively straight forward. The following script conducts a PCA using the covariance matrix (nonstandardarized #variables) and stores the results in an object. This object contains the eigenvectors, eigenvalue, and the new principle #component vectors. Lets start by producing a correlation matrix to verify that new principle component variables are #uncorrelated.  
pc.result<-prcomp(Bank\_Full.2008[,c(1,6,10,12,13,14,15)],scale.=TRUE)  
pc.scores<-pc.result$x  
pairs(pc.scores)



cor(pc.scores)

## PC1 PC2 PC3 PC4 PC5  
## PC1 1.000000e+00 -1.427296e-16 -9.148076e-16 -1.202620e-16 9.366677e-16  
## PC2 -1.427296e-16 1.000000e+00 -5.497426e-15 -2.570594e-16 7.589130e-16  
## PC3 -9.148076e-16 -5.497426e-15 1.000000e+00 -1.525847e-15 -6.003010e-16  
## PC4 -1.202620e-16 -2.570594e-16 -1.525847e-15 1.000000e+00 -7.099381e-16  
## PC5 9.366677e-16 7.589130e-16 -6.003010e-16 -7.099381e-16 1.000000e+00  
## PC6 -1.032868e-16 1.785161e-14 -9.561581e-15 1.287745e-15 -1.103208e-15  
## PC7 3.088198e-15 -3.035899e-15 1.328950e-15 3.270716e-16 -6.093781e-16  
## PC6 PC7  
## PC1 -1.032868e-16 3.088198e-15  
## PC2 1.785161e-14 -3.035899e-15  
## PC3 -9.561581e-15 1.328950e-15  
## PC4 1.287745e-15 3.270716e-16  
## PC5 -1.103208e-15 -6.093781e-16  
## PC6 1.000000e+00 -1.512648e-15  
## PC7 -1.512648e-15 1.000000e+00

#We can again verify that the total variance in the new PC variables is exactly the same as the original data. The eigenvectors are stored inside of "pc.result" as well in the "rotation" object.  
var.pca<-apply(pc.scores,2,var)  
var.pca

## PC1 PC2 PC3 PC4 PC5 PC6 PC7   
## 1.4707498 1.1539191 1.0635289 0.9943371 0.9088628 0.8764210 0.5321814

#Total Variance of PC's  
sum(var.pca)

## [1] 7

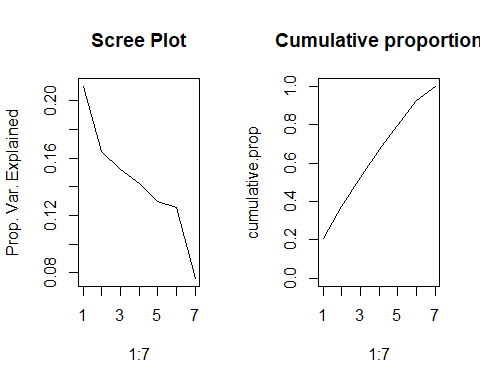
#Total Variance of Original Variables.  
sum(var.raw)

## [1] 9429835

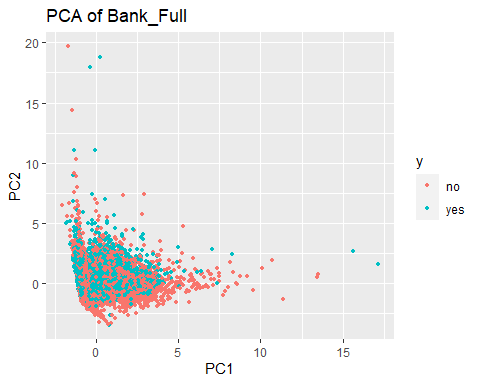
#List of eigenvectors  
pc.result$rotation

## PC1 PC2 PC3 PC4 PC5  
## age -0.007038238 0.61876837 -0.30570311 0.02864049 0.002653833  
## balance -0.026411505 0.59292986 -0.34132501 0.13541651 -0.276511458  
## day -0.045536753 0.10334149 0.33534634 0.90698451 0.225902049  
## duration -0.077477477 0.43916842 0.43889255 -0.37809106 0.657217710  
## campaign 0.147971974 -0.22767737 -0.67677310 0.11806214 0.655109872  
## pdays 0.691815968 0.03974483 0.15912307 -0.02711785 -0.100011876  
## previous 0.700479480 0.09270859 0.04021234 0.02437815 0.037365723  
## PC6 PC7  
## age 0.72248193 -0.02853618  
## balance -0.65879419 -0.04927316  
## day 0.01496671 -0.03021812  
## duration -0.17997798 -0.03495188  
## campaign -0.10181403 -0.12153918  
## pdays 0.01400625 -0.69538258  
## previous -0.02883964 0.70448163

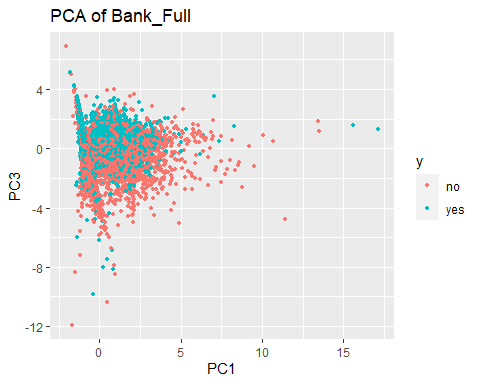
#A scree plot of the eigenvalues used to determine how many pc's to keep can be plotted in the following way:  
par(mfrow=c(1,2))  
eigenvals<-(pc.result$sdev)^2  
plot(1:7,eigenvals/sum(eigenvals),type="l",main="Scree Plot",ylab="Prop. Var. Explained")  
cumulative.prop<-cumsum(eigenvals/sum(eigenvals))  
plot(1:7,cumulative.prop,type="l",main="Cumulative proportion",ylim=c(0,1))



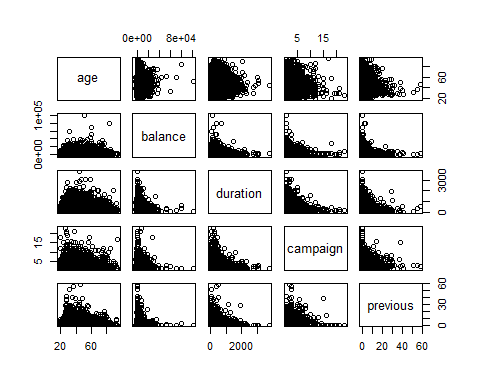
par(mfrow=c(1,1))  
#The scree plots show the elbow around 7.The variance explained for 95% is 7   
#Use ggplot2 to plot the first few pc's  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full.2008$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC2)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



#Looking at PC1 and PC3  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full.2008$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC3)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



# Not great. Some separation. Still debating using day in there.  
# PCA w/o day & pday:  
# This includes: age, balance, duration, campaign, pdays, previous  
# We might want to do this again when we address normality for LDA/QDA.  
# Not necessary for PCA but slides indicate it may improve the results.  
# Without day and pdays  
reduced<-Bank\_Full.2008[,c(1,6,12,13,15)]  
pairs(reduced)



apply(reduced,2,summary)

## age balance duration campaign previous  
## Min. 18.00000 -4057.000 0.000 1.00000 0.000000  
## 1st Qu. 31.00000 144.000 112.000 1.00000 0.000000  
## Median 37.00000 539.000 197.000 2.00000 0.000000  
## Mean 40.43602 1436.968 267.627 2.09353 1.333505  
## 3rd Qu. 48.00000 1536.000 336.000 2.00000 2.000000  
## Max. 95.00000 102127.000 3785.000 23.00000 58.000000

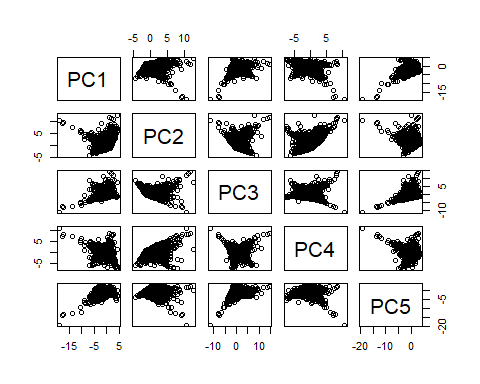
var.raw<-apply(reduced,2,var)  
var.raw

## age balance duration campaign previous   
## 1.512222e+02 9.346700e+06 6.357241e+04 2.762934e+00 7.568860e+00

#Total variance  
sum(var.raw)

## [1] 9410434

pc.result<-prcomp(Bank\_Full.2008[,c(1,6,12,13,15)],scale.=TRUE)  
pc.scores<-pc.result$x  
pairs(pc.scores)



cor(pc.scores)

## PC1 PC2 PC3 PC4 PC5  
## PC1 1.000000e+00 -6.805662e-15 5.950315e-15 4.730126e-15 -1.778636e-14  
## PC2 -6.805662e-15 1.000000e+00 -2.460516e-16 -3.073818e-15 8.955968e-15  
## PC3 5.950315e-15 -2.460516e-16 1.000000e+00 5.036443e-15 -4.070655e-15  
## PC4 4.730126e-15 -3.073818e-15 5.036443e-15 1.000000e+00 -7.440006e-15  
## PC5 -1.778636e-14 8.955968e-15 -4.070655e-15 -7.440006e-15 1.000000e+00

var.pca<-apply(pc.scores,2,var)  
var.pca

## PC1 PC2 PC3 PC4 PC5   
## 1.1546225 1.1099629 0.9705134 0.8895309 0.8753704

#Total Variance of PC's  
sum(var.pca)

## [1] 5

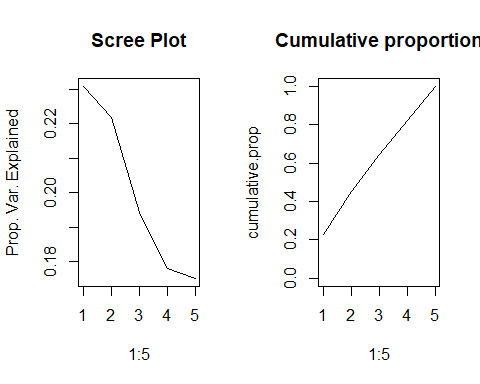
#Total Variance of Original Variables.  
sum(var.raw)

## [1] 9410434

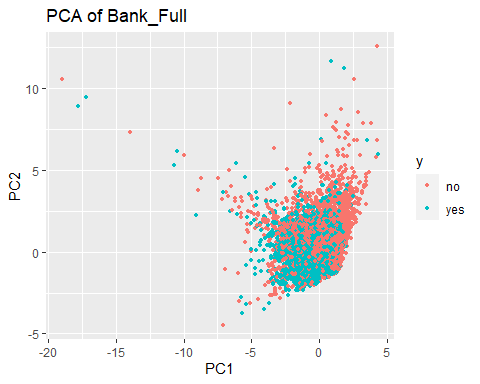
#List of eigenvectors  
pc.result$rotation

## PC1 PC2 PC3 PC4 PC5  
## age -0.58942529 0.3193297 -0.1749158 -0.2344791 0.6819313  
## balance -0.56345708 0.3263786 -0.3163908 0.3208739 -0.6106799  
## duration -0.48010419 -0.2769614 0.6434495 -0.4506033 -0.2751760  
## campaign 0.31408200 0.6204137 -0.0768656 -0.6628779 -0.2666906  
## previous 0.07711298 0.5743560 0.6703475 0.4467763 0.1232638

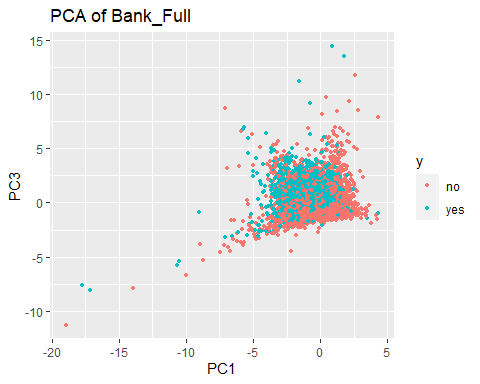
#A scree plot of the eigenvalues used to determine how many pc's to keep can be plotted in the following way:  
par(mfrow=c(1,2))  
eigenvals<-(pc.result$sdev)^2  
plot(1:5,eigenvals/sum(eigenvals),type="l",main="Scree Plot",ylab="Prop. Var. Explained")  
cumulative.prop<-cumsum(eigenvals/sum(eigenvals))  
plot(1:5,cumulative.prop,type="l",main="Cumulative proportion",ylim=c(0,1))



par(mfrow=c(1,1))  
#Removing day and pdays The scree plots show the elbow around 4 predictors.The variance explained for 85 of the variance is 4.   
#Use ggplot2 to plot the first few pc's  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full.2008$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC2)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



#Looking at PC1 and PC3  
pc.scores<-data.frame(pc.scores)  
pc.scores$y<-Bank\_Full.2008$y  
ggplot(data = pc.scores, aes(x = PC1, y = PC3)) +  
 geom\_point(aes(col=y), size=1)+  
 ggtitle("PCA of Bank\_Full")



#Without day and pday we find that the model requires first 4 principle components or combinations of these to explain about 80% of the variability withot considering year 2008 data.  
#This looks better

# train test split  
#Using the data without 2008 as excluding it makes the dataset more balanced than when we have 2008 as we know that 2008 had financial crisis and because of which we see that the data is not good to include in prediction.  
# 80/20 would be: 13985:3496  
set.seed(1234)  
index<-sample(1:dim(Bank\_Full)[1],3496,replace=F)  
test<-Bank\_Full[index,]  
train<-Bank\_Full[-index,]  
  
Bank\_Full.2008$year = droplevels(Bank\_Full.2008$year)  
  
set.seed(1235)  
index<-sample(1:dim(Bank\_Full.2008)[1],3496,replace=F)  
test2008<-Bank\_Full.2008[index,]  
train2008<-Bank\_Full.2008[-index,]

# Using glm  
  
set.seed(1234)  
  
train<-na.omit(train)  
train2008<-na.omit(train2008)  
  
#With all Years  
Model\_Full<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year,data=train,family= binomial(link="logit"))  
(vif(Model\_Full)[,3])^2

## age job marital education default balance housing loan   
## 2.269021 1.144955 1.211293 1.311367 1.020542 1.043356 1.454897 1.064396   
## contact day month duration campaign pdays previous poutcome   
## 1.566519 1.330861 1.172618 1.204775 1.118471 3.290195 1.748483 1.649036   
## year   
## 2.031722

#On letting all the predictors in the model and running the vif function, nothing stands out in terms of excluding the predictors from the model with a VIF score > 10  
  
#Summary of current fit  
summary(Model\_Full)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6914 -0.3375 -0.2098 -0.1320 3.1856   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.885e+00 2.428e-01 -28.351 < 2e-16 \*\*\*  
## age -1.254e-03 2.359e-03 -0.531 0.595108   
## jobblue-collar -1.988e-01 7.798e-02 -2.549 0.010789 \*   
## jobentrepreneur -2.157e-01 1.345e-01 -1.604 0.108670   
## jobhousemaid -5.391e-01 1.496e-01 -3.605 0.000312 \*\*\*  
## jobmanagement -4.681e-02 7.864e-02 -0.595 0.551676   
## jobretired 2.661e-02 1.067e-01 0.249 0.803092   
## jobself-employed -2.016e-01 1.215e-01 -1.659 0.097131 .   
## jobservices -1.699e-01 9.094e-02 -1.868 0.061775 .   
## jobstudent 3.528e-02 1.157e-01 0.305 0.760310   
## jobtechnician -4.577e-02 7.396e-02 -0.619 0.536041   
## jobunemployed -2.314e-01 1.211e-01 -1.911 0.055973 .   
## jobunknown -2.675e-01 2.500e-01 -1.070 0.284555   
## maritalmarried -2.278e-01 6.326e-02 -3.601 0.000317 \*\*\*  
## maritalsingle -4.299e-02 7.253e-02 -0.593 0.553339   
## educationsecondary 1.945e-01 6.979e-02 2.787 0.005322 \*\*   
## educationtertiary 3.514e-01 8.100e-02 4.338 1.44e-05 \*\*\*  
## educationunknown 2.371e-01 1.114e-01 2.129 0.033268 \*   
## defaultyes 2.315e-01 1.715e-01 1.350 0.176993   
## balance 9.737e-06 5.699e-06 1.708 0.087552 .   
## housingyes -4.854e-01 4.750e-02 -10.219 < 2e-16 \*\*\*  
## loanyes -2.742e-01 6.434e-02 -4.262 2.03e-05 \*\*\*  
## contacttelephone -2.140e-01 8.171e-02 -2.619 0.008809 \*\*   
## contactunknown -1.027e-01 8.746e-02 -1.174 0.240277   
## day 2.142e-02 2.651e-03 8.083 6.34e-16 \*\*\*  
## monthfeb 1.347e+00 1.366e-01 9.865 < 2e-16 \*\*\*  
## monthmar 2.797e+00 1.589e-01 17.601 < 2e-16 \*\*\*  
## monthapr 1.451e+00 1.258e-01 11.532 < 2e-16 \*\*\*  
## monthmay 1.190e+00 1.247e-01 9.547 < 2e-16 \*\*\*  
## monthjun 2.246e+00 1.385e-01 16.219 < 2e-16 \*\*\*  
## monthjul 2.060e+00 1.346e-01 15.309 < 2e-16 \*\*\*  
## monthaug 2.131e+00 1.316e-01 16.191 < 2e-16 \*\*\*  
## monthsep 2.204e+00 1.560e-01 14.126 < 2e-16 \*\*\*  
## monthoct 2.454e+00 1.468e-01 16.719 < 2e-16 \*\*\*  
## monthnov 2.253e+00 1.419e-01 15.882 < 2e-16 \*\*\*  
## monthdec 2.607e+00 2.094e-01 12.448 < 2e-16 \*\*\*  
## duration 4.521e-03 7.129e-05 63.412 < 2e-16 \*\*\*  
## campaign -4.752e-02 1.039e-02 -4.572 4.84e-06 \*\*\*  
## pdays -7.411e-04 3.075e-04 -2.410 0.015952 \*   
## previous -8.041e-03 1.165e-02 -0.690 0.490174   
## poutcomeother 2.074e-01 9.677e-02 2.143 0.032116 \*   
## poutcomesuccess 1.899e+00 8.807e-02 21.560 < 2e-16 \*\*\*  
## poutcomeunknown 3.293e-01 1.018e-01 3.234 0.001219 \*\*   
## year2009 2.193e+00 7.527e-02 29.140 < 2e-16 \*\*\*  
## year2010 3.289e+00 8.641e-02 38.064 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18477 on 41668 degrees of freedom  
## AIC: 18567  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18477 on 41668 degrees of freedom  
#AIC: 18567  
#Call:  
#glm(formula = as.factor(y) ~ age + job + marital + education +   
# default + balance + housing + loan + contact + day + month +   
# duration + campaign + pdays + previous + poutcome + year,   
# family = binomial(link = "logit"), data = train)  
  
  
#Hosmer Lemeshow test for lack of fit.Use as needed.The g=10 is an option that deals with the continuous predictors if any are there.  
#This should be increased with caution.   
hoslem.test(Model\_Full$y, fitted(Model\_Full), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Full$y, fitted(Model\_Full)  
## X-squared = 536.81, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
# Using the summary coefficients we can generate CI for each one in the table showing odds ratio  
exp(cbind("Odds ratio" = coef(Model\_Full), confint.default(Model\_Full, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.001022899 6.355059e-04 0.001646441  
## age 0.998746882 9.941390e-01 1.003376141  
## jobblue-collar 0.819717277 7.035431e-01 0.955075016  
## jobentrepreneur 0.805970575 6.192453e-01 1.049000470  
## jobhousemaid 0.583244684 4.350532e-01 0.781914465  
## jobmanagement 0.954268677 8.179614e-01 1.113290600  
## jobretired 1.026966098 8.331450e-01 1.265877315  
## jobself-employed 0.817454791 6.442324e-01 1.037253564  
## jobservices 0.843776573 7.060234e-01 1.008406965  
## jobstudent 1.035913380 8.258045e-01 1.299480046  
## jobtechnician 0.955263840 8.263573e-01 1.104278949  
## jobunemployed 0.793419386 6.258123e-01 1.005915546  
## jobunknown 0.765264852 4.688276e-01 1.249137958  
## maritalmarried 0.796268461 7.034118e-01 0.901383018  
## maritalsingle 0.957920158 8.309886e-01 1.104240264  
## educationsecondary 1.214713978 1.059412e+00 1.392781928  
## educationtertiary 1.420992843 1.212404e+00 1.665468973  
## educationunknown 1.267592861 1.018988e+00 1.576850623  
## defaultyes 1.260541038 9.006862e-01 1.764170225  
## balance 1.000009737 9.999986e-01 1.000020907  
## housingyes 0.615422722 5.607098e-01 0.675474405  
## loanyes 0.760174625 6.701079e-01 0.862346835  
## contacttelephone 0.807327511 6.878582e-01 0.947546566  
## contactunknown 0.902398674 7.602492e-01 1.071126910  
## day 1.021654463 1.016361e+00 1.026975750  
## monthfeb 3.846465464 2.943217e+00 5.026914075  
## monthmar 16.401928392 1.201196e+01 22.396290863  
## monthapr 4.265582796 3.333539e+00 5.458221433  
## monthmay 3.288049683 2.575218e+00 4.198197033  
## monthjun 9.448089302 7.202394e+00 12.393988257  
## monthjul 7.846587650 6.027515e+00 10.214646423  
## monthaug 8.425311268 6.509403e+00 10.905127045  
## monthsep 9.058696045 6.672311e+00 12.298582598  
## monthoct 11.638737721 8.728702e+00 15.518941748  
## monthnov 9.517976147 7.207521e+00 12.569074770  
## monthdec 13.551620098 8.989939e+00 20.427993277  
## duration 1.004531166 1.004391e+00 1.004671543  
## campaign 0.953590193 9.343592e-01 0.973217027  
## pdays 0.999259187 9.986571e-01 0.999861615  
## previous 0.991990872 9.695899e-01 1.014909347  
## poutcomeother 1.230430718 1.017866e+00 1.487385991  
## poutcomesuccess 6.678519372 5.619680e+00 7.936861815  
## poutcomeunknown 1.389949407 1.138536e+00 1.696879944  
## year2009 8.966456047 7.736538e+00 10.391900956  
## year2010 26.820110374 2.264156e+01 31.769817150

#stepwise:  
#This starts with a null model and then builds up using forward selection up to all the predictors that were specified in main model previously.  
# Code from AutoClassify.R  
Model\_Step<-Model\_Full %>%stepAIC(trace=FALSE)  
summary(Model\_Step)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + poutcome + year, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6855 -0.3370 -0.2098 -0.1322 3.1866   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.964e+00 2.122e-01 -32.821 < 2e-16 \*\*\*  
## jobblue-collar -1.954e-01 7.791e-02 -2.507 0.012167 \*   
## jobentrepreneur -2.095e-01 1.342e-01 -1.561 0.118532   
## jobhousemaid -5.437e-01 1.489e-01 -3.651 0.000261 \*\*\*  
## jobmanagement -4.689e-02 7.856e-02 -0.597 0.550589   
## jobretired 3.817e-03 9.546e-02 0.040 0.968100   
## jobself-employed -2.002e-01 1.215e-01 -1.648 0.099453 .   
## jobservices -1.672e-01 9.089e-02 -1.840 0.065761 .   
## jobstudent 4.879e-02 1.134e-01 0.430 0.667021   
## jobtechnician -4.461e-02 7.394e-02 -0.603 0.546229   
## jobunemployed -2.307e-01 1.210e-01 -1.906 0.056656 .   
## jobunknown -2.745e-01 2.497e-01 -1.100 0.271535   
## maritalmarried -2.270e-01 6.299e-02 -3.603 0.000314 \*\*\*  
## maritalsingle -3.046e-02 6.793e-02 -0.448 0.653908   
## educationsecondary 1.989e-01 6.947e-02 2.863 0.004193 \*\*   
## educationtertiary 3.565e-01 8.041e-02 4.433 9.28e-06 \*\*\*  
## educationunknown 2.380e-01 1.113e-01 2.137 0.032561 \*   
## balance 9.135e-06 5.685e-06 1.607 0.108058   
## housingyes -4.849e-01 4.725e-02 -10.263 < 2e-16 \*\*\*  
## loanyes -2.690e-01 6.419e-02 -4.190 2.79e-05 \*\*\*  
## contacttelephone -2.237e-01 8.060e-02 -2.776 0.005507 \*\*   
## contactunknown -1.023e-01 8.747e-02 -1.170 0.242092   
## day 2.147e-02 2.650e-03 8.104 5.30e-16 \*\*\*  
## monthfeb 1.347e+00 1.365e-01 9.865 < 2e-16 \*\*\*  
## monthmar 2.795e+00 1.588e-01 17.594 < 2e-16 \*\*\*  
## monthapr 1.449e+00 1.257e-01 11.528 < 2e-16 \*\*\*  
## monthmay 1.189e+00 1.246e-01 9.542 < 2e-16 \*\*\*  
## monthjun 2.243e+00 1.384e-01 16.206 < 2e-16 \*\*\*  
## monthjul 2.060e+00 1.345e-01 15.316 < 2e-16 \*\*\*  
## monthaug 2.127e+00 1.315e-01 16.170 < 2e-16 \*\*\*  
## monthsep 2.201e+00 1.559e-01 14.114 < 2e-16 \*\*\*  
## monthoct 2.448e+00 1.466e-01 16.696 < 2e-16 \*\*\*  
## monthnov 2.250e+00 1.418e-01 15.864 < 2e-16 \*\*\*  
## monthdec 2.602e+00 2.093e-01 12.434 < 2e-16 \*\*\*  
## duration 4.520e-03 7.128e-05 63.406 < 2e-16 \*\*\*  
## campaign -4.785e-02 1.039e-02 -4.604 4.15e-06 \*\*\*  
## pdays -7.227e-04 3.066e-04 -2.358 0.018398 \*   
## poutcomeother 2.043e-01 9.649e-02 2.117 0.034242 \*   
## poutcomesuccess 1.900e+00 8.804e-02 21.585 < 2e-16 \*\*\*  
## poutcomeunknown 3.576e-01 9.442e-02 3.788 0.000152 \*\*\*  
## year2009 2.190e+00 7.523e-02 29.115 < 2e-16 \*\*\*  
## year2010 3.279e+00 8.594e-02 38.151 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18480 on 41671 degrees of freedom  
## AIC: 18564  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18480 on 41671 degrees of freedom  
#AIC: 18564  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + balance +   
# housing + loan + contact + day + month + duration + campaign +   
# pdays + poutcome + year, family = binomial(link = "logit"),   
# data = train)  
(vif(Model\_Step)[,3])^2

## job marital education balance housing loan contact day   
## 1.107318 1.091406 1.302182 1.034893 1.439287 1.059360 1.545652 1.330023   
## month duration campaign pdays poutcome year   
## 1.171876 1.204378 1.115711 3.272684 1.533005 2.011946

hoslem.test(Model\_Step$y, fitted(Model\_Step), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Step$y, fitted(Model\_Step)  
## X-squared = 529.16, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Step), confint.default(Model\_Step, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 9.451513e-04 0.000623578 0.001432557  
## jobblue-collar 8.225438e-01 0.706054104 0.958252812  
## jobentrepreneur 8.110128e-01 0.623450475 1.055002377  
## jobhousemaid 5.806017e-01 0.433626148 0.777394048  
## jobmanagement 9.541945e-01 0.818032883 1.113020289  
## jobretired 1.003825e+00 0.832539550 1.210349912  
## jobself-employed 8.185999e-01 0.645145106 1.038689995  
## jobservices 8.459980e-01 0.707954669 1.010958351  
## jobstudent 1.049996e+00 0.840751271 1.311316352  
## jobtechnician 9.563666e-01 0.827352529 1.105498498  
## jobunemployed 7.939821e-01 0.626300107 1.006558217  
## jobunknown 7.599393e-01 0.465867911 1.239638503  
## maritalmarried 7.969248e-01 0.704362133 0.901651455  
## maritalsingle 9.700036e-01 0.849086024 1.108141033  
## educationsecondary 1.220074e+00 1.064763209 1.398038439  
## educationtertiary 1.428312e+00 1.220043714 1.672133723  
## educationunknown 1.268674e+00 1.019957936 1.578038457  
## balance 1.000009e+00 0.999997993 1.000020277  
## housingyes 6.157788e-01 0.561318144 0.675523402  
## loanyes 7.641482e-01 0.673807469 0.866601394  
## contacttelephone 7.995305e-01 0.682696168 0.936359360  
## contactunknown 9.027425e-01 0.760520273 1.071561129  
## day 1.021705e+00 1.016412930 1.027024374  
## monthfeb 3.844106e+00 2.941768178 5.023220227  
## monthmar 1.635863e+01 11.982171927 22.333576359  
## monthapr 4.259565e+00 3.329356839 5.449669976  
## monthmay 3.283657e+00 2.572122473 4.192026308  
## monthjun 9.421325e+00 7.182968245 12.357199735  
## monthjul 7.846721e+00 6.028349745 10.213580823  
## monthaug 8.390774e+00 6.483794036 10.858626287  
## monthsep 9.033785e+00 6.654751231 12.263307783  
## monthoct 1.156821e+01 8.678543321 15.420042263  
## monthnov 9.483305e+00 7.182235538 12.521597976  
## monthdec 1.349713e+01 8.955337551 20.342329146  
## duration 1.004530e+00 1.004389559 1.004670243  
## campaign 9.532795e-01 0.934058093 0.972896459  
## pdays 9.992776e-01 0.998677334 0.999878136  
## poutcomeother 1.226642e+00 1.015288106 1.481993367  
## poutcomesuccess 6.687445e+00 5.627602242 7.946887479  
## poutcomeunknown 1.429953e+00 1.188360395 1.720661204  
## year2009 8.938763e+00 7.713302627 10.358918996  
## year2010 2.654431e+01 22.429346569 31.414217089

Model\_Null<-glm(as.factor(y) ~ 1, data=train,family = binomial(link="logit"))  
#Forward:  
Model\_FWD<-stepAIC(Model\_Null,  
 scope = list(upper=Model\_Full),  
 direction="forward",  
 test="Chisq",  
 data=train)

## Start: AIC=30189.9  
## as.factor(y) ~ 1  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + duration 1 25393 25397 4794.6 < 2.2e-16 \*\*\*  
## + year 2 26190 26196 3998.3 < 2.2e-16 \*\*\*  
## + poutcome 3 27754 27762 2434.3 < 2.2e-16 \*\*\*  
## + month 11 28152 28176 2036.0 < 2.2e-16 \*\*\*  
## + contact 2 29063 29069 1124.8 < 2.2e-16 \*\*\*  
## + housing 1 29375 29379 812.7 < 2.2e-16 \*\*\*  
## + job 11 29509 29533 678.9 < 2.2e-16 \*\*\*  
## + previous 1 29819 29823 368.9 < 2.2e-16 \*\*\*  
## + pdays 1 29824 29828 364.3 < 2.2e-16 \*\*\*  
## + campaign 1 29858 29862 330.2 < 2.2e-16 \*\*\*  
## + loan 1 29963 29967 225.4 < 2.2e-16 \*\*\*  
## + education 3 29964 29972 223.6 < 2.2e-16 \*\*\*  
## + marital 2 30006 30012 181.5 < 2.2e-16 \*\*\*  
## + balance 1 30095 30099 92.5 < 2.2e-16 \*\*\*  
## + day 1 30158 30162 30.1 4.111e-08 \*\*\*  
## + age 1 30166 30170 21.7 3.208e-06 \*\*\*  
## + default 1 30167 30171 20.9 4.789e-06 \*\*\*  
## <none> 30188 30190   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=25397.25  
## as.factor(y) ~ duration  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + year 2 20823 20831 4570.4 < 2.2e-16 \*\*\*  
## + poutcome 3 22795 22805 2597.9 < 2.2e-16 \*\*\*  
## + month 11 23089 23115 2304.7 < 2.2e-16 \*\*\*  
## + contact 2 24096 24104 1297.4 < 2.2e-16 \*\*\*  
## + housing 1 24371 24377 1022.3 < 2.2e-16 \*\*\*  
## + job 11 24588 24614 805.2 < 2.2e-16 \*\*\*  
## + pdays 1 24926 24932 467.4 < 2.2e-16 \*\*\*  
## + previous 1 24934 24940 459.2 < 2.2e-16 \*\*\*  
## + campaign 1 25119 25125 274.1 < 2.2e-16 \*\*\*  
## + education 3 25136 25146 257.8 < 2.2e-16 \*\*\*  
## + loan 1 25154 25160 239.7 < 2.2e-16 \*\*\*  
## + marital 2 25224 25232 169.6 < 2.2e-16 \*\*\*  
## + balance 1 25316 25322 76.9 < 2.2e-16 \*\*\*  
## + age 1 25366 25372 27.8 1.350e-07 \*\*\*  
## + day 1 25372 25378 20.9 4.728e-06 \*\*\*  
## + default 1 25376 25382 17.4 3.045e-05 \*\*\*  
## <none> 25393 25397   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=20830.86  
## as.factor(y) ~ duration + year  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + month 11 19552 19582 1270.97 < 2.2e-16 \*\*\*  
## + poutcome 3 19967 19981 855.59 < 2.2e-16 \*\*\*  
## + housing 1 20245 20255 577.85 < 2.2e-16 \*\*\*  
## + job 11 20563 20593 260.01 < 2.2e-16 \*\*\*  
## + education 3 20668 20682 155.30 < 2.2e-16 \*\*\*  
## + contact 2 20712 20724 110.54 < 2.2e-16 \*\*\*  
## + loan 1 20754 20764 68.87 < 2.2e-16 \*\*\*  
## + pdays 1 20767 20777 55.54 9.151e-14 \*\*\*  
## + balance 1 20780 20790 42.83 5.963e-11 \*\*\*  
## + marital 2 20779 20791 43.58 3.437e-10 \*\*\*  
## + campaign 1 20802 20812 21.20 4.133e-06 \*\*\*  
## + day 1 20803 20813 19.68 9.169e-06 \*\*\*  
## + age 1 20818 20828 5.14 0.02333 \*   
## <none> 20823 20831   
## + previous 1 20822 20832 1.02 0.31231   
## + default 1 20823 20833 0.01 0.92716   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=19581.89  
## as.factor(y) ~ duration + year + month  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + poutcome 3 18886 18922 665.48 < 2.2e-16 \*\*\*  
## + housing 1 19360 19392 192.05 < 2.2e-16 \*\*\*  
## + day 1 19467 19499 85.34 < 2.2e-16 \*\*\*  
## + job 11 19447 19499 104.52 < 2.2e-16 \*\*\*  
## + education 3 19467 19503 84.74 < 2.2e-16 \*\*\*  
## + marital 2 19495 19529 56.44 5.537e-13 \*\*\*  
## + pdays 1 19505 19537 47.29 6.108e-12 \*\*\*  
## + loan 1 19511 19543 41.23 1.354e-10 \*\*\*  
## + campaign 1 19528 19560 24.20 8.669e-07 \*\*\*  
## + contact 2 19536 19570 16.04 0.0003281 \*\*\*  
## + balance 1 19543 19575 8.87 0.0028942 \*\*   
## + previous 1 19547 19579 5.22 0.0222836 \*   
## + age 1 19548 19580 4.16 0.0414694 \*   
## <none> 19552 19582   
## + default 1 19551 19583 0.70 0.4031484   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18922.41  
## as.factor(y) ~ duration + year + month + poutcome  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + housing 1 18736 18774 150.252 < 2.2e-16 \*\*\*  
## + day 1 18807 18845 79.835 < 2.2e-16 \*\*\*  
## + job 11 18791 18849 95.740 1.241e-15 \*\*\*  
## + education 3 18813 18855 73.223 8.707e-16 \*\*\*  
## + marital 2 18831 18871 55.955 7.071e-13 \*\*\*  
## + loan 1 18857 18895 29.483 5.640e-08 \*\*\*  
## + campaign 1 18867 18905 19.051 1.272e-05 \*\*\*  
## + pdays 1 18870 18908 16.182 5.754e-05 \*\*\*  
## + contact 2 18873 18913 13.499 0.001171 \*\*   
## + balance 1 18879 18917 7.932 0.004856 \*\*   
## + age 1 18881 18919 5.550 0.018482 \*   
## + previous 1 18884 18922 2.366 0.124031   
## <none> 18886 18922   
## + default 1 18885 18923 0.993 0.318896   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18774.16  
## as.factor(y) ~ duration + year + month + poutcome + housing  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + day 1 18670 18710 65.868 4.822e-16 \*\*\*  
## + education 3 18667 18711 69.701 4.945e-15 \*\*\*  
## + job 11 18659 18719 77.336 4.821e-12 \*\*\*  
## + marital 2 18687 18729 49.071 2.210e-11 \*\*\*  
## + loan 1 18714 18754 21.834 2.973e-06 \*\*\*  
## + campaign 1 18718 18758 17.949 2.269e-05 \*\*\*  
## + contact 2 18717 18759 19.309 6.414e-05 \*\*\*  
## + age 1 18720 18760 15.742 7.259e-05 \*\*\*  
## + pdays 1 18729 18769 7.696 0.005534 \*\*   
## + balance 1 18732 18772 4.245 0.039368 \*   
## <none> 18736 18774   
## + previous 1 18735 18775 1.173 0.278700   
## + default 1 18735 18775 0.951 0.329356   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18710.29  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + education 3 18603 18649 67.704 1.324e-14 \*\*\*  
## + job 11 18597 18659 73.472 2.660e-11 \*\*\*  
## + marital 2 18623 18667 47.197 5.641e-11 \*\*\*  
## + campaign 1 18644 18686 26.688 2.391e-07 \*\*\*  
## + loan 1 18650 18692 20.107 7.322e-06 \*\*\*  
## + contact 2 18652 18696 18.626 9.024e-05 \*\*\*  
## + age 1 18655 18697 15.595 7.848e-05 \*\*\*  
## + pdays 1 18663 18705 7.217 0.00722 \*\*   
## + balance 1 18666 18708 3.841 0.05001 .   
## <none> 18670 18710   
## + default 1 18669 18711 0.972 0.32411   
## + previous 1 18669 18711 0.865 0.35235   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18648.59  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + marital 2 18569 18619 33.476 5.381e-08 \*\*\*  
## + campaign 1 18576 18624 26.741 2.327e-07 \*\*\*  
## + loan 1 18583 18631 19.149 1.209e-05 \*\*\*  
## + job 11 18568 18636 34.821 0.0002652 \*\*\*  
## + contact 2 18591 18641 11.262 0.0035858 \*\*   
## + pdays 1 18596 18644 6.302 0.0120593 \*   
## + age 1 18598 18646 4.400 0.0359392 \*   
## + balance 1 18600 18648 2.665 0.1025856   
## <none> 18603 18649   
## + default 1 18602 18650 1.049 0.3057907   
## + previous 1 18602 18650 0.811 0.3679136   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18619.11  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + campaign 1 18544 18596 25.1674 5.256e-07 \*\*\*  
## + loan 1 18551 18603 18.2360 1.952e-05 \*\*\*  
## + job 11 18540 18612 28.7243 0.002505 \*\*   
## + contact 2 18560 18614 9.3949 0.009118 \*\*   
## + pdays 1 18562 18614 6.7286 0.009488 \*\*   
## + balance 1 18566 18618 3.4572 0.062978 .   
## <none> 18569 18619   
## + default 1 18568 18620 0.7646 0.381887   
## + previous 1 18568 18620 0.7541 0.385171   
## + age 1 18569 18621 0.1263 0.722320   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18595.94  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + loan 1 18525 18579 18.7948 1.456e-05 \*\*\*  
## + job 11 18515 18589 28.5941 0.002623 \*\*   
## + pdays 1 18537 18591 6.5521 0.010476 \*   
## + contact 2 18536 18592 7.6479 0.021842 \*   
## + balance 1 18541 18595 3.4673 0.062594 .   
## <none> 18544 18596   
## + default 1 18543 18597 0.7855 0.375464   
## + previous 1 18544 18598 0.3769 0.539256   
## + age 1 18544 18598 0.1064 0.744280   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18579.15  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign + loan  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + job 11 18497 18573 28.4914 0.002721 \*\*  
## + pdays 1 18519 18575 6.3479 0.011752 \*   
## + contact 2 18517 18575 8.3246 0.015572 \*   
## + balance 1 18523 18579 2.5308 0.111641   
## <none> 18525 18579   
## + default 1 18524 18580 1.3769 0.240638   
## + previous 1 18525 18581 0.3100 0.577696   
## + age 1 18525 18581 0.2113 0.645716   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18572.66  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign + loan + job  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + contact 2 18488 18568 8.9344 0.01148 \*  
## + pdays 1 18491 18569 5.9022 0.01512 \*  
## + balance 1 18494 18572 2.3468 0.12554   
## <none> 18497 18573   
## + default 1 18495 18573 1.5944 0.20670   
## + age 1 18496 18574 0.7432 0.38865   
## + previous 1 18496 18574 0.3629 0.54693   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18567.72  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign + loan + job + contact  
##   
## Df Deviance AIC LRT Pr(Chi)   
## + pdays 1 18482 18564 5.6593 0.01736 \*  
## + balance 1 18485 18567 2.5676 0.10908   
## <none> 18488 18568   
## + default 1 18486 18568 1.4958 0.22131   
## + previous 1 18487 18569 0.2981 0.58510   
## + age 1 18488 18570 0.1673 0.68252   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=18564.06  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign + loan + job + contact +   
## pdays  
##   
## Df Deviance AIC LRT Pr(Chi)  
## + balance 1 18480 18564 2.50890 0.1132  
## <none> 18482 18564   
## + default 1 18481 18565 1.56023 0.2116  
## + previous 1 18482 18566 0.48123 0.4879  
## + age 1 18482 18566 0.17307 0.6774  
##   
## Step: AIC=18563.55  
## as.factor(y) ~ duration + year + month + poutcome + housing +   
## day + education + marital + campaign + loan + job + contact +   
## pdays + balance  
##   
## Df Deviance AIC LRT Pr(Chi)  
## <none> 18480 18564   
## + default 1 18478 18564 1.75217 0.1856  
## + previous 1 18479 18565 0.49476 0.4818  
## + age 1 18479 18565 0.28742 0.5919

summary(Model\_FWD)

##   
## Call:  
## glm(formula = as.factor(y) ~ duration + year + month + poutcome +   
## housing + day + education + marital + campaign + loan + job +   
## contact + pdays + balance, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6855 -0.3370 -0.2098 -0.1322 3.1866   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.964e+00 2.122e-01 -32.821 < 2e-16 \*\*\*  
## duration 4.520e-03 7.128e-05 63.406 < 2e-16 \*\*\*  
## year2009 2.190e+00 7.523e-02 29.115 < 2e-16 \*\*\*  
## year2010 3.279e+00 8.594e-02 38.151 < 2e-16 \*\*\*  
## monthfeb 1.347e+00 1.365e-01 9.865 < 2e-16 \*\*\*  
## monthmar 2.795e+00 1.588e-01 17.594 < 2e-16 \*\*\*  
## monthapr 1.449e+00 1.257e-01 11.528 < 2e-16 \*\*\*  
## monthmay 1.189e+00 1.246e-01 9.542 < 2e-16 \*\*\*  
## monthjun 2.243e+00 1.384e-01 16.206 < 2e-16 \*\*\*  
## monthjul 2.060e+00 1.345e-01 15.316 < 2e-16 \*\*\*  
## monthaug 2.127e+00 1.315e-01 16.170 < 2e-16 \*\*\*  
## monthsep 2.201e+00 1.559e-01 14.114 < 2e-16 \*\*\*  
## monthoct 2.448e+00 1.466e-01 16.696 < 2e-16 \*\*\*  
## monthnov 2.250e+00 1.418e-01 15.864 < 2e-16 \*\*\*  
## monthdec 2.602e+00 2.093e-01 12.434 < 2e-16 \*\*\*  
## poutcomeother 2.043e-01 9.649e-02 2.117 0.034242 \*   
## poutcomesuccess 1.900e+00 8.804e-02 21.585 < 2e-16 \*\*\*  
## poutcomeunknown 3.576e-01 9.442e-02 3.788 0.000152 \*\*\*  
## housingyes -4.849e-01 4.725e-02 -10.263 < 2e-16 \*\*\*  
## day 2.147e-02 2.650e-03 8.104 5.30e-16 \*\*\*  
## educationsecondary 1.989e-01 6.947e-02 2.863 0.004193 \*\*   
## educationtertiary 3.565e-01 8.041e-02 4.433 9.28e-06 \*\*\*  
## educationunknown 2.380e-01 1.113e-01 2.137 0.032561 \*   
## maritalmarried -2.270e-01 6.299e-02 -3.603 0.000314 \*\*\*  
## maritalsingle -3.046e-02 6.793e-02 -0.448 0.653908   
## campaign -4.785e-02 1.039e-02 -4.604 4.15e-06 \*\*\*  
## loanyes -2.690e-01 6.419e-02 -4.190 2.79e-05 \*\*\*  
## jobblue-collar -1.954e-01 7.791e-02 -2.507 0.012167 \*   
## jobentrepreneur -2.095e-01 1.342e-01 -1.561 0.118532   
## jobhousemaid -5.437e-01 1.489e-01 -3.651 0.000261 \*\*\*  
## jobmanagement -4.689e-02 7.856e-02 -0.597 0.550589   
## jobretired 3.817e-03 9.546e-02 0.040 0.968100   
## jobself-employed -2.002e-01 1.215e-01 -1.648 0.099453 .   
## jobservices -1.672e-01 9.089e-02 -1.840 0.065761 .   
## jobstudent 4.879e-02 1.134e-01 0.430 0.667021   
## jobtechnician -4.461e-02 7.394e-02 -0.603 0.546229   
## jobunemployed -2.307e-01 1.210e-01 -1.906 0.056656 .   
## jobunknown -2.745e-01 2.497e-01 -1.100 0.271535   
## contacttelephone -2.237e-01 8.060e-02 -2.776 0.005507 \*\*   
## contactunknown -1.023e-01 8.747e-02 -1.170 0.242092   
## pdays -7.227e-04 3.066e-04 -2.358 0.018398 \*   
## balance 9.135e-06 5.685e-06 1.607 0.108058   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18480 on 41671 degrees of freedom  
## AIC: 18564  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18480 on 41671 degrees of freedom  
#AIC: 18564  
#Call:  
#glm(formula = as.factor(y) ~ duration + year + month + poutcome +   
# housing + day + education + marital + campaign + loan + job +   
# contact + pdays + balance, family = binomial(link = "logit"),   
# data = train)  
(vif(Model\_FWD)[,3])^2

## duration year month poutcome housing day education marital   
## 1.204378 2.011946 1.171876 1.533005 1.439287 1.330023 1.302182 1.091406   
## campaign loan job contact pdays balance   
## 1.115711 1.059360 1.107318 1.545652 3.272684 1.034893

hoslem.test(Model\_FWD$y, fitted(Model\_FWD), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_FWD$y, fitted(Model\_FWD)  
## X-squared = 529.16, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_FWD), confint.default(Model\_FWD, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 9.451513e-04 0.000623578 0.001432557  
## duration 1.004530e+00 1.004389559 1.004670243  
## year2009 8.938763e+00 7.713302627 10.358918996  
## year2010 2.654431e+01 22.429346569 31.414217089  
## monthfeb 3.844106e+00 2.941768178 5.023220227  
## monthmar 1.635863e+01 11.982171927 22.333576359  
## monthapr 4.259565e+00 3.329356839 5.449669976  
## monthmay 3.283657e+00 2.572122473 4.192026308  
## monthjun 9.421325e+00 7.182968245 12.357199735  
## monthjul 7.846721e+00 6.028349745 10.213580823  
## monthaug 8.390774e+00 6.483794036 10.858626287  
## monthsep 9.033785e+00 6.654751231 12.263307783  
## monthoct 1.156821e+01 8.678543321 15.420042263  
## monthnov 9.483305e+00 7.182235538 12.521597976  
## monthdec 1.349713e+01 8.955337551 20.342329146  
## poutcomeother 1.226642e+00 1.015288106 1.481993367  
## poutcomesuccess 6.687445e+00 5.627602242 7.946887479  
## poutcomeunknown 1.429953e+00 1.188360395 1.720661204  
## housingyes 6.157788e-01 0.561318144 0.675523402  
## day 1.021705e+00 1.016412930 1.027024374  
## educationsecondary 1.220074e+00 1.064763209 1.398038439  
## educationtertiary 1.428312e+00 1.220043714 1.672133723  
## educationunknown 1.268674e+00 1.019957936 1.578038457  
## maritalmarried 7.969248e-01 0.704362133 0.901651455  
## maritalsingle 9.700036e-01 0.849086024 1.108141033  
## campaign 9.532795e-01 0.934058093 0.972896459  
## loanyes 7.641482e-01 0.673807469 0.866601394  
## jobblue-collar 8.225438e-01 0.706054104 0.958252812  
## jobentrepreneur 8.110128e-01 0.623450475 1.055002377  
## jobhousemaid 5.806017e-01 0.433626148 0.777394048  
## jobmanagement 9.541945e-01 0.818032883 1.113020289  
## jobretired 1.003825e+00 0.832539550 1.210349912  
## jobself-employed 8.185999e-01 0.645145106 1.038689995  
## jobservices 8.459980e-01 0.707954669 1.010958351  
## jobstudent 1.049996e+00 0.840751271 1.311316352  
## jobtechnician 9.563666e-01 0.827352529 1.105498498  
## jobunemployed 7.939821e-01 0.626300107 1.006558217  
## jobunknown 7.599393e-01 0.465867911 1.239638503  
## contacttelephone 7.995305e-01 0.682696168 0.936359360  
## contactunknown 9.027425e-01 0.760520273 1.071561129  
## pdays 9.992776e-01 0.998677334 0.999878136  
## balance 1.000009e+00 0.999997993 1.000020277

#Backward:  
Model\_Bwd<-stepAIC(Model\_Full,direction="backward",trace=FALSE)  
summary(Model\_Bwd)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + poutcome + year, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6855 -0.3370 -0.2098 -0.1322 3.1866   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.964e+00 2.122e-01 -32.821 < 2e-16 \*\*\*  
## jobblue-collar -1.954e-01 7.791e-02 -2.507 0.012167 \*   
## jobentrepreneur -2.095e-01 1.342e-01 -1.561 0.118532   
## jobhousemaid -5.437e-01 1.489e-01 -3.651 0.000261 \*\*\*  
## jobmanagement -4.689e-02 7.856e-02 -0.597 0.550589   
## jobretired 3.817e-03 9.546e-02 0.040 0.968100   
## jobself-employed -2.002e-01 1.215e-01 -1.648 0.099453 .   
## jobservices -1.672e-01 9.089e-02 -1.840 0.065761 .   
## jobstudent 4.879e-02 1.134e-01 0.430 0.667021   
## jobtechnician -4.461e-02 7.394e-02 -0.603 0.546229   
## jobunemployed -2.307e-01 1.210e-01 -1.906 0.056656 .   
## jobunknown -2.745e-01 2.497e-01 -1.100 0.271535   
## maritalmarried -2.270e-01 6.299e-02 -3.603 0.000314 \*\*\*  
## maritalsingle -3.046e-02 6.793e-02 -0.448 0.653908   
## educationsecondary 1.989e-01 6.947e-02 2.863 0.004193 \*\*   
## educationtertiary 3.565e-01 8.041e-02 4.433 9.28e-06 \*\*\*  
## educationunknown 2.380e-01 1.113e-01 2.137 0.032561 \*   
## balance 9.135e-06 5.685e-06 1.607 0.108058   
## housingyes -4.849e-01 4.725e-02 -10.263 < 2e-16 \*\*\*  
## loanyes -2.690e-01 6.419e-02 -4.190 2.79e-05 \*\*\*  
## contacttelephone -2.237e-01 8.060e-02 -2.776 0.005507 \*\*   
## contactunknown -1.023e-01 8.747e-02 -1.170 0.242092   
## day 2.147e-02 2.650e-03 8.104 5.30e-16 \*\*\*  
## monthfeb 1.347e+00 1.365e-01 9.865 < 2e-16 \*\*\*  
## monthmar 2.795e+00 1.588e-01 17.594 < 2e-16 \*\*\*  
## monthapr 1.449e+00 1.257e-01 11.528 < 2e-16 \*\*\*  
## monthmay 1.189e+00 1.246e-01 9.542 < 2e-16 \*\*\*  
## monthjun 2.243e+00 1.384e-01 16.206 < 2e-16 \*\*\*  
## monthjul 2.060e+00 1.345e-01 15.316 < 2e-16 \*\*\*  
## monthaug 2.127e+00 1.315e-01 16.170 < 2e-16 \*\*\*  
## monthsep 2.201e+00 1.559e-01 14.114 < 2e-16 \*\*\*  
## monthoct 2.448e+00 1.466e-01 16.696 < 2e-16 \*\*\*  
## monthnov 2.250e+00 1.418e-01 15.864 < 2e-16 \*\*\*  
## monthdec 2.602e+00 2.093e-01 12.434 < 2e-16 \*\*\*  
## duration 4.520e-03 7.128e-05 63.406 < 2e-16 \*\*\*  
## campaign -4.785e-02 1.039e-02 -4.604 4.15e-06 \*\*\*  
## pdays -7.227e-04 3.066e-04 -2.358 0.018398 \*   
## poutcomeother 2.043e-01 9.649e-02 2.117 0.034242 \*   
## poutcomesuccess 1.900e+00 8.804e-02 21.585 < 2e-16 \*\*\*  
## poutcomeunknown 3.576e-01 9.442e-02 3.788 0.000152 \*\*\*  
## year2009 2.190e+00 7.523e-02 29.115 < 2e-16 \*\*\*  
## year2010 3.279e+00 8.594e-02 38.151 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18480 on 41671 degrees of freedom  
## AIC: 18564  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18480 on 41671 degrees of freedom  
#AIC: 18564  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + balance +   
# housing + loan + contact + day + month + duration + campaign +   
# pdays + poutcome + year, family = binomial(link = "logit"),   
# data = train)  
(vif(Model\_Bwd)[,3])^2

## job marital education balance housing loan contact day   
## 1.107318 1.091406 1.302182 1.034893 1.439287 1.059360 1.545652 1.330023   
## month duration campaign pdays poutcome year   
## 1.171876 1.204378 1.115711 3.272684 1.533005 2.011946

hoslem.test(Model\_Bwd$y, fitted(Model\_Bwd), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Bwd$y, fitted(Model\_Bwd)  
## X-squared = 529.16, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Bwd), confint.default(Model\_Bwd, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 9.451513e-04 0.000623578 0.001432557  
## jobblue-collar 8.225438e-01 0.706054104 0.958252812  
## jobentrepreneur 8.110128e-01 0.623450475 1.055002377  
## jobhousemaid 5.806017e-01 0.433626148 0.777394048  
## jobmanagement 9.541945e-01 0.818032883 1.113020289  
## jobretired 1.003825e+00 0.832539550 1.210349912  
## jobself-employed 8.185999e-01 0.645145106 1.038689995  
## jobservices 8.459980e-01 0.707954669 1.010958351  
## jobstudent 1.049996e+00 0.840751271 1.311316352  
## jobtechnician 9.563666e-01 0.827352529 1.105498498  
## jobunemployed 7.939821e-01 0.626300107 1.006558217  
## jobunknown 7.599393e-01 0.465867911 1.239638503  
## maritalmarried 7.969248e-01 0.704362133 0.901651455  
## maritalsingle 9.700036e-01 0.849086024 1.108141033  
## educationsecondary 1.220074e+00 1.064763209 1.398038439  
## educationtertiary 1.428312e+00 1.220043714 1.672133723  
## educationunknown 1.268674e+00 1.019957936 1.578038457  
## balance 1.000009e+00 0.999997993 1.000020277  
## housingyes 6.157788e-01 0.561318144 0.675523402  
## loanyes 7.641482e-01 0.673807469 0.866601394  
## contacttelephone 7.995305e-01 0.682696168 0.936359360  
## contactunknown 9.027425e-01 0.760520273 1.071561129  
## day 1.021705e+00 1.016412930 1.027024374  
## monthfeb 3.844106e+00 2.941768178 5.023220227  
## monthmar 1.635863e+01 11.982171927 22.333576359  
## monthapr 4.259565e+00 3.329356839 5.449669976  
## monthmay 3.283657e+00 2.572122473 4.192026308  
## monthjun 9.421325e+00 7.182968245 12.357199735  
## monthjul 7.846721e+00 6.028349745 10.213580823  
## monthaug 8.390774e+00 6.483794036 10.858626287  
## monthsep 9.033785e+00 6.654751231 12.263307783  
## monthoct 1.156821e+01 8.678543321 15.420042263  
## monthnov 9.483305e+00 7.182235538 12.521597976  
## monthdec 1.349713e+01 8.955337551 20.342329146  
## duration 1.004530e+00 1.004389559 1.004670243  
## campaign 9.532795e-01 0.934058093 0.972896459  
## pdays 9.992776e-01 0.998677334 0.999878136  
## poutcomeother 1.226642e+00 1.015288106 1.481993367  
## poutcomesuccess 6.687445e+00 5.627602242 7.946887479  
## poutcomeunknown 1.429953e+00 1.188360395 1.720661204  
## year2009 8.938763e+00 7.713302627 10.358918996  
## year2010 2.654431e+01 22.429346569 31.414217089

#Both:  
Model\_Both<-stepAIC(Model\_Full,direction="both",trace=FALSE)  
summary(Model\_Both)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + poutcome + year, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6855 -0.3370 -0.2098 -0.1322 3.1866   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.964e+00 2.122e-01 -32.821 < 2e-16 \*\*\*  
## jobblue-collar -1.954e-01 7.791e-02 -2.507 0.012167 \*   
## jobentrepreneur -2.095e-01 1.342e-01 -1.561 0.118532   
## jobhousemaid -5.437e-01 1.489e-01 -3.651 0.000261 \*\*\*  
## jobmanagement -4.689e-02 7.856e-02 -0.597 0.550589   
## jobretired 3.817e-03 9.546e-02 0.040 0.968100   
## jobself-employed -2.002e-01 1.215e-01 -1.648 0.099453 .   
## jobservices -1.672e-01 9.089e-02 -1.840 0.065761 .   
## jobstudent 4.879e-02 1.134e-01 0.430 0.667021   
## jobtechnician -4.461e-02 7.394e-02 -0.603 0.546229   
## jobunemployed -2.307e-01 1.210e-01 -1.906 0.056656 .   
## jobunknown -2.745e-01 2.497e-01 -1.100 0.271535   
## maritalmarried -2.270e-01 6.299e-02 -3.603 0.000314 \*\*\*  
## maritalsingle -3.046e-02 6.793e-02 -0.448 0.653908   
## educationsecondary 1.989e-01 6.947e-02 2.863 0.004193 \*\*   
## educationtertiary 3.565e-01 8.041e-02 4.433 9.28e-06 \*\*\*  
## educationunknown 2.380e-01 1.113e-01 2.137 0.032561 \*   
## balance 9.135e-06 5.685e-06 1.607 0.108058   
## housingyes -4.849e-01 4.725e-02 -10.263 < 2e-16 \*\*\*  
## loanyes -2.690e-01 6.419e-02 -4.190 2.79e-05 \*\*\*  
## contacttelephone -2.237e-01 8.060e-02 -2.776 0.005507 \*\*   
## contactunknown -1.023e-01 8.747e-02 -1.170 0.242092   
## day 2.147e-02 2.650e-03 8.104 5.30e-16 \*\*\*  
## monthfeb 1.347e+00 1.365e-01 9.865 < 2e-16 \*\*\*  
## monthmar 2.795e+00 1.588e-01 17.594 < 2e-16 \*\*\*  
## monthapr 1.449e+00 1.257e-01 11.528 < 2e-16 \*\*\*  
## monthmay 1.189e+00 1.246e-01 9.542 < 2e-16 \*\*\*  
## monthjun 2.243e+00 1.384e-01 16.206 < 2e-16 \*\*\*  
## monthjul 2.060e+00 1.345e-01 15.316 < 2e-16 \*\*\*  
## monthaug 2.127e+00 1.315e-01 16.170 < 2e-16 \*\*\*  
## monthsep 2.201e+00 1.559e-01 14.114 < 2e-16 \*\*\*  
## monthoct 2.448e+00 1.466e-01 16.696 < 2e-16 \*\*\*  
## monthnov 2.250e+00 1.418e-01 15.864 < 2e-16 \*\*\*  
## monthdec 2.602e+00 2.093e-01 12.434 < 2e-16 \*\*\*  
## duration 4.520e-03 7.128e-05 63.406 < 2e-16 \*\*\*  
## campaign -4.785e-02 1.039e-02 -4.604 4.15e-06 \*\*\*  
## pdays -7.227e-04 3.066e-04 -2.358 0.018398 \*   
## poutcomeother 2.043e-01 9.649e-02 2.117 0.034242 \*   
## poutcomesuccess 1.900e+00 8.804e-02 21.585 < 2e-16 \*\*\*  
## poutcomeunknown 3.576e-01 9.442e-02 3.788 0.000152 \*\*\*  
## year2009 2.190e+00 7.523e-02 29.115 < 2e-16 \*\*\*  
## year2010 3.279e+00 8.594e-02 38.151 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18480 on 41671 degrees of freedom  
## AIC: 18564  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18480 on 41671 degrees of freedom  
#AIC: 18564  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + balance +   
# housing + loan + contact + day + month + duration + campaign +   
# pdays + poutcome + year, family = binomial(link = "logit"),   
# data = train)  
(vif(Model\_Both)[,3])^2

## job marital education balance housing loan contact day   
## 1.107318 1.091406 1.302182 1.034893 1.439287 1.059360 1.545652 1.330023   
## month duration campaign pdays poutcome year   
## 1.171876 1.204378 1.115711 3.272684 1.533005 2.011946

hoslem.test(Model\_Both$y, fitted(Model\_Both), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Both$y, fitted(Model\_Both)  
## X-squared = 529.16, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Both), confint.default(Model\_Both, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 9.451513e-04 0.000623578 0.001432557  
## jobblue-collar 8.225438e-01 0.706054104 0.958252812  
## jobentrepreneur 8.110128e-01 0.623450475 1.055002377  
## jobhousemaid 5.806017e-01 0.433626148 0.777394048  
## jobmanagement 9.541945e-01 0.818032883 1.113020289  
## jobretired 1.003825e+00 0.832539550 1.210349912  
## jobself-employed 8.185999e-01 0.645145106 1.038689995  
## jobservices 8.459980e-01 0.707954669 1.010958351  
## jobstudent 1.049996e+00 0.840751271 1.311316352  
## jobtechnician 9.563666e-01 0.827352529 1.105498498  
## jobunemployed 7.939821e-01 0.626300107 1.006558217  
## jobunknown 7.599393e-01 0.465867911 1.239638503  
## maritalmarried 7.969248e-01 0.704362133 0.901651455  
## maritalsingle 9.700036e-01 0.849086024 1.108141033  
## educationsecondary 1.220074e+00 1.064763209 1.398038439  
## educationtertiary 1.428312e+00 1.220043714 1.672133723  
## educationunknown 1.268674e+00 1.019957936 1.578038457  
## balance 1.000009e+00 0.999997993 1.000020277  
## housingyes 6.157788e-01 0.561318144 0.675523402  
## loanyes 7.641482e-01 0.673807469 0.866601394  
## contacttelephone 7.995305e-01 0.682696168 0.936359360  
## contactunknown 9.027425e-01 0.760520273 1.071561129  
## day 1.021705e+00 1.016412930 1.027024374  
## monthfeb 3.844106e+00 2.941768178 5.023220227  
## monthmar 1.635863e+01 11.982171927 22.333576359  
## monthapr 4.259565e+00 3.329356839 5.449669976  
## monthmay 3.283657e+00 2.572122473 4.192026308  
## monthjun 9.421325e+00 7.182968245 12.357199735  
## monthjul 7.846721e+00 6.028349745 10.213580823  
## monthaug 8.390774e+00 6.483794036 10.858626287  
## monthsep 9.033785e+00 6.654751231 12.263307783  
## monthoct 1.156821e+01 8.678543321 15.420042263  
## monthnov 9.483305e+00 7.182235538 12.521597976  
## monthdec 1.349713e+01 8.955337551 20.342329146  
## duration 1.004530e+00 1.004389559 1.004670243  
## campaign 9.532795e-01 0.934058093 0.972896459  
## pdays 9.992776e-01 0.998677334 0.999878136  
## poutcomeother 1.226642e+00 1.015288106 1.481993367  
## poutcomesuccess 6.687445e+00 5.627602242 7.946887479  
## poutcomeunknown 1.429953e+00 1.188360395 1.720661204  
## year2009 8.938763e+00 7.713302627 10.358918996  
## year2010 2.654431e+01 22.429346569 31.414217089

#Simple model : All predictors with no day and days  
Model\_SIM1<-Model\_Full<-glm(as.factor(y) ~ age+job+marital+education+balance+housing+loan+contact+month+duration+campaign+previous+poutcome+year,data=train,family= binomial(link="logit"))  
summary(Model\_SIM1)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + marital + education +   
## balance + housing + loan + contact + month + duration + campaign +   
## previous + poutcome + year, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6543 -0.3412 -0.2114 -0.1315 3.2078   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.386e+00 2.167e-01 -29.474 < 2e-16 \*\*\*  
## age -1.252e-03 2.353e-03 -0.532 0.594662   
## jobblue-collar -2.115e-01 7.776e-02 -2.720 0.006523 \*\*   
## jobentrepreneur -2.177e-01 1.341e-01 -1.624 0.104434   
## jobhousemaid -5.240e-01 1.490e-01 -3.517 0.000437 \*\*\*  
## jobmanagement -4.687e-02 7.854e-02 -0.597 0.550618   
## jobretired 4.810e-02 1.062e-01 0.453 0.650696   
## jobself-employed -1.987e-01 1.213e-01 -1.638 0.101501   
## jobservices -1.756e-01 9.071e-02 -1.936 0.052920 .   
## jobstudent 5.796e-02 1.152e-01 0.503 0.614874   
## jobtechnician -4.047e-02 7.379e-02 -0.548 0.583400   
## jobunemployed -2.280e-01 1.211e-01 -1.883 0.059706 .   
## jobunknown -2.986e-01 2.497e-01 -1.196 0.231881   
## maritalmarried -2.243e-01 6.313e-02 -3.553 0.000381 \*\*\*  
## maritalsingle -3.354e-02 7.237e-02 -0.463 0.643057   
## educationsecondary 1.882e-01 6.956e-02 2.705 0.006823 \*\*   
## educationtertiary 3.532e-01 8.075e-02 4.374 1.22e-05 \*\*\*  
## educationunknown 2.377e-01 1.112e-01 2.138 0.032504 \*   
## balance 9.864e-06 5.631e-06 1.752 0.079826 .   
## housingyes -5.190e-01 4.724e-02 -10.988 < 2e-16 \*\*\*  
## loanyes -2.784e-01 6.415e-02 -4.339 1.43e-05 \*\*\*  
## contacttelephone -2.284e-01 8.162e-02 -2.798 0.005135 \*\*   
## contactunknown -1.143e-01 8.710e-02 -1.313 0.189291   
## monthfeb 9.440e-01 1.273e-01 7.418 1.19e-13 \*\*\*  
## monthmar 2.531e+00 1.554e-01 16.290 < 2e-16 \*\*\*  
## monthapr 1.280e+00 1.245e-01 10.279 < 2e-16 \*\*\*  
## monthmay 9.238e-01 1.210e-01 7.632 2.30e-14 \*\*\*  
## monthjun 1.872e+00 1.307e-01 14.328 < 2e-16 \*\*\*  
## monthjul 1.795e+00 1.308e-01 13.722 < 2e-16 \*\*\*  
## monthaug 1.822e+00 1.267e-01 14.383 < 2e-16 \*\*\*  
## monthsep 1.882e+00 1.512e-01 12.447 < 2e-16 \*\*\*  
## monthoct 2.256e+00 1.450e-01 15.554 < 2e-16 \*\*\*  
## monthnov 2.002e+00 1.383e-01 14.478 < 2e-16 \*\*\*  
## monthdec 2.347e+00 2.070e-01 11.338 < 2e-16 \*\*\*  
## duration 4.493e-03 7.111e-05 63.178 < 2e-16 \*\*\*  
## campaign -3.890e-02 1.027e-02 -3.788 0.000152 \*\*\*  
## previous -8.607e-03 1.144e-02 -0.752 0.451951   
## poutcomeother 2.387e-01 9.611e-02 2.484 0.012987 \*   
## poutcomesuccess 1.962e+00 8.522e-02 23.022 < 2e-16 \*\*\*  
## poutcomeunknown 5.028e-01 7.101e-02 7.081 1.43e-12 \*\*\*  
## year2009 2.081e+00 7.376e-02 28.208 < 2e-16 \*\*\*  
## year2010 3.189e+00 8.534e-02 37.369 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18550 on 41671 degrees of freedom  
## AIC: 18634  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18550 on 41671 degrees of freedom  
#AIC: 18634  
#Call:  
#glm(formula = as.factor(y) ~ age + job + marital + education +   
# balance + housing + loan + contact + month + duration + campaign +   
# previous + poutcome + year, family = binomial(link = "logit"),   
# data = train)  
(vif(Model\_Full)[,3])^2

## age job marital education balance housing loan contact   
## 2.264175 1.144241 1.210284 1.311955 1.041059 1.443677 1.061613 1.563201   
## month duration campaign previous poutcome year   
## 1.147297 1.197065 1.099959 1.722443 1.263337 1.984311

hoslem.test(Model\_SIM1$y, fitted(Model\_SIM1), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM1$y, fitted(Model\_SIM1)  
## X-squared = 520.22, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#Simple model2 by hand seleting the predictors  
Model\_SIM2<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+previous+year,data=train,family= binomial(link="logit"))  
summary(Model\_SIM2)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + previous + year, family = binomial(link = "logit"),   
## data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.4210 -0.4059 -0.2242 -0.1567 3.1050   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.779e+00 1.300e-01 -36.779 < 2e-16 \*\*\*  
## age 3.239e-03 1.991e-03 1.627 0.103757   
## jobblue-collar -4.390e-01 7.333e-02 -5.986 2.15e-09 \*\*\*  
## jobentrepreneur -3.497e-01 1.276e-01 -2.741 0.006130 \*\*   
## jobhousemaid -2.975e-01 1.414e-01 -2.104 0.035391 \*   
## jobmanagement -2.344e-02 7.433e-02 -0.315 0.752470   
## jobretired 2.323e-01 9.972e-02 2.329 0.019838 \*   
## jobself-employed -1.560e-01 1.147e-01 -1.361 0.173610   
## jobservices -2.835e-01 8.579e-02 -3.305 0.000949 \*\*\*  
## jobstudent 5.049e-01 1.072e-01 4.712 2.46e-06 \*\*\*  
## jobtechnician -2.545e-02 6.975e-02 -0.365 0.715193   
## jobunemployed -1.209e-01 1.125e-01 -1.075 0.282466   
## jobunknown -8.048e-02 2.425e-01 -0.332 0.740013   
## educationsecondary 2.122e-01 6.621e-02 3.205 0.001350 \*\*   
## educationtertiary 5.144e-01 7.658e-02 6.717 1.85e-11 \*\*\*  
## educationunknown 2.943e-01 1.052e-01 2.798 0.005138 \*\*   
## balance 2.364e-05 5.211e-06 4.536 5.72e-06 \*\*\*  
## loanyes -4.061e-01 6.117e-02 -6.640 3.13e-11 \*\*\*  
## duration 4.254e-03 6.731e-05 63.210 < 2e-16 \*\*\*  
## campaign -4.430e-02 9.830e-03 -4.507 6.58e-06 \*\*\*  
## previous -3.040e-03 8.206e-03 -0.370 0.711027   
## year2009 1.654e+00 4.551e-02 36.338 < 2e-16 \*\*\*  
## year2010 3.301e+00 6.313e-02 52.283 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 20412 on 41690 degrees of freedom  
## AIC: 20458  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 20412 on 41690 degrees of freedom  
#AIC: 20458  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + previous + year, family = binomial(link = "logit"),   
# data = train)  
hoslem.test(Model\_SIM1$y, fitted(Model\_SIM2), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM1$y, fitted(Model\_SIM2)  
## X-squared = 458.69, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#With interactions:  
#Simple model2 by hand seleting the predictors  
Model\_SIM3<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+previous+year+year\*month+year\*day,data=train,family= binomial(link="logit"))  
summary(Model\_SIM3)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + previous + year + year \* month +   
## year \* day, family = binomial(link = "logit"), data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.4822 -0.3161 -0.1952 -0.1332 3.0623   
##   
## Coefficients: (6 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.828e+01 1.005e+02 -0.182 0.85572   
## age -4.191e-03 2.090e-03 -2.006 0.04491 \*   
## jobblue-collar -1.900e-01 7.756e-02 -2.450 0.01428 \*   
## jobentrepreneur -1.507e-01 1.348e-01 -1.118 0.26360   
## jobhousemaid -4.767e-01 1.482e-01 -3.217 0.00129 \*\*   
## jobmanagement -1.715e-02 7.809e-02 -0.220 0.82620   
## jobretired 1.559e-02 1.056e-01 0.148 0.88265   
## jobself-employed -1.125e-01 1.214e-01 -0.927 0.35384   
## jobservices -8.587e-02 9.051e-02 -0.949 0.34274   
## jobstudent 1.330e-01 1.126e-01 1.181 0.23767   
## jobtechnician -2.503e-02 7.379e-02 -0.339 0.73440   
## jobunemployed -4.278e-02 1.194e-01 -0.358 0.72020   
## jobunknown -1.804e-01 2.501e-01 -0.721 0.47067   
## educationsecondary 2.066e-01 6.966e-02 2.966 0.00302 \*\*   
## educationtertiary 3.500e-01 8.077e-02 4.334 1.46e-05 \*\*\*  
## educationunknown 2.186e-01 1.110e-01 1.970 0.04886 \*   
## balance 4.447e-06 5.883e-06 0.756 0.44970   
## loanyes -2.968e-01 6.443e-02 -4.606 4.11e-06 \*\*\*  
## duration 4.692e-03 7.311e-05 64.185 < 2e-16 \*\*\*  
## campaign -4.479e-02 1.058e-02 -4.233 2.30e-05 \*\*\*  
## previous -4.271e-03 8.877e-03 -0.481 0.63046   
## year2009 1.105e+01 1.005e+02 0.110 0.91249   
## year2010 1.707e+01 1.005e+02 0.170 0.86518   
## monthfeb 2.063e-02 1.935e-01 0.107 0.91512   
## monthmar 3.254e-01 2.101e-01 1.549 0.12150   
## monthapr 3.066e-01 2.147e-01 1.428 0.15321   
## monthmay 1.290e+01 1.005e+02 0.128 0.89792   
## monthjun 1.347e+01 1.005e+02 0.134 0.89338   
## monthjul 1.370e+01 1.005e+02 0.136 0.89161   
## monthaug 1.387e+01 1.005e+02 0.138 0.89024   
## monthsep 1.270e-02 2.014e-01 0.063 0.94969   
## monthoct 1.778e+01 1.005e+02 0.177 0.85956   
## monthnov 1.368e+01 1.005e+02 0.136 0.89173   
## monthdec 4.980e+00 2.792e-01 17.834 < 2e-16 \*\*\*  
## day 2.490e-03 4.903e-03 0.508 0.61147   
## year2009:monthfeb 3.461e+00 3.131e-01 11.055 < 2e-16 \*\*\*  
## year2010:monthfeb NA NA NA NA   
## year2009:monthmar 5.121e+00 3.394e-01 15.086 < 2e-16 \*\*\*  
## year2010:monthmar NA NA NA NA   
## year2009:monthapr 2.733e+00 3.118e-01 8.765 < 2e-16 \*\*\*  
## year2010:monthapr NA NA NA NA   
## year2009:monthmay -1.014e+01 1.005e+02 -0.101 0.91968   
## year2010:monthmay -1.267e+01 1.005e+02 -0.126 0.89970   
## year2009:monthjun -8.233e+00 1.005e+02 -0.082 0.93472   
## year2010:monthjun -1.342e+01 1.005e+02 -0.134 0.89379   
## year2009:monthjul -8.816e+00 1.005e+02 -0.088 0.93011   
## year2010:monthjul -1.352e+01 1.005e+02 -0.135 0.89298   
## year2009:monthaug -9.293e+00 1.005e+02 -0.092 0.92634   
## year2010:monthaug -1.364e+01 1.005e+02 -0.136 0.89204   
## year2009:monthsep 5.102e+00 3.344e-01 15.257 < 2e-16 \*\*\*  
## year2010:monthsep NA NA NA NA   
## year2009:monthoct -1.306e+01 1.005e+02 -0.130 0.89662   
## year2010:monthoct -1.821e+01 1.005e+02 -0.181 0.85620   
## year2009:monthnov -8.252e+00 1.005e+02 -0.082 0.93457   
## year2010:monthnov -1.393e+01 1.005e+02 -0.139 0.88977   
## year2009:monthdec NA NA NA NA   
## year2010:monthdec NA NA NA NA   
## year2009:day 5.447e-02 6.297e-03 8.650 < 2e-16 \*\*\*  
## year2010:day -2.904e-03 7.275e-03 -0.399 0.68969   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 30188 on 41712 degrees of freedom  
## Residual deviance: 18378 on 41660 degrees of freedom  
## AIC: 18484  
##   
## Number of Fisher Scoring iterations: 11

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 20386 on 41686 degrees of freedom  
#AIC: 20440  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + previous + year + year \* month +   
# year \* day, family = binomial(link = "logit"), data = train)  
hoslem.test(Model\_SIM1$y, fitted(Model\_SIM2), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM1$y, fitted(Model\_SIM2)  
## X-squared = 458.69, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#With interactions:  
#Simple model2 by hand seleting the predictors  
Model\_SIM4<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+pdays+previous+year+I(duration^2),I(pdays^2),data=train,family= binomial(link="logit"))

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

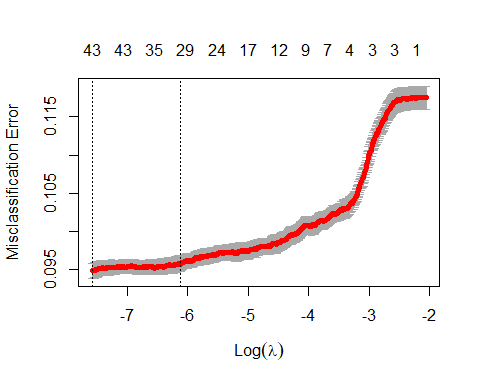
summary(Model\_SIM4)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + pdays + previous + year + I(duration^2),   
## family = binomial(link = "logit"), data = train, weights = I(pdays^2))  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5722 0 0 0 7251   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.113e+15 2.641e+04 -7.999e+10 <2e-16 \*\*\*  
## age 4.747e+12 3.543e+02 1.340e+10 <2e-16 \*\*\*  
## jobblue-collar -2.229e+14 1.071e+04 -2.082e+10 <2e-16 \*\*\*  
## jobentrepreneur -2.392e+14 2.122e+04 -1.128e+10 <2e-16 \*\*\*  
## jobhousemaid 6.354e+14 2.795e+04 2.273e+10 <2e-16 \*\*\*  
## jobmanagement 3.262e+14 1.295e+04 2.520e+10 <2e-16 \*\*\*  
## jobretired 1.989e+14 1.965e+04 1.012e+10 <2e-16 \*\*\*  
## jobself-employed 5.239e+12 2.128e+04 2.462e+08 <2e-16 \*\*\*  
## jobservices 2.786e+14 1.275e+04 2.185e+10 <2e-16 \*\*\*  
## jobstudent 4.625e+14 2.160e+04 2.141e+10 <2e-16 \*\*\*  
## jobtechnician 7.024e+13 1.137e+04 6.176e+09 <2e-16 \*\*\*  
## jobunemployed 9.002e+14 2.128e+04 4.230e+10 <2e-16 \*\*\*  
## jobunknown 7.002e+14 5.342e+04 1.311e+10 <2e-16 \*\*\*  
## educationsecondary 2.077e+14 9.974e+03 2.082e+10 <2e-16 \*\*\*  
## educationtertiary 1.382e+14 1.304e+04 1.059e+10 <2e-16 \*\*\*  
## educationunknown 1.783e+14 1.836e+04 9.714e+09 <2e-16 \*\*\*  
## balance 4.988e+09 1.091e+00 4.570e+09 <2e-16 \*\*\*  
## loanyes -4.463e+14 8.947e+03 -4.989e+10 <2e-16 \*\*\*  
## duration 5.385e+12 3.105e+01 1.735e+11 <2e-16 \*\*\*  
## campaign -9.062e+13 1.947e+03 -4.655e+10 <2e-16 \*\*\*  
## pdays -1.171e+12 2.627e+01 -4.456e+10 <2e-16 \*\*\*  
## previous 9.114e+12 7.876e+02 1.157e+10 <2e-16 \*\*\*  
## year2009 7.499e+14 1.592e+04 4.709e+10 <2e-16 \*\*\*  
## year2010 -2.440e+14 1.828e+04 -1.335e+10 <2e-16 \*\*\*  
## I(duration^2) -2.677e+09 2.582e-02 -1.037e+11 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 469874077 on 41712 degrees of freedom  
## Residual deviance: 8649774827 on 41688 degrees of freedom  
## AIC: 8649774877  
##   
## Number of Fisher Scoring iterations: 25

#Null deviance: 469874077 on 41712 degrees of freedom  
#Residual deviance: 8649774827 on 41688 degrees of freedom  
#AIC: 8649774877  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + pdays + previous + year + I(duration^2),   
# family = binomial(link = "logit"), data = train, weights = I(pdays^2))  
hoslem.test(Model\_SIM1$y, fitted(Model\_SIM2), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM1$y, fitted(Model\_SIM2)  
## X-squared = 458.69, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#LASSO  
#LASSO is obtained in the exact same way for logistic as is MLR. The only difference is to let R know that our response is #categorical through the "family="binomial"" option. Cross validation is used to obtain the optimal penalty value. A final #refit using the entire data set can then be obtained once the optimal penalty value is determined. For this example, the object #"finalmodel" produces the final lasso model  
dat.train.x <- model.matrix(y~age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year-1,train)  
dat.train.y<-train$y  
cvfit <- cv.glmnet(dat.train.x, dat.train.y, family = "binomial", type.measure = "class", nlambda = 1000)  
plot(cvfit)



coef(cvfit, s = "lambda.min")

## 46 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) -6.032666e+00  
## age -6.455071e-05  
## jobadmin. 1.288368e-01  
## jobblue-collar -5.623092e-02  
## jobentrepreneur -2.697938e-02  
## jobhousemaid -3.528048e-01  
## jobmanagement 9.162769e-02  
## jobretired 1.223585e-01  
## jobself-employed -1.129643e-02  
## jobservices .   
## jobstudent 2.109067e-01  
## jobtechnician 8.388718e-02  
## jobunemployed -6.045932e-02  
## jobunknown -6.007432e-02  
## maritalmarried -1.954264e-01  
## maritalsingle .   
## educationsecondary 1.055458e-01  
## educationtertiary 2.664175e-01  
## educationunknown 1.157690e-01  
## defaultyes 1.206023e-01  
## balance 9.261099e-06  
## housingyes -4.624033e-01  
## loanyes -2.528138e-01  
## contacttelephone -1.891315e-01  
## contactunknown -8.458229e-02  
## day 1.429225e-02  
## monthfeb 5.977596e-01  
## monthmar 2.089640e+00  
## monthapr 7.794837e-01  
## monthmay 4.554895e-01  
## monthjun 1.442660e+00  
## monthjul 1.286791e+00  
## monthaug 1.368800e+00  
## monthsep 1.482135e+00  
## monthoct 1.763355e+00  
## monthnov 1.466868e+00  
## monthdec 1.876458e+00  
## duration 4.425056e-03  
## campaign -3.859519e-02  
## pdays -7.240670e-04  
## previous -5.674110e-03  
## poutcomeother 1.270893e-01  
## poutcomesuccess 1.835129e+00  
## poutcomeunknown 2.488124e-01  
## year2009 2.044602e+00  
## year2010 3.120211e+00

#CV misclassification error rate is little below .1  
print("CV Error Rate:")

## [1] "CV Error Rate:"

cvfit$cvm[which(cvfit$lambda==cvfit$lambda.min)]

## [1] 0.09483854

#Optimal penalty  
print("Penalty Value:")

## [1] "Penalty Value:"

cvfit$lambda.min

## [1] 0.0005113941

#For final model predictions go ahead and refit lasso using entire  
#data set  
finalmodel<-glmnet(dat.train.x, dat.train.y, family = "binomial",lambda=cvfit$lambda.min)  
# print("CV Error Rate:")  
#[1] "CV Error Rate:"  
#> cvfit$cvm[which(cvfit$lambda==cvfit$lambda.min)]  
#[1] 0.09479059  
#>   
#> #Optimal penalty  
#> print("Penalty Value:")  
#[1] "Penalty Value:"  
#> cvfit$lambda.min  
#[1] 0.0005113941

set.seed(1234)  
  
#With all Years  
Model\_Full\_2008<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year,data=train2008,family= binomial(link="logit"))  
(vif(Model\_Full\_2008)[,3])^2

## age job marital education default balance housing loan   
## 2.702001 1.157426 1.249892 1.300687 1.010320 1.041250 1.518364 1.044441   
## contact day month duration campaign pdays previous poutcome   
## 1.126039 1.377566 1.080279 1.113291 1.047183 3.005687 1.573040 1.564751   
## year   
## 1.376484

#On letting all the predictors in the model and running the vif function, nothing stands out in terms of excluding the predictors from the modle with a VIF score > 10  
  
#Summary of current fit  
summary(Model\_Full\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.0033 -0.5375 -0.3273 -0.1791 3.0858   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.917e+00 2.689e-01 -18.287 < 2e-16 \*\*\*  
## age 3.715e-03 3.001e-03 1.238 0.21576   
## jobblue-collar -2.899e-01 1.036e-01 -2.798 0.00514 \*\*   
## jobentrepreneur -4.291e-01 1.962e-01 -2.187 0.02874 \*   
## jobhousemaid -3.580e-01 2.003e-01 -1.787 0.07391 .   
## jobmanagement -1.026e-03 9.798e-02 -0.010 0.99165   
## jobretired 2.264e-02 1.333e-01 0.170 0.86511   
## jobself-employed -1.791e-01 1.599e-01 -1.120 0.26263   
## jobservices -1.231e-01 1.189e-01 -1.035 0.30058   
## jobstudent 1.567e-01 1.327e-01 1.181 0.23768   
## jobtechnician -1.579e-02 9.427e-02 -0.167 0.86699   
## jobunemployed -2.343e-01 1.508e-01 -1.554 0.12025   
## jobunknown -6.376e-02 3.168e-01 -0.201 0.84047   
## maritalmarried -9.675e-02 8.690e-02 -1.113 0.26559   
## maritalsingle 6.028e-02 9.846e-02 0.612 0.54041   
## educationsecondary 2.818e-01 9.383e-02 3.004 0.00267 \*\*   
## educationtertiary 4.557e-01 1.060e-01 4.300 1.71e-05 \*\*\*  
## educationunknown 2.832e-01 1.433e-01 1.976 0.04819 \*   
## defaultyes -2.834e-02 3.327e-01 -0.085 0.93213   
## balance 2.076e-06 7.382e-06 0.281 0.77850   
## housingyes -6.775e-01 6.332e-02 -10.700 < 2e-16 \*\*\*  
## loanyes -2.774e-01 9.515e-02 -2.915 0.00355 \*\*   
## contacttelephone -4.937e-01 1.011e-01 -4.881 1.06e-06 \*\*\*  
## contactunknown -1.411e+00 2.297e-01 -6.142 8.17e-10 \*\*\*  
## day 2.585e-02 3.384e-03 7.639 2.19e-14 \*\*\*  
## monthfeb 1.386e+00 1.477e-01 9.382 < 2e-16 \*\*\*  
## monthmar 2.734e+00 1.692e-01 16.155 < 2e-16 \*\*\*  
## monthapr 1.512e+00 1.334e-01 11.329 < 2e-16 \*\*\*  
## monthmay 1.207e+00 1.361e-01 8.865 < 2e-16 \*\*\*  
## monthjun 2.394e+00 1.580e-01 15.158 < 2e-16 \*\*\*  
## monthjul 2.123e+00 1.738e-01 12.211 < 2e-16 \*\*\*  
## monthaug 2.116e+00 1.487e-01 14.230 < 2e-16 \*\*\*  
## monthsep 2.133e+00 1.684e-01 12.664 < 2e-16 \*\*\*  
## monthoct 2.135e+00 1.583e-01 13.488 < 2e-16 \*\*\*  
## monthnov 2.665e+00 1.863e-01 14.302 < 2e-16 \*\*\*  
## monthdec 2.517e+00 2.243e-01 11.223 < 2e-16 \*\*\*  
## duration 3.733e-03 1.084e-04 34.451 < 2e-16 \*\*\*  
## campaign -8.143e-02 1.835e-02 -4.438 9.10e-06 \*\*\*  
## pdays 4.407e-05 3.322e-04 0.133 0.89444   
## previous 1.168e-02 1.165e-02 1.002 0.31618   
## poutcomeother 1.046e-01 1.055e-01 0.991 0.32183   
## poutcomesuccess 1.867e+00 9.500e-02 19.649 < 2e-16 \*\*\*  
## poutcomeunknown 5.075e-01 1.103e-01 4.603 4.17e-06 \*\*\*  
## year2010 1.085e+00 6.986e-02 15.538 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10246 on 13941 degrees of freedom  
## AIC: 10334  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 10246 on 13941 degrees of freedom  
#AIC: 10334  
#Call:  
#glm(formula = as.factor(y) ~ age + job + marital + education +   
# default + balance + housing + loan + contact + day + month +   
# duration + campaign + pdays + previous + poutcome + year,   
# family = binomial(link = "logit"), data = train2008)  
  
#Hosmer Lemeshow test for lack of fit. Use as needed. The g=10 is an option that deals with the continuous predictors if any are there.  
#This should be increased with caution.   
hoslem.test(Model\_Full\_2008$y, fitted(Model\_Full\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Full\_2008$y, fitted(Model\_Full\_2008)  
## X-squared = 246.72, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
# Using the summary coefficients we can generate CI for each one in the table shwing odds ratio  
exp(cbind("Odds ratio" = coef(Model\_Full\_2008), confint.default(Model\_Full\_2008, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.007318408 0.004320477 0.01239657  
## age 1.003722210 0.997835117 1.00964404  
## jobblue-collar 0.748307434 0.610784154 0.91679526  
## jobentrepreneur 0.651062844 0.443197962 0.95641872  
## jobhousemaid 0.699053208 0.472049485 1.03522068  
## jobmanagement 0.998974983 0.824426434 1.21047916  
## jobretired 1.022902918 0.787704344 1.32832882  
## jobself-employed 0.836028783 0.611131881 1.14368788  
## jobservices 0.884145757 0.700286382 1.11627720  
## jobstudent 1.169665414 0.901758647 1.51716558  
## jobtechnician 0.984334328 0.818271726 1.18409819  
## jobunemployed 0.791142527 0.588717093 1.06317025  
## jobunknown 0.938229682 0.504298753 1.74554256  
## maritalmarried 0.907786102 0.765618286 1.07635309  
## maritalsingle 1.062130872 0.875725671 1.28821391  
## educationsecondary 1.325540287 1.102872966 1.59316359  
## educationtertiary 1.577307600 1.281479512 1.94142727  
## educationunknown 1.327333031 1.002255241 1.75784860  
## defaultyes 0.972060876 0.506389411 1.86595993  
## balance 1.000002076 0.999987609 1.00001654  
## housingyes 0.507863461 0.448589192 0.57496993  
## loanyes 0.757757192 0.628831218 0.91311618  
## contacttelephone 0.610379808 0.500619755 0.74420457  
## contactunknown 0.244020789 0.155573432 0.38275266  
## day 1.026188834 1.019404914 1.03301790  
## monthfeb 3.999150103 2.993694590 5.34229564  
## monthmar 15.390787646 11.046255699 21.44403957  
## monthapr 4.534755669 3.491143770 5.89033576  
## monthmay 3.342184369 2.559619029 4.36400739  
## monthjun 10.962257872 8.043393340 14.94034825  
## monthjul 8.354257341 5.942000336 11.74581147  
## monthaug 8.296550400 6.199212740 11.10346611  
## monthsep 8.440319595 6.067186606 11.74168515  
## monthoct 8.460136367 6.203287699 11.53806027  
## monthnov 14.365181873 9.970438835 20.69702785  
## monthdec 12.397151521 7.986921935 19.24262777  
## duration 1.003740386 1.003527216 1.00395360  
## campaign 0.921797330 0.889233954 0.95555316  
## pdays 1.000044075 0.999393208 1.00069537  
## previous 1.011747576 0.988903721 1.03511913  
## poutcomeother 1.110224511 0.902758096 1.36536960  
## poutcomesuccess 6.466458132 5.367875727 7.78987497  
## poutcomeunknown 1.661211808 1.338330800 2.06198996  
## year2010 2.960884042 2.582006848 3.39535672

#stepwise:  
#This starts with a null model and then builds up using forward selection up to all the predictors that were specified in main model previously.  
# Code from AutoClassify.R  
Model\_Step\_2008<-Model\_Full\_2008 %>%stepAIC(trace=FALSE)  
summary(Model\_Step\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + housing +   
## loan + contact + day + month + duration + campaign + poutcome +   
## year, family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.9950 -0.5381 -0.3273 -0.1795 3.0916   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.7008278 0.2138128 -21.986 < 2e-16 \*\*\*  
## jobblue-collar -0.2939209 0.1035453 -2.839 0.00453 \*\*   
## jobentrepreneur -0.4180368 0.1957072 -2.136 0.03268 \*   
## jobhousemaid -0.3321005 0.1987704 -1.671 0.09477 .   
## jobmanagement 0.0071637 0.0977659 0.073 0.94159   
## jobretired 0.1043816 0.1152471 0.906 0.36508   
## jobself-employed -0.1736622 0.1597662 -1.087 0.27705   
## jobservices -0.1278065 0.1188843 -1.075 0.28235   
## jobstudent 0.1244246 0.1302670 0.955 0.33950   
## jobtechnician -0.0138727 0.0942452 -0.147 0.88298   
## jobunemployed -0.2283329 0.1506277 -1.516 0.12955   
## jobunknown -0.0342016 0.3157230 -0.108 0.91374   
## maritalmarried -0.1059043 0.0864863 -1.225 0.22076   
## maritalsingle 0.0164175 0.0920867 0.178 0.85850   
## educationsecondary 0.2687447 0.0933249 2.880 0.00398 \*\*   
## educationtertiary 0.4386674 0.1051525 4.172 3.02e-05 \*\*\*  
## educationunknown 0.2808148 0.1432851 1.960 0.05002 .   
## housingyes -0.6802740 0.0625464 -10.876 < 2e-16 \*\*\*  
## loanyes -0.2778936 0.0949990 -2.925 0.00344 \*\*   
## contacttelephone -0.4701659 0.0995239 -4.724 2.31e-06 \*\*\*  
## contactunknown -1.4258983 0.2287338 -6.234 4.55e-10 \*\*\*  
## day 0.0258086 0.0033827 7.630 2.35e-14 \*\*\*  
## monthfeb 1.3884476 0.1475900 9.407 < 2e-16 \*\*\*  
## monthmar 2.7374611 0.1690176 16.196 < 2e-16 \*\*\*  
## monthapr 1.5136411 0.1332305 11.361 < 2e-16 \*\*\*  
## monthmay 1.2084326 0.1358847 8.893 < 2e-16 \*\*\*  
## monthjun 2.3980977 0.1575701 15.219 < 2e-16 \*\*\*  
## monthjul 2.1281387 0.1735089 12.265 < 2e-16 \*\*\*  
## monthaug 2.1267500 0.1483408 14.337 < 2e-16 \*\*\*  
## monthsep 2.1389486 0.1681809 12.718 < 2e-16 \*\*\*  
## monthoct 2.1485689 0.1579253 13.605 < 2e-16 \*\*\*  
## monthnov 2.6783417 0.1858305 14.413 < 2e-16 \*\*\*  
## monthdec 2.5273175 0.2242108 11.272 < 2e-16 \*\*\*  
## duration 0.0037359 0.0001083 34.500 < 2e-16 \*\*\*  
## campaign -0.0802116 0.0182904 -4.385 1.16e-05 \*\*\*  
## poutcomeother 0.1121676 0.1049346 1.069 0.28510   
## poutcomesuccess 1.8629641 0.0921676 20.213 < 2e-16 \*\*\*  
## poutcomeunknown 0.4629562 0.0691100 6.699 2.10e-11 \*\*\*  
## year2010 1.0976999 0.0691851 15.866 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10249 on 13946 degrees of freedom  
## AIC: 10327  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14784 on 13984 degrees of freedom  
#Residual deviance: 10221 on 13946 degrees of freedom  
#AIC: 10299  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + housing +   
# loan + contact + day + month + duration + campaign + poutcome +   
# year, family = binomial(link = "logit"), data = train2008)  
(vif(Model\_Step\_2008)[,3])^2

## job marital education housing loan contact day month   
## 1.110102 1.109564 1.291016 1.481903 1.040975 1.104116 1.376434 1.077312   
## duration campaign poutcome year   
## 1.112679 1.041817 1.069001 1.350383

hoslem.test(Model\_Step\_2008$y, fitted(Model\_Step\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Step\_2008$y, fitted(Model\_Step\_2008)  
## X-squared = 250.36, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Step\_2008), confint.default(Model\_Step\_2008, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.009087751 0.005976661 0.01381829  
## jobblue-collar 0.745335421 0.608434518 0.91303973  
## jobentrepreneur 0.658338023 0.448603495 0.96612924  
## jobhousemaid 0.717415232 0.485933603 1.05916654  
## jobmanagement 1.007189454 0.831558197 1.21991534  
## jobretired 1.110023921 0.885592254 1.39133230  
## jobself-employed 0.840580761 0.614590995 1.14966867  
## jobservices 0.880023654 0.697107614 1.11093555  
## jobstudent 1.132496618 0.877310744 1.46190913  
## jobtechnician 0.986223119 0.819886121 1.18630626  
## jobunemployed 0.795859295 0.592409188 1.06917994  
## jobunknown 0.966376638 0.520478155 1.79428050  
## maritalmarried 0.899510723 0.759257410 1.06567223  
## maritalsingle 1.016552996 0.848683420 1.21762717  
## educationsecondary 1.308321048 1.089622395 1.57091482  
## educationtertiary 1.550639519 1.261841763 1.90553442  
## educationunknown 1.324208334 0.999981196 1.75356069  
## housingyes 0.506478192 0.448044527 0.57253274  
## loanyes 0.757377415 0.628708046 0.91237984  
## contacttelephone 0.624898600 0.514155657 0.75949424  
## contactunknown 0.240292500 0.153476421 0.37621731  
## day 1.026144476 1.019363716 1.03297034  
## monthfeb 4.008622396 3.001693226 5.35332971  
## monthmar 15.447715734 11.091652250 21.51455130  
## monthapr 4.543242975 3.499128770 5.89891315  
## monthmay 3.348232608 2.565371817 4.36999484  
## monthjun 11.002226732 8.078983039 14.98319683  
## monthjul 8.399218618 5.977888879 11.80130224  
## monthaug 8.387563056 6.271448210 11.21769832  
## monthsep 8.490506065 6.106294178 11.80563712  
## monthoct 8.572581615 6.290502416 11.68255740  
## monthnov 14.560927212 10.116023704 20.95888735  
## monthdec 12.519876921 8.067731587 19.42892081  
## duration 1.003742932 1.003529922 1.00395599  
## campaign 0.922920999 0.890421627 0.95660656  
## poutcomeother 1.118700298 0.910737531 1.37415042  
## poutcomesuccess 6.442805912 5.378013338 7.71841671  
## poutcomeunknown 1.588763806 1.387499496 1.81922259  
## year2010 2.997263991 2.617185477 3.43253908

Model\_Null\_2008<-glm(as.factor(y) ~ 1, data=train2008,family = binomial(link="logit"))  
#Forward:  
Model\_FWD\_2008<-stepAIC(Model\_Full\_2008,  
 scope = list(upper=Model\_Full\_2008),  
 direction="forward",  
 test="Chisq",  
 data=train2008)

## Start: AIC=10333.99  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year

summary(Model\_FWD\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.0033 -0.5375 -0.3273 -0.1791 3.0858   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.917e+00 2.689e-01 -18.287 < 2e-16 \*\*\*  
## age 3.715e-03 3.001e-03 1.238 0.21576   
## jobblue-collar -2.899e-01 1.036e-01 -2.798 0.00514 \*\*   
## jobentrepreneur -4.291e-01 1.962e-01 -2.187 0.02874 \*   
## jobhousemaid -3.580e-01 2.003e-01 -1.787 0.07391 .   
## jobmanagement -1.026e-03 9.798e-02 -0.010 0.99165   
## jobretired 2.264e-02 1.333e-01 0.170 0.86511   
## jobself-employed -1.791e-01 1.599e-01 -1.120 0.26263   
## jobservices -1.231e-01 1.189e-01 -1.035 0.30058   
## jobstudent 1.567e-01 1.327e-01 1.181 0.23768   
## jobtechnician -1.579e-02 9.427e-02 -0.167 0.86699   
## jobunemployed -2.343e-01 1.508e-01 -1.554 0.12025   
## jobunknown -6.376e-02 3.168e-01 -0.201 0.84047   
## maritalmarried -9.675e-02 8.690e-02 -1.113 0.26559   
## maritalsingle 6.028e-02 9.846e-02 0.612 0.54041   
## educationsecondary 2.818e-01 9.383e-02 3.004 0.00267 \*\*   
## educationtertiary 4.557e-01 1.060e-01 4.300 1.71e-05 \*\*\*  
## educationunknown 2.832e-01 1.433e-01 1.976 0.04819 \*   
## defaultyes -2.834e-02 3.327e-01 -0.085 0.93213   
## balance 2.076e-06 7.382e-06 0.281 0.77850   
## housingyes -6.775e-01 6.332e-02 -10.700 < 2e-16 \*\*\*  
## loanyes -2.774e-01 9.515e-02 -2.915 0.00355 \*\*   
## contacttelephone -4.937e-01 1.011e-01 -4.881 1.06e-06 \*\*\*  
## contactunknown -1.411e+00 2.297e-01 -6.142 8.17e-10 \*\*\*  
## day 2.585e-02 3.384e-03 7.639 2.19e-14 \*\*\*  
## monthfeb 1.386e+00 1.477e-01 9.382 < 2e-16 \*\*\*  
## monthmar 2.734e+00 1.692e-01 16.155 < 2e-16 \*\*\*  
## monthapr 1.512e+00 1.334e-01 11.329 < 2e-16 \*\*\*  
## monthmay 1.207e+00 1.361e-01 8.865 < 2e-16 \*\*\*  
## monthjun 2.394e+00 1.580e-01 15.158 < 2e-16 \*\*\*  
## monthjul 2.123e+00 1.738e-01 12.211 < 2e-16 \*\*\*  
## monthaug 2.116e+00 1.487e-01 14.230 < 2e-16 \*\*\*  
## monthsep 2.133e+00 1.684e-01 12.664 < 2e-16 \*\*\*  
## monthoct 2.135e+00 1.583e-01 13.488 < 2e-16 \*\*\*  
## monthnov 2.665e+00 1.863e-01 14.302 < 2e-16 \*\*\*  
## monthdec 2.517e+00 2.243e-01 11.223 < 2e-16 \*\*\*  
## duration 3.733e-03 1.084e-04 34.451 < 2e-16 \*\*\*  
## campaign -8.143e-02 1.835e-02 -4.438 9.10e-06 \*\*\*  
## pdays 4.407e-05 3.322e-04 0.133 0.89444   
## previous 1.168e-02 1.165e-02 1.002 0.31618   
## poutcomeother 1.046e-01 1.055e-01 0.991 0.32183   
## poutcomesuccess 1.867e+00 9.500e-02 19.649 < 2e-16 \*\*\*  
## poutcomeunknown 5.075e-01 1.103e-01 4.603 4.17e-06 \*\*\*  
## year2010 1.085e+00 6.986e-02 15.538 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10246 on 13941 degrees of freedom  
## AIC: 10334  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 10246 on 13941 degrees of freedom  
#AIC: 10334  
#Call:  
#glm(formula = as.factor(y) ~ age + job + marital + education +   
# default + balance + housing + loan + contact + day + month +   
# duration + campaign + pdays + previous + poutcome + year,   
# family = binomial(link = "logit"), data = train2008)  
(vif(Model\_FWD\_2008)[,3])^2

## age job marital education default balance housing loan   
## 2.702001 1.157426 1.249892 1.300687 1.010320 1.041250 1.518364 1.044441   
## contact day month duration campaign pdays previous poutcome   
## 1.126039 1.377566 1.080279 1.113291 1.047183 3.005687 1.573040 1.564751   
## year   
## 1.376484

hoslem.test(Model\_FWD\_2008$y, fitted(Model\_FWD\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_FWD\_2008$y, fitted(Model\_FWD\_2008)  
## X-squared = 246.72, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_FWD\_2008), confint.default(Model\_FWD\_2008, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.007318408 0.004320477 0.01239657  
## age 1.003722210 0.997835117 1.00964404  
## jobblue-collar 0.748307434 0.610784154 0.91679526  
## jobentrepreneur 0.651062844 0.443197962 0.95641872  
## jobhousemaid 0.699053208 0.472049485 1.03522068  
## jobmanagement 0.998974983 0.824426434 1.21047916  
## jobretired 1.022902918 0.787704344 1.32832882  
## jobself-employed 0.836028783 0.611131881 1.14368788  
## jobservices 0.884145757 0.700286382 1.11627720  
## jobstudent 1.169665414 0.901758647 1.51716558  
## jobtechnician 0.984334328 0.818271726 1.18409819  
## jobunemployed 0.791142527 0.588717093 1.06317025  
## jobunknown 0.938229682 0.504298753 1.74554256  
## maritalmarried 0.907786102 0.765618286 1.07635309  
## maritalsingle 1.062130872 0.875725671 1.28821391  
## educationsecondary 1.325540287 1.102872966 1.59316359  
## educationtertiary 1.577307600 1.281479512 1.94142727  
## educationunknown 1.327333031 1.002255241 1.75784860  
## defaultyes 0.972060876 0.506389411 1.86595993  
## balance 1.000002076 0.999987609 1.00001654  
## housingyes 0.507863461 0.448589192 0.57496993  
## loanyes 0.757757192 0.628831218 0.91311618  
## contacttelephone 0.610379808 0.500619755 0.74420457  
## contactunknown 0.244020789 0.155573432 0.38275266  
## day 1.026188834 1.019404914 1.03301790  
## monthfeb 3.999150103 2.993694590 5.34229564  
## monthmar 15.390787646 11.046255699 21.44403957  
## monthapr 4.534755669 3.491143770 5.89033576  
## monthmay 3.342184369 2.559619029 4.36400739  
## monthjun 10.962257872 8.043393340 14.94034825  
## monthjul 8.354257341 5.942000336 11.74581147  
## monthaug 8.296550400 6.199212740 11.10346611  
## monthsep 8.440319595 6.067186606 11.74168515  
## monthoct 8.460136367 6.203287699 11.53806027  
## monthnov 14.365181873 9.970438835 20.69702785  
## monthdec 12.397151521 7.986921935 19.24262777  
## duration 1.003740386 1.003527216 1.00395360  
## campaign 0.921797330 0.889233954 0.95555316  
## pdays 1.000044075 0.999393208 1.00069537  
## previous 1.011747576 0.988903721 1.03511913  
## poutcomeother 1.110224511 0.902758096 1.36536960  
## poutcomesuccess 6.466458132 5.367875727 7.78987497  
## poutcomeunknown 1.661211808 1.338330800 2.06198996  
## year2010 2.960884042 2.582006848 3.39535672

#Backward:  
Model\_Bwd\_2008<-stepAIC(Model\_Full\_2008,direction="backward",trace=FALSE)  
summary(Model\_Bwd\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + housing +   
## loan + contact + day + month + duration + campaign + poutcome +   
## year, family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.9950 -0.5381 -0.3273 -0.1795 3.0916   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.7008278 0.2138128 -21.986 < 2e-16 \*\*\*  
## jobblue-collar -0.2939209 0.1035453 -2.839 0.00453 \*\*   
## jobentrepreneur -0.4180368 0.1957072 -2.136 0.03268 \*   
## jobhousemaid -0.3321005 0.1987704 -1.671 0.09477 .   
## jobmanagement 0.0071637 0.0977659 0.073 0.94159   
## jobretired 0.1043816 0.1152471 0.906 0.36508   
## jobself-employed -0.1736622 0.1597662 -1.087 0.27705   
## jobservices -0.1278065 0.1188843 -1.075 0.28235   
## jobstudent 0.1244246 0.1302670 0.955 0.33950   
## jobtechnician -0.0138727 0.0942452 -0.147 0.88298   
## jobunemployed -0.2283329 0.1506277 -1.516 0.12955   
## jobunknown -0.0342016 0.3157230 -0.108 0.91374   
## maritalmarried -0.1059043 0.0864863 -1.225 0.22076   
## maritalsingle 0.0164175 0.0920867 0.178 0.85850   
## educationsecondary 0.2687447 0.0933249 2.880 0.00398 \*\*   
## educationtertiary 0.4386674 0.1051525 4.172 3.02e-05 \*\*\*  
## educationunknown 0.2808148 0.1432851 1.960 0.05002 .   
## housingyes -0.6802740 0.0625464 -10.876 < 2e-16 \*\*\*  
## loanyes -0.2778936 0.0949990 -2.925 0.00344 \*\*   
## contacttelephone -0.4701659 0.0995239 -4.724 2.31e-06 \*\*\*  
## contactunknown -1.4258983 0.2287338 -6.234 4.55e-10 \*\*\*  
## day 0.0258086 0.0033827 7.630 2.35e-14 \*\*\*  
## monthfeb 1.3884476 0.1475900 9.407 < 2e-16 \*\*\*  
## monthmar 2.7374611 0.1690176 16.196 < 2e-16 \*\*\*  
## monthapr 1.5136411 0.1332305 11.361 < 2e-16 \*\*\*  
## monthmay 1.2084326 0.1358847 8.893 < 2e-16 \*\*\*  
## monthjun 2.3980977 0.1575701 15.219 < 2e-16 \*\*\*  
## monthjul 2.1281387 0.1735089 12.265 < 2e-16 \*\*\*  
## monthaug 2.1267500 0.1483408 14.337 < 2e-16 \*\*\*  
## monthsep 2.1389486 0.1681809 12.718 < 2e-16 \*\*\*  
## monthoct 2.1485689 0.1579253 13.605 < 2e-16 \*\*\*  
## monthnov 2.6783417 0.1858305 14.413 < 2e-16 \*\*\*  
## monthdec 2.5273175 0.2242108 11.272 < 2e-16 \*\*\*  
## duration 0.0037359 0.0001083 34.500 < 2e-16 \*\*\*  
## campaign -0.0802116 0.0182904 -4.385 1.16e-05 \*\*\*  
## poutcomeother 0.1121676 0.1049346 1.069 0.28510   
## poutcomesuccess 1.8629641 0.0921676 20.213 < 2e-16 \*\*\*  
## poutcomeunknown 0.4629562 0.0691100 6.699 2.10e-11 \*\*\*  
## year2010 1.0976999 0.0691851 15.866 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10249 on 13946 degrees of freedom  
## AIC: 10327  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 10249 on 13946 degrees of freedom  
#AIC: 10327  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + housing +   
# loan + contact + day + month + duration + campaign + poutcome +   
# year, family = binomial(link = "logit"), data = train2008)  
(vif(Model\_Bwd\_2008)[,3])^2

## job marital education housing loan contact day month   
## 1.110102 1.109564 1.291016 1.481903 1.040975 1.104116 1.376434 1.077312   
## duration campaign poutcome year   
## 1.112679 1.041817 1.069001 1.350383

hoslem.test(Model\_Bwd\_2008$y, fitted(Model\_Bwd\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Bwd\_2008$y, fitted(Model\_Bwd\_2008)  
## X-squared = 250.36, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Bwd\_2008), confint.default(Model\_Bwd\_2008, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.009087751 0.005976661 0.01381829  
## jobblue-collar 0.745335421 0.608434518 0.91303973  
## jobentrepreneur 0.658338023 0.448603495 0.96612924  
## jobhousemaid 0.717415232 0.485933603 1.05916654  
## jobmanagement 1.007189454 0.831558197 1.21991534  
## jobretired 1.110023921 0.885592254 1.39133230  
## jobself-employed 0.840580761 0.614590995 1.14966867  
## jobservices 0.880023654 0.697107614 1.11093555  
## jobstudent 1.132496618 0.877310744 1.46190913  
## jobtechnician 0.986223119 0.819886121 1.18630626  
## jobunemployed 0.795859295 0.592409188 1.06917994  
## jobunknown 0.966376638 0.520478155 1.79428050  
## maritalmarried 0.899510723 0.759257410 1.06567223  
## maritalsingle 1.016552996 0.848683420 1.21762717  
## educationsecondary 1.308321048 1.089622395 1.57091482  
## educationtertiary 1.550639519 1.261841763 1.90553442  
## educationunknown 1.324208334 0.999981196 1.75356069  
## housingyes 0.506478192 0.448044527 0.57253274  
## loanyes 0.757377415 0.628708046 0.91237984  
## contacttelephone 0.624898600 0.514155657 0.75949424  
## contactunknown 0.240292500 0.153476421 0.37621731  
## day 1.026144476 1.019363716 1.03297034  
## monthfeb 4.008622396 3.001693226 5.35332971  
## monthmar 15.447715734 11.091652250 21.51455130  
## monthapr 4.543242975 3.499128770 5.89891315  
## monthmay 3.348232608 2.565371817 4.36999484  
## monthjun 11.002226732 8.078983039 14.98319683  
## monthjul 8.399218618 5.977888879 11.80130224  
## monthaug 8.387563056 6.271448210 11.21769832  
## monthsep 8.490506065 6.106294178 11.80563712  
## monthoct 8.572581615 6.290502416 11.68255740  
## monthnov 14.560927212 10.116023704 20.95888735  
## monthdec 12.519876921 8.067731587 19.42892081  
## duration 1.003742932 1.003529922 1.00395599  
## campaign 0.922920999 0.890421627 0.95660656  
## poutcomeother 1.118700298 0.910737531 1.37415042  
## poutcomesuccess 6.442805912 5.378013338 7.71841671  
## poutcomeunknown 1.588763806 1.387499496 1.81922259  
## year2010 2.997263991 2.617185477 3.43253908

#Both:  
Model\_Both\_2008<-stepAIC(Model\_Full\_2008,direction="both",trace=FALSE)  
summary(Model\_Both\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ job + marital + education + housing +   
## loan + contact + day + month + duration + campaign + poutcome +   
## year, family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.9950 -0.5381 -0.3273 -0.1795 3.0916   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.7008278 0.2138128 -21.986 < 2e-16 \*\*\*  
## jobblue-collar -0.2939209 0.1035453 -2.839 0.00453 \*\*   
## jobentrepreneur -0.4180368 0.1957072 -2.136 0.03268 \*   
## jobhousemaid -0.3321005 0.1987704 -1.671 0.09477 .   
## jobmanagement 0.0071637 0.0977659 0.073 0.94159   
## jobretired 0.1043816 0.1152471 0.906 0.36508   
## jobself-employed -0.1736622 0.1597662 -1.087 0.27705   
## jobservices -0.1278065 0.1188843 -1.075 0.28235   
## jobstudent 0.1244246 0.1302670 0.955 0.33950   
## jobtechnician -0.0138727 0.0942452 -0.147 0.88298   
## jobunemployed -0.2283329 0.1506277 -1.516 0.12955   
## jobunknown -0.0342016 0.3157230 -0.108 0.91374   
## maritalmarried -0.1059043 0.0864863 -1.225 0.22076   
## maritalsingle 0.0164175 0.0920867 0.178 0.85850   
## educationsecondary 0.2687447 0.0933249 2.880 0.00398 \*\*   
## educationtertiary 0.4386674 0.1051525 4.172 3.02e-05 \*\*\*  
## educationunknown 0.2808148 0.1432851 1.960 0.05002 .   
## housingyes -0.6802740 0.0625464 -10.876 < 2e-16 \*\*\*  
## loanyes -0.2778936 0.0949990 -2.925 0.00344 \*\*   
## contacttelephone -0.4701659 0.0995239 -4.724 2.31e-06 \*\*\*  
## contactunknown -1.4258983 0.2287338 -6.234 4.55e-10 \*\*\*  
## day 0.0258086 0.0033827 7.630 2.35e-14 \*\*\*  
## monthfeb 1.3884476 0.1475900 9.407 < 2e-16 \*\*\*  
## monthmar 2.7374611 0.1690176 16.196 < 2e-16 \*\*\*  
## monthapr 1.5136411 0.1332305 11.361 < 2e-16 \*\*\*  
## monthmay 1.2084326 0.1358847 8.893 < 2e-16 \*\*\*  
## monthjun 2.3980977 0.1575701 15.219 < 2e-16 \*\*\*  
## monthjul 2.1281387 0.1735089 12.265 < 2e-16 \*\*\*  
## monthaug 2.1267500 0.1483408 14.337 < 2e-16 \*\*\*  
## monthsep 2.1389486 0.1681809 12.718 < 2e-16 \*\*\*  
## monthoct 2.1485689 0.1579253 13.605 < 2e-16 \*\*\*  
## monthnov 2.6783417 0.1858305 14.413 < 2e-16 \*\*\*  
## monthdec 2.5273175 0.2242108 11.272 < 2e-16 \*\*\*  
## duration 0.0037359 0.0001083 34.500 < 2e-16 \*\*\*  
## campaign -0.0802116 0.0182904 -4.385 1.16e-05 \*\*\*  
## poutcomeother 0.1121676 0.1049346 1.069 0.28510   
## poutcomesuccess 1.8629641 0.0921676 20.213 < 2e-16 \*\*\*  
## poutcomeunknown 0.4629562 0.0691100 6.699 2.10e-11 \*\*\*  
## year2010 1.0976999 0.0691851 15.866 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10249 on 13946 degrees of freedom  
## AIC: 10327  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 10249 on 13946 degrees of freedom  
#AIC: 10327  
#Call:  
#glm(formula = as.factor(y) ~ job + marital + education + housing +   
# loan + contact + day + month + duration + campaign + poutcome +   
# year, family = binomial(link = "logit"), data = train2008)  
(vif(Model\_Both\_2008)[,3])^2

## job marital education housing loan contact day month   
## 1.110102 1.109564 1.291016 1.481903 1.040975 1.104116 1.376434 1.077312   
## duration campaign poutcome year   
## 1.112679 1.041817 1.069001 1.350383

hoslem.test(Model\_Both\_2008$y, fitted(Model\_Both\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_Both\_2008$y, fitted(Model\_Both\_2008)  
## X-squared = 250.36, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
exp(cbind("Odds ratio" = coef(Model\_Both\_2008), confint.default(Model\_Both\_2008, level = 0.95)))

## Odds ratio 2.5 % 97.5 %  
## (Intercept) 0.009087751 0.005976661 0.01381829  
## jobblue-collar 0.745335421 0.608434518 0.91303973  
## jobentrepreneur 0.658338023 0.448603495 0.96612924  
## jobhousemaid 0.717415232 0.485933603 1.05916654  
## jobmanagement 1.007189454 0.831558197 1.21991534  
## jobretired 1.110023921 0.885592254 1.39133230  
## jobself-employed 0.840580761 0.614590995 1.14966867  
## jobservices 0.880023654 0.697107614 1.11093555  
## jobstudent 1.132496618 0.877310744 1.46190913  
## jobtechnician 0.986223119 0.819886121 1.18630626  
## jobunemployed 0.795859295 0.592409188 1.06917994  
## jobunknown 0.966376638 0.520478155 1.79428050  
## maritalmarried 0.899510723 0.759257410 1.06567223  
## maritalsingle 1.016552996 0.848683420 1.21762717  
## educationsecondary 1.308321048 1.089622395 1.57091482  
## educationtertiary 1.550639519 1.261841763 1.90553442  
## educationunknown 1.324208334 0.999981196 1.75356069  
## housingyes 0.506478192 0.448044527 0.57253274  
## loanyes 0.757377415 0.628708046 0.91237984  
## contacttelephone 0.624898600 0.514155657 0.75949424  
## contactunknown 0.240292500 0.153476421 0.37621731  
## day 1.026144476 1.019363716 1.03297034  
## monthfeb 4.008622396 3.001693226 5.35332971  
## monthmar 15.447715734 11.091652250 21.51455130  
## monthapr 4.543242975 3.499128770 5.89891315  
## monthmay 3.348232608 2.565371817 4.36999484  
## monthjun 11.002226732 8.078983039 14.98319683  
## monthjul 8.399218618 5.977888879 11.80130224  
## monthaug 8.387563056 6.271448210 11.21769832  
## monthsep 8.490506065 6.106294178 11.80563712  
## monthoct 8.572581615 6.290502416 11.68255740  
## monthnov 14.560927212 10.116023704 20.95888735  
## monthdec 12.519876921 8.067731587 19.42892081  
## duration 1.003742932 1.003529922 1.00395599  
## campaign 0.922920999 0.890421627 0.95660656  
## poutcomeother 1.118700298 0.910737531 1.37415042  
## poutcomesuccess 6.442805912 5.378013338 7.71841671  
## poutcomeunknown 1.588763806 1.387499496 1.81922259  
## year2010 2.997263991 2.617185477 3.43253908

#Simple model : All predictors with no day and days  
Model\_SIM1\_2008<-glm(as.factor(y) ~ age+job+marital+education+balance+housing+loan+contact+month+duration+campaign+previous+poutcome+year,data=train2008,family= binomial(link="logit"))  
summary(Model\_SIM1\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + marital + education +   
## balance + housing + loan + contact + month + duration + campaign +   
## previous + poutcome + year, family = binomial(link = "logit"),   
## data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.9199 -0.5409 -0.3327 -0.1773 2.9140   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.238e+00 2.416e-01 -17.539 < 2e-16 \*\*\*  
## age 3.749e-03 2.988e-03 1.255 0.20959   
## jobblue-collar -3.046e-01 1.031e-01 -2.954 0.00313 \*\*   
## jobentrepreneur -4.586e-01 1.950e-01 -2.352 0.01868 \*   
## jobhousemaid -3.490e-01 2.001e-01 -1.745 0.08103 .   
## jobmanagement -4.210e-03 9.764e-02 -0.043 0.96561   
## jobretired 4.459e-02 1.326e-01 0.336 0.73668   
## jobself-employed -1.930e-01 1.595e-01 -1.210 0.22616   
## jobservices -1.324e-01 1.185e-01 -1.117 0.26384   
## jobstudent 1.723e-01 1.321e-01 1.304 0.19227   
## jobtechnician -1.289e-02 9.387e-02 -0.137 0.89079   
## jobunemployed -2.457e-01 1.507e-01 -1.630 0.10300   
## jobunknown -9.834e-02 3.168e-01 -0.310 0.75622   
## maritalmarried -8.628e-02 8.652e-02 -0.997 0.31864   
## maritalsingle 7.711e-02 9.800e-02 0.787 0.43140   
## educationsecondary 2.778e-01 9.339e-02 2.975 0.00293 \*\*   
## educationtertiary 4.579e-01 1.055e-01 4.341 1.42e-05 \*\*\*  
## educationunknown 2.897e-01 1.426e-01 2.031 0.04222 \*   
## balance 2.797e-06 7.267e-06 0.385 0.70031   
## housingyes -7.188e-01 6.259e-02 -11.486 < 2e-16 \*\*\*  
## loanyes -2.791e-01 9.488e-02 -2.941 0.00327 \*\*   
## contacttelephone -5.192e-01 1.010e-01 -5.140 2.74e-07 \*\*\*  
## contactunknown -1.354e+00 2.277e-01 -5.945 2.76e-09 \*\*\*  
## monthfeb 9.037e-01 1.337e-01 6.758 1.40e-11 \*\*\*  
## monthmar 2.424e+00 1.642e-01 14.761 < 2e-16 \*\*\*  
## monthapr 1.330e+00 1.319e-01 10.082 < 2e-16 \*\*\*  
## monthmay 9.035e-01 1.309e-01 6.900 5.20e-12 \*\*\*  
## monthjun 1.963e+00 1.470e-01 13.348 < 2e-16 \*\*\*  
## monthjul 1.787e+00 1.670e-01 10.701 < 2e-16 \*\*\*  
## monthaug 1.789e+00 1.431e-01 12.501 < 2e-16 \*\*\*  
## monthsep 1.754e+00 1.607e-01 10.916 < 2e-16 \*\*\*  
## monthoct 1.909e+00 1.557e-01 12.258 < 2e-16 \*\*\*  
## monthnov 2.313e+00 1.803e-01 12.824 < 2e-16 \*\*\*  
## monthdec 2.210e+00 2.203e-01 10.032 < 2e-16 \*\*\*  
## duration 3.695e-03 1.078e-04 34.266 < 2e-16 \*\*\*  
## campaign -7.573e-02 1.828e-02 -4.142 3.45e-05 \*\*\*  
## previous 9.654e-03 1.159e-02 0.833 0.40503   
## poutcomeother 1.262e-01 1.049e-01 1.203 0.22910   
## poutcomesuccess 1.870e+00 9.207e-02 20.308 < 2e-16 \*\*\*  
## poutcomeunknown 4.921e-01 7.656e-02 6.428 1.29e-10 \*\*\*  
## year2010 1.110e+00 6.938e-02 16.000 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10305 on 13944 degrees of freedom  
## AIC: 10387  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 10305 on 13944 degrees of freedom  
#AIC: 10387  
#Call:  
#glm(formula = as.factor(y) ~ age + job + marital + education +   
# balance + housing + loan + contact + month + duration + campaign +   
# previous + poutcome + year, family = binomial(link = "logit"),   
# data = train2008)  
(vif(Model\_SIM1\_2008)[,3])^2

## age job marital education balance housing loan contact   
## 2.690883 1.156632 1.247599 1.300771 1.040105 1.492614 1.043610 1.121691   
## month duration campaign previous poutcome year   
## 1.049650 1.106841 1.045272 1.566237 1.196148 1.368529

hoslem.test(Model\_SIM1\_2008$y, fitted(Model\_SIM1\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM1\_2008$y, fitted(Model\_SIM1\_2008)  
## X-squared = 225.06, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#Simple model2 by hand selecting the predictors  
Model\_SIM2\_2008<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+previous+year,data=train2008,family= binomial(link="logit"))  
summary(Model\_SIM2\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + previous + year, family = binomial(link = "logit"),   
## data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.8560 -0.6057 -0.4425 -0.2539 2.7217   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.945e+00 1.552e-01 -18.972 < 2e-16 \*\*\*  
## age 6.226e-03 2.451e-03 2.540 0.011081 \*   
## jobblue-collar -6.344e-01 9.487e-02 -6.687 2.27e-11 \*\*\*  
## jobentrepreneur -6.577e-01 1.812e-01 -3.630 0.000283 \*\*\*  
## jobhousemaid -1.307e-01 1.850e-01 -0.706 0.479944   
## jobmanagement -2.288e-02 9.044e-02 -0.253 0.800268   
## jobretired 2.659e-01 1.220e-01 2.180 0.029233 \*   
## jobself-employed -1.835e-01 1.468e-01 -1.250 0.211377   
## jobservices -3.621e-01 1.091e-01 -3.320 0.000901 \*\*\*  
## jobstudent 5.954e-01 1.210e-01 4.919 8.70e-07 \*\*\*  
## jobtechnician -4.848e-02 8.669e-02 -0.559 0.576006   
## jobunemployed -1.441e-01 1.356e-01 -1.063 0.287904   
## jobunknown 1.943e-02 2.958e-01 0.066 0.947624   
## educationsecondary 3.138e-01 8.716e-02 3.600 0.000318 \*\*\*  
## educationtertiary 6.554e-01 9.829e-02 6.668 2.60e-11 \*\*\*  
## educationunknown 4.250e-01 1.329e-01 3.198 0.001383 \*\*   
## balance 2.440e-05 6.810e-06 3.583 0.000340 \*\*\*  
## loanyes -6.142e-01 8.868e-02 -6.926 4.33e-12 \*\*\*  
## duration 3.450e-03 9.949e-05 34.672 < 2e-16 \*\*\*  
## campaign -1.022e-01 1.709e-02 -5.979 2.24e-09 \*\*\*  
## previous 1.023e-02 8.525e-03 1.200 0.230169   
## year2010 1.526e+00 5.615e-02 27.177 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 11787 on 13963 degrees of freedom  
## AIC: 11831  
##   
## Number of Fisher Scoring iterations: 5

#Null deviance: 14819 on 13984 degrees of freedom  
#Residual deviance: 11787 on 13963 degrees of freedom  
#AIC: 11831  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + previous + year, family = binomial(link = "logit"),   
# data = train2008)  
hoslem.test(Model\_SIM2\_2008$y, fitted(Model\_SIM2\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM2\_2008$y, fitted(Model\_SIM2\_2008)  
## X-squared = 147.38, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#With interactions:  
#Simple model2 by hand seleting the predictors  
Model\_SIM3\_2008<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+previous+year+year\*month+year\*day,data=train2008,family= binomial(link="logit"))  
summary(Model\_SIM3\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + previous + year + year \* month +   
## year \* day, family = binomial(link = "logit"), data = train2008)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.3216 -0.5601 -0.3191 -0.1342 2.8255   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.862e+00 3.138e-01 -21.867 < 2e-16 \*\*\*  
## age -6.328e-04 2.582e-03 -0.245 0.806392   
## jobblue-collar -3.431e-01 1.010e-01 -3.395 0.000686 \*\*\*  
## jobentrepreneur -3.506e-01 1.926e-01 -1.821 0.068627 .   
## jobhousemaid -3.379e-01 1.951e-01 -1.732 0.083276 .   
## jobmanagement 1.501e-02 9.508e-02 0.158 0.874588   
## jobretired 3.322e-02 1.289e-01 0.258 0.796545   
## jobself-employed -1.206e-01 1.557e-01 -0.775 0.438516   
## jobservices -1.225e-01 1.164e-01 -1.053 0.292549   
## jobstudent 2.801e-01 1.269e-01 2.207 0.027304 \*   
## jobtechnician -2.397e-02 9.162e-02 -0.262 0.793603   
## jobunemployed -3.237e-02 1.454e-01 -0.223 0.823889   
## jobunknown 9.696e-03 3.013e-01 0.032 0.974329   
## educationsecondary 3.149e-01 9.235e-02 3.410 0.000650 \*\*\*  
## educationtertiary 4.982e-01 1.042e-01 4.779 1.76e-06 \*\*\*  
## educationunknown 3.296e-01 1.401e-01 2.352 0.018667 \*   
## balance 3.708e-06 7.236e-06 0.512 0.608405   
## loanyes -4.111e-01 9.329e-02 -4.407 1.05e-05 \*\*\*  
## duration 4.028e-03 1.108e-04 36.365 < 2e-16 \*\*\*  
## campaign -9.250e-02 1.788e-02 -5.175 2.28e-07 \*\*\*  
## previous 9.263e-03 9.159e-03 1.011 0.311855   
## year2010 5.575e+00 3.280e-01 16.996 < 2e-16 \*\*\*  
## monthfeb 3.231e+00 2.655e-01 12.171 < 2e-16 \*\*\*  
## monthmar 5.116e+00 2.859e-01 17.896 < 2e-16 \*\*\*  
## monthapr 2.924e+00 2.444e-01 11.964 < 2e-16 \*\*\*  
## monthmay 2.647e+00 2.479e-01 10.678 < 2e-16 \*\*\*  
## monthjun 4.764e+00 2.710e-01 17.578 < 2e-16 \*\*\*  
## monthjul 4.556e+00 3.076e-01 14.811 < 2e-16 \*\*\*  
## monthaug 4.220e+00 2.582e-01 16.345 < 2e-16 \*\*\*  
## monthsep 4.685e+00 2.880e-01 16.269 < 2e-16 \*\*\*  
## monthoct 4.485e+00 2.673e-01 16.780 < 2e-16 \*\*\*  
## monthnov 5.066e+00 2.842e-01 17.828 < 2e-16 \*\*\*  
## monthdec 4.587e+00 3.013e-01 15.223 < 2e-16 \*\*\*  
## day 5.058e-02 4.201e-03 12.041 < 2e-16 \*\*\*  
## year2010:monthfeb -3.136e+00 3.342e-01 -9.385 < 2e-16 \*\*\*  
## year2010:monthmar -4.816e+00 3.630e-01 -13.268 < 2e-16 \*\*\*  
## year2010:monthapr -2.650e+00 3.328e-01 -7.963 1.67e-15 \*\*\*  
## year2010:monthmay -2.456e+00 3.327e-01 -7.381 1.57e-13 \*\*\*  
## year2010:monthjun -4.715e+00 3.533e-01 -13.349 < 2e-16 \*\*\*  
## year2010:monthjul -4.367e+00 3.723e-01 -11.732 < 2e-16 \*\*\*  
## year2010:monthaug -4.122e+00 3.427e-01 -12.027 < 2e-16 \*\*\*  
## year2010:monthsep -4.717e+00 3.587e-01 -13.148 < 2e-16 \*\*\*  
## year2010:monthoct -4.749e+00 3.505e-01 -13.551 < 2e-16 \*\*\*  
## year2010:monthnov -5.275e+00 4.382e-01 -12.039 < 2e-16 \*\*\*  
## year2010:monthdec NA NA NA NA   
## year2010:day -4.951e-02 7.064e-03 -7.008 2.42e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 14819 on 13984 degrees of freedom  
## Residual deviance: 10528 on 13940 degrees of freedom  
## AIC: 10618  
##   
## Number of Fisher Scoring iterations: 6

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18378 on 41660 degrees of freedom  
#AIC: 18484  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + previous + year + year \* month +   
# year \* day, family = binomial(link = "logit"), data = train)  
hoslem.test(Model\_SIM3\_2008$y, fitted(Model\_SIM3\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM3\_2008$y, fitted(Model\_SIM3\_2008)  
## X-squared = 136.66, df = 8, p-value < 2.2e-16

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#With interactions:  
#Simple model2 by hand seleting the predictors  
Model\_SIM4\_2008<-glm(as.factor(y) ~ age+job+education+balance+loan+duration+campaign+pdays+previous+year+I(duration^2),I(pdays^2),data=train2008,family= binomial(link="logit"))

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

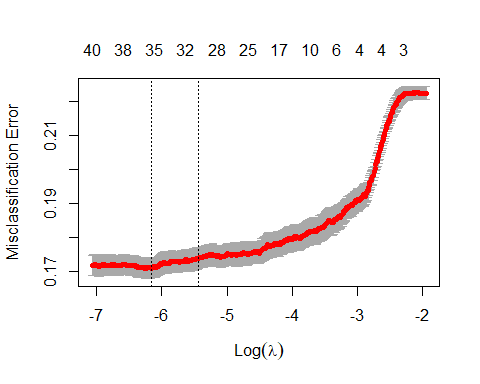
summary(Model\_SIM4\_2008)

##   
## Call:  
## glm(formula = as.factor(y) ~ age + job + education + balance +   
## loan + duration + campaign + pdays + previous + year + I(duration^2),   
## family = binomial(link = "logit"), data = train2008, weights = I(pdays^2))  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5722 0 0 0 7251   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.467e+15 2.470e+04 -5.940e+10 <2e-16 \*\*\*  
## age 7.943e+12 3.824e+02 2.077e+10 <2e-16 \*\*\*  
## jobblue-collar -3.951e+14 1.175e+04 -3.363e+10 <2e-16 \*\*\*  
## jobentrepreneur -4.723e+14 2.382e+04 -1.982e+10 <2e-16 \*\*\*  
## jobhousemaid -7.001e+14 3.500e+04 -2.000e+10 <2e-16 \*\*\*  
## jobmanagement 1.463e+14 1.410e+04 1.038e+10 <2e-16 \*\*\*  
## jobretired -1.051e+14 2.150e+04 -4.889e+09 <2e-16 \*\*\*  
## jobself-employed -5.030e+13 2.296e+04 -2.191e+09 <2e-16 \*\*\*  
## jobservices -4.301e+13 1.404e+04 -3.063e+09 <2e-16 \*\*\*  
## jobstudent 3.658e+14 2.299e+04 1.591e+10 <2e-16 \*\*\*  
## jobtechnician -6.461e+13 1.245e+04 -5.189e+09 <2e-16 \*\*\*  
## jobunemployed 4.474e+14 2.339e+04 1.912e+10 <2e-16 \*\*\*  
## jobunknown 5.118e+14 5.860e+04 8.733e+09 <2e-16 \*\*\*  
## educationsecondary 1.385e+14 1.092e+04 1.268e+10 <2e-16 \*\*\*  
## educationtertiary 6.448e+13 1.419e+04 4.542e+09 <2e-16 \*\*\*  
## educationunknown 6.759e+12 2.002e+04 3.376e+08 <2e-16 \*\*\*  
## balance 1.326e+10 1.342e+00 9.875e+09 <2e-16 \*\*\*  
## loanyes -3.734e+14 9.825e+03 -3.800e+10 <2e-16 \*\*\*  
## duration 5.237e+12 3.379e+01 1.550e+11 <2e-16 \*\*\*  
## campaign -1.411e+14 2.102e+03 -6.715e+10 <2e-16 \*\*\*  
## pdays -5.144e+11 2.834e+01 -1.815e+10 <2e-16 \*\*\*  
## previous 2.490e+13 8.162e+02 3.051e+10 <2e-16 \*\*\*  
## year2010 -9.312e+14 9.368e+03 -9.940e+10 <2e-16 \*\*\*  
## I(duration^2) -2.546e+09 2.845e-02 -8.948e+10 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 406634274 on 13984 degrees of freedom  
## Residual deviance: 7155302233 on 13961 degrees of freedom  
## AIC: 7155302281  
##   
## Number of Fisher Scoring iterations: 25

#Null deviance: 30188 on 41712 degrees of freedom  
#Residual deviance: 18378 on 41660 degrees of freedom  
#AIC: 18484  
#Call:  
#glm(formula = as.factor(y) ~ age + job + education + balance +   
# loan + duration + campaign + previous + year + year \* month +   
# year \* day, family = binomial(link = "logit"), data = train)  
hoslem.test(Model\_SIM4\_2008$y, fitted(Model\_SIM4\_2008), g=10)

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: Model\_SIM4\_2008$y, fitted(Model\_SIM4\_2008)  
## X-squared = 38.139, df = 8, p-value = 7.096e-06

#p-value < 2.2e-16  
# rejects. But the sample size is large, so I don't think it's an issue  
  
#LASSO  
#LASSO is obtained in the exact same way for logistic as is MLR. The only difference is to let R know that our response is categorical through the "family="binomial"" option. Cross validation is used to obtain the optimal penalty value. A final refit using the entire data set can then be obtained once the optimal penalty value is determined. For this example, the object "final model" produces the final lasso model  
set.seed(1234)  
  
dat.train.x.2008 <- model.matrix(y~age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year-1,train2008)  
dat.train.y.2008<-train2008$y  
cvfit.2008 <- cv.glmnet(dat.train.x.2008, dat.train.y.2008, family = "binomial", type.measure = "class", nlambda = 1000)  
plot(cvfit.2008)



coef(cvfit.2008, s = "lambda.min")

## 45 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) -3.240513e+00  
## age .   
## jobadmin. .   
## jobblue-collar -2.714744e-01  
## jobentrepreneur -2.843122e-01  
## jobhousemaid -2.380532e-01  
## jobmanagement 1.665240e-02  
## jobretired 5.077271e-02  
## jobself-employed -4.985861e-02  
## jobservices -5.238727e-02  
## jobstudent 1.703424e-01  
## jobtechnician .   
## jobunemployed -1.462968e-01  
## jobunknown .   
## maritalmarried -7.252364e-02  
## maritalsingle 2.405167e-02  
## educationsecondary 3.109068e-02  
## educationtertiary 2.141293e-01  
## educationunknown 1.950241e-03  
## defaultyes .   
## balance 3.193580e-06  
## housingyes -6.344519e-01  
## loanyes -2.347192e-01  
## contacttelephone -3.981903e-01  
## contactunknown -1.113612e+00  
## day 1.193664e-02  
## monthfeb 2.590817e-01  
## monthmar 1.688092e+00  
## monthapr 5.357427e-01  
## monthmay 1.408510e-01  
## monthjun 1.294212e+00  
## monthjul 1.065935e+00  
## monthaug 1.086944e+00  
## monthsep 1.059203e+00  
## monthoct 1.150269e+00  
## monthnov 1.572423e+00  
## monthdec 1.419337e+00  
## duration 3.552945e-03  
## campaign -5.710122e-02  
## pdays .   
## previous .   
## poutcomeother .   
## poutcomesuccess 1.742989e+00  
## poutcomeunknown 3.212755e-01  
## year2010 1.024925e+00

#CV misclassification error rate is little below .1  
print("CV Error Rate:")

## [1] "CV Error Rate:"

cvfit.2008$cvm[which(cvfit.2008$lambda==cvfit.2008$lambda.min)]

## [1] 0.1708259

#Optimal penalty  
print("Penalty Value:")

## [1] "Penalty Value:"

cvfit.2008$lambda.min

## [1] 0.002119464

#[1] "CV Error Rate:"  
#[1] 0.1708259  
#[1] "Penalty Value:"  
#[1] 0.002119464  
  
#For final model predictions go ahead and refit lasso using entire  
#data set  
finalmodel.2008<-glmnet(dat.train.x.2008, dat.train.y.2008, family = "binomial",lambda=cvfit.2008$lambda.min)

set.seed(1234)  
#For all years  
#Lets compare the stepwise and lasso models using the test set.However the true predictions from the models are predictive #probabalities. To help get a handle on this, the following code makes predictions on the test set using the LASSO model.  
dat.test.x.2008<-model.matrix(y~age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year-1,test2008)  
dat.test.y.2008<-test2008$y  
fit.pred.lasso.2008 <- predict(finalmodel.2008, newx = dat.test.x.2008, type = "response")  
  
test2008$y[1:15]

## [1] "no" "no" "no" "no" "yes" "no" "no" "no" "no" "no" "yes" "no"   
## [13] "no" "no" "no"

fit.pred.lasso.2008[1:15]

## [1] 0.21450912 0.04916310 0.19383818 0.05610022 0.98935093 0.02869177  
## [7] 0.10780890 0.09585333 0.08806205 0.08904300 0.65138843 0.07089809  
## [13] 0.03044869 0.08533746 0.05276506

#Making predictions for stepwise as well for later  
fit.pred.step.2008<-predict(Model\_Step\_2008,newdata=test2008,type="response")  
summary(fit.pred.step.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.007989 0.048285 0.112341 0.224392 0.314418 0.999707

#Making predictions for backward as well for later  
fit.pred.bck.2008<-predict(Model\_Bwd\_2008,newdata=test2008,type="response")  
summary(fit.pred.bck.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.007989 0.048285 0.112341 0.224392 0.314418 0.999707

#Making predictions for backward as well for later  
fit.pred.fwd.2008<-predict(Model\_FWD\_2008,newdata=test2008,type="response")  
summary(fit.pred.fwd.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.007955 0.048043 0.111541 0.224306 0.313898 0.999718

#Making predictions for Both as well for later  
fit.pred.both.2008<-predict(Model\_Both\_2008,newdata=test2008,type="response")  
summary(fit.pred.both.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.007989 0.048285 0.112341 0.224392 0.314418 0.999707

#Making predictions for SIM1 as well for later  
fit.pred.SIM1.2008<-predict(Model\_SIM1\_2008,newdata=test2008,type="response")  
summary(fit.pred.SIM1.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.007231 0.048379 0.112879 0.225013 0.315873 0.999686

#Making predictions for SIM2 as well for later  
fit.pred.SIM2.2008<-predict(Model\_SIM2\_2008,newdata=test2008,type="response")  
summary(fit.pred.SIM2.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.01285 0.08499 0.14108 0.22062 0.27026 0.99970

#Making predictions for SIM3 as well for later  
fit.pred.SIM3.2008<-predict(Model\_SIM3\_2008,newdata=test2008,type="response")

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
## prediction from a rank-deficient fit may be misleading

summary(fit.pred.SIM3.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00240 0.04735 0.10984 0.22016 0.34913 0.99990

#Making predictions for SIM4 as well for later  
fit.pred.SIM4.2008<-predict(Model\_SIM4\_2008,newdata=test2008,type="response")  
summary(fit.pred.SIM4.2008)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 0.0000 0.0000 0.2451 0.0000 1.0000

#Confusion Matrix  
#Lets use the predicted probablities to classify the observations and make a final confusion matrix for the two models. We can use it to calculate error metrics.  
#Lets use a cutoff of 0.5 to make the classification.  
cutoff<-0.5  
class.lasso.2008<-factor(ifelse(fit.pred.lasso.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.step.2008<-factor(ifelse(fit.pred.step.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.bck.2008<-factor(ifelse(fit.pred.bck.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.fwd.2008<-factor(ifelse(fit.pred.fwd.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.both.2008<-factor(ifelse(fit.pred.both.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM1.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM2.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM3.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM4.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
  
#Confusion Matrix for Lasso  
conf.lasso.2008<-table(class.lasso.2008,test2008$y)  
print("Confusion matrix for LASSO")

## [1] "Confusion matrix for LASSO"

conf.lasso.2008

##   
## class.lasso.2008 no yes  
## Low 2539 444  
## High 178 335

#class.lasso.2008 no yes  
# Low 2539 444  
# High 178 335  
  
#Confusion Matrix for Stepwise  
conf.step.2008<-table(class.step.2008,test2008$y)  
print("Confusion matrix for Stepwise")

## [1] "Confusion matrix for Stepwise"

conf.step.2008

##   
## class.step.2008 no yes  
## Low 2534 426  
## High 183 353

#class.step.2008 no yes  
# Low 2534 426  
# High 183 353  
  
#Confusion Matrix for Backward  
conf.bck.2008<-table(class.bck.2008,test2008$y)  
print("Confusion matrix for Backward")

## [1] "Confusion matrix for Backward"

conf.bck.2008

##   
## class.bck.2008 no yes  
## Low 2534 426  
## High 183 353

#class.bck.2008 no yes  
# Low 2534 426  
# High 183 353  
  
#Confusion Matrix for Forward  
conf.fwd.2008<-table(class.fwd.2008,test2008$y)  
print("Confusion matrix for Forward")

## [1] "Confusion matrix for Forward"

conf.fwd.2008

##   
## class.fwd.2008 no yes  
## Low 2531 425  
## High 186 354

#class.fwd.2008 no yes  
# Low 2531 425  
# High 186 354  
  
#Confusion Matrix for Both  
conf.both.2008<-table(class.both.2008,test2008$y)  
print("Confusion matrix for Both")

## [1] "Confusion matrix for Both"

conf.both.2008

##   
## class.both.2008 no yes  
## Low 2534 426  
## High 183 353

#class.both.2008 no yes  
# Low 2534 426  
# High 183 353  
  
#Accuracy  
print("Overall accuracy for LASSO")

## [1] "Overall accuracy for LASSO"

sum(diag(conf.lasso.2008))/sum(conf.lasso.2008)

## [1] 0.8220824

#0.8220824  
print("Overall accuracy for Step")

## [1] "Overall accuracy for Step"

sum(diag(conf.step.2008))/sum(conf.step.2008)

## [1] 0.8258009

#0.8258009  
print("Overall accuracy for Backward")

## [1] "Overall accuracy for Backward"

sum(diag(conf.bck.2008))/sum(conf.bck.2008)

## [1] 0.8258009

#0.8258009  
print("Overall accuracy for Forward")

## [1] "Overall accuracy for Forward"

#0.8252288  
sum(diag(conf.fwd.2008))/sum(conf.fwd.2008)

## [1] 0.8252288

print("Overall accuracy for Both")

## [1] "Overall accuracy for Both"

sum(diag(conf.both.2008))/sum(conf.both.2008)

## [1] 0.8258009

#0.8258009  
  
#Confusion Matrix  
#Lets use the predicted probablities to classify the observations and make a final confusion matrix for the two models. We can use it to calculate error metrics.  
#Lets use a cutoff of 0.3 to make the classification.  
cutoff<-0.3  
class.lasso.2008<-factor(ifelse(fit.pred.lasso.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.step.2008<-factor(ifelse(fit.pred.step.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.bck.2008<-factor(ifelse(fit.pred.bck.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.fwd.2008<-factor(ifelse(fit.pred.fwd.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.both.2008<-factor(ifelse(fit.pred.both.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM1.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM2.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM3.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
class.SIM4.2008<-factor(ifelse(fit.pred.SIM1.2008>cutoff,"High","Low"),levels=c("Low","High"))  
  
#Confusion Matrix for Lasso  
conf.lasso.2008<-table(class.lasso.2008,test2008$y)  
print("Confusion matrix for LASSO")

## [1] "Confusion matrix for LASSO"

conf.lasso.2008

##   
## class.lasso.2008 no yes  
## Low 2352 258  
## High 365 521

#class.lasso.2008 no yes  
# Low 2352 258  
# High 365 521  
  
#Confusion Matrix for Stepwise  
conf.step.2008<-table(class.step.2008,test2008$y)  
print("Confusion matrix for Stepwise")

## [1] "Confusion matrix for Stepwise"

conf.step.2008

##   
## class.step.2008 no yes  
## Low 2340 249  
## High 377 530

#class.step.2008 no yes  
# Low 2340 249  
# High 377 530  
  
#Confusion Matrix for Backward  
conf.bck.2008<-table(class.bck.2008,test2008$y)  
print("Confusion matrix for Backward")

## [1] "Confusion matrix for Backward"

conf.bck.2008

##   
## class.bck.2008 no yes  
## Low 2340 249  
## High 377 530

#class.bck.2008 no yes  
# Low 2340 249  
# High 377 530  
  
#Confusion Matrix for Forward  
conf.fwd.2008<-table(class.fwd.2008,test2008$y)  
print("Confusion matrix for Forward")

## [1] "Confusion matrix for Forward"

conf.fwd.2008

##   
## class.fwd.2008 no yes  
## Low 2341 249  
## High 376 530

#class.fwd.2008 no yes  
# Low 2341 249  
# High 376 530  
  
#Confusion Matrix for Both  
conf.both.2008<-table(class.both.2008,test2008$y)  
print("Confusion matrix for Both")

## [1] "Confusion matrix for Both"

conf.both.2008

##   
## class.both.2008 no yes  
## Low 2340 249  
## High 377 530

#class.both.2008 no yes  
# Low 2340 249  
# High 377 530  
  
#Accuracy  
print("Overall accuracy for LASSO")

## [1] "Overall accuracy for LASSO"

sum(diag(conf.lasso.2008))/sum(conf.lasso.2008)

## [1] 0.8217963

#0.8217963  
print("Overall accuracy for Step")

## [1] "Overall accuracy for Step"

sum(diag(conf.step.2008))/sum(conf.step.2008)

## [1] 0.8209382

#0.8209382  
print("Overall accuracy for Backward")

## [1] "Overall accuracy for Backward"

sum(diag(conf.bck.2008))/sum(conf.bck.2008)

## [1] 0.8209382

#0.8209382  
print("Overall accuracy for Forward")

## [1] "Overall accuracy for Forward"

#0.8212243  
sum(diag(conf.fwd.2008))/sum(conf.fwd.2008)

## [1] 0.8212243

print("Overall accuracy for Both")

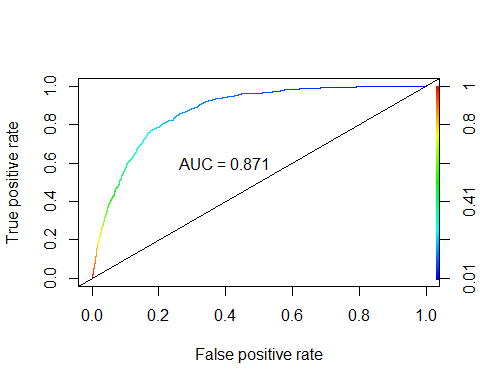
## [1] "Overall accuracy for Both"

sum(diag(conf.both.2008))/sum(conf.both.2008)

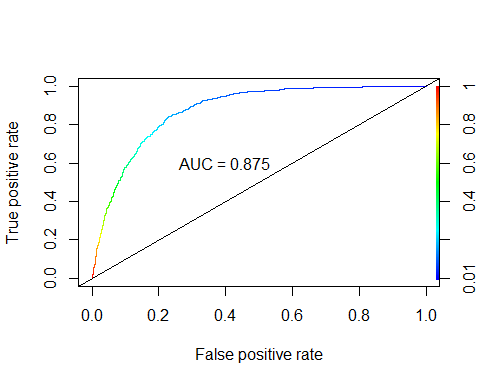
## [1] 0.8209382

#0.8209382

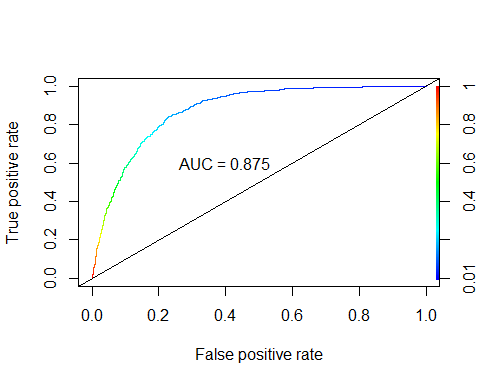
set.seed(1234)  
#ROC curves: LASSO model on test set  
pred1 <- prediction(fit.pred.lasso.2008[,1], dat.test.y.2008)  
roc.perf1 = performance(pred1, measure = "tpr", x.measure = "fpr")  
auc.val1 <- performance(pred1, measure = "auc")  
auc.val1 <- auc.val1@y.values  
plot(roc.perf1, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.val1[[1]],3), sep = ""))



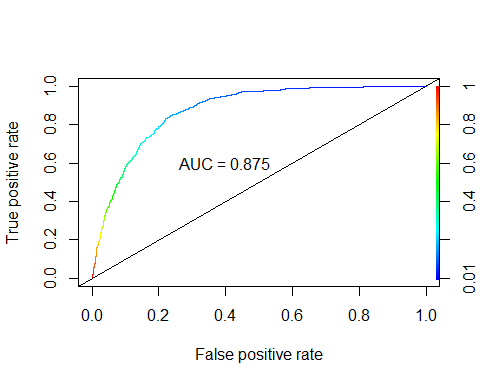
# AUC=0.871  
  
# Code from HW 12  
#ROC curves: Step model on test set  
results.step<-prediction(fit.pred.step.2008, test2008$y,label.ordering=c("no","yes"))  
roc.step = performance(results.step, measure = "tpr", x.measure = "fpr")  
auc.val.step = performance(results.step, measure="auc")  
auc.val.step = auc.val.step@y.values  
plot(roc.step,colorize = TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.val.step[[1]],3), sep = ""))



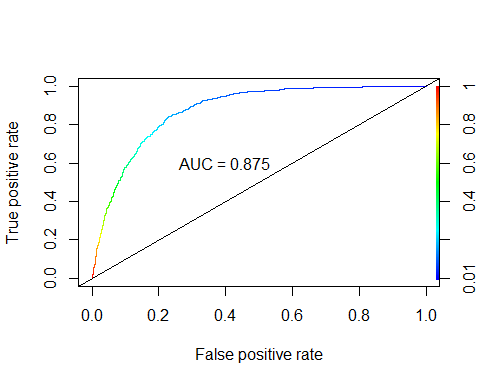
# AUC=0.875  
  
# Code from HW 12  
#ROC curves: Backward model on test set  
results.step<-prediction(fit.pred.bck.2008, test2008$y,label.ordering=c("no","yes"))  
roc.step = performance(results.step, measure = "tpr", x.measure = "fpr")  
auc.val.step = performance(results.step, measure="auc")  
auc.val.step = auc.val.step@y.values  
plot(roc.step,colorize = TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.val.step[[1]],3), sep = ""))



# AUC=0.875  
  
# Code from HW 12  
#ROC curves: Forward model on test set  
results.step<-prediction(fit.pred.fwd.2008, test2008$y,label.ordering=c("no","yes"))  
roc.step = performance(results.step, measure = "tpr", x.measure = "fpr")  
auc.val.step = performance(results.step, measure="auc")  
auc.val.step = auc.val.step@y.values  
plot(roc.step,colorize = TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.val.step[[1]],3), sep = ""))

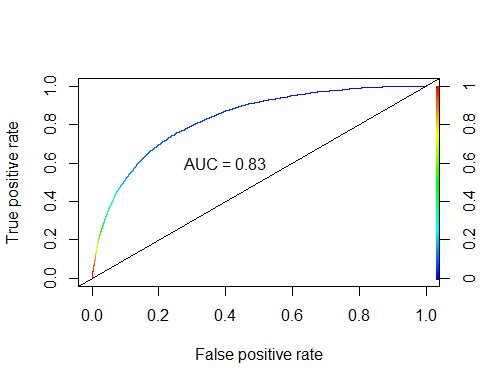


# AUC=0.875  
  
# Code from HW 12  
#ROC curves: Both model on test set  
results.step<-prediction(fit.pred.both.2008, test2008$y,label.ordering=c("no","yes"))  
roc.step = performance(results.step, measure = "tpr", x.measure = "fpr")  
auc.val.step = performance(results.step, measure="auc")  
auc.val.step = auc.val.step@y.values  
plot(roc.step,colorize = TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.val.step[[1]],3), sep = ""))

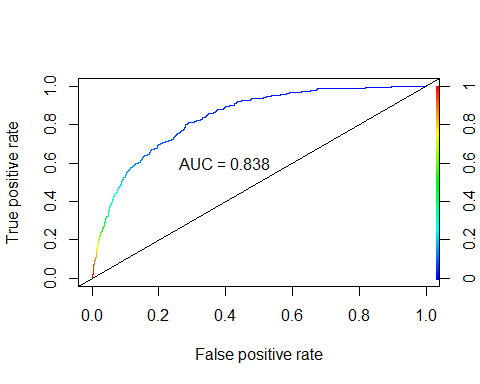


# AUC=0.875

#All years  
#Training Set  
lda.train.x <- train[,c(1,6,10,12,13,14,15)]  
lda.train.y <- train$y  
fit.lda <- lda(lda.train.y ~ ., data = lda.train.x)  
pred.lda <- predict(fit.lda, newdata = lda.train.x)  
preds <- pred.lda$posterior  
preds <- as.data.frame(preds)  
# ROC for TRAINING data  
pred <- prediction(preds[,2],lda.train.y)  
roc.perf = performance(pred, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC=0.83. Not surprising. Maybe better if we normalize and then weed out some variables  
  
#LDA for TEST data  
lda.test.x <- test[,c(1,6,10,12,13,14,15)]  
lda.test.y <- test$y  
pred.lda1 <- predict(fit.lda, newdata = lda.test.x)  
preds1 <- pred.lda1$posterior  
preds1 <- as.data.frame(preds1)  
pred1 <- prediction(preds1[,2],lda.test.y)  
roc.perf = performance(pred1, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred1, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC = 0.838  
  
  
pred.lda.cm<-predict(fit.lda,newdata=lda.test.x)$class #Predictions can come in many forms, the class form provides the categorical level of your response.  
Truth<-lda.test.y  
x<-table(pred.lda.cm,Truth) # Creating a confusion matrix  
x

## Truth  
## pred.lda.cm no yes  
## no 3003 288  
## yes 104 101

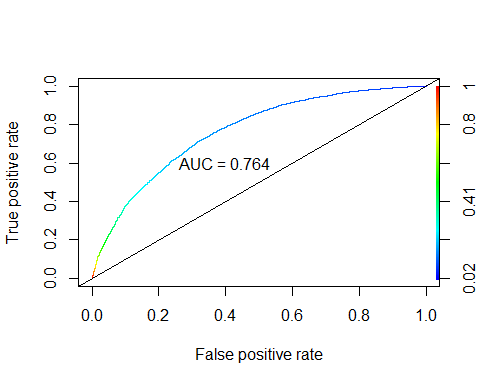
# Truth  
#pred.lda.cm no yes  
# no 3003 288  
# yes 104 101  
#Missclassification Error  
ME<-(x[2,1]+x[1,2])/3496 #change denom to N  
ME

## [1] 0.1121281

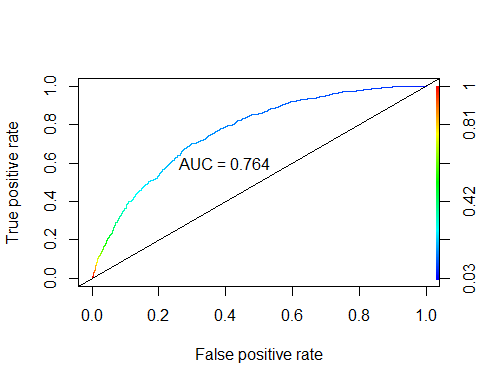
#0.1121281  
#Calculating overall accuracy  
1-ME

## [1] 0.8878719

#0.8878719  
  
#Without 2008  
#Training Set  
lda.train.x.2008 <- train2008[,c(1,6,10,12,13,14,15)]  
lda.train.y.2008 <- train2008$y  
fit.lda.2008 <- lda(lda.train.y.2008 ~ ., data = lda.train.x.2008)  
pred.lda.2008 <- predict(fit.lda.2008, newdata = lda.train.x.2008)  
preds.2008 <- pred.lda.2008$posterior  
preds.2008 <- as.data.frame(preds.2008)  
  
# ROC for TRAINING data  
pred.2008 <- prediction(preds.2008[,2],lda.train.y.2008)  
roc.perf.2008 = performance(pred.2008, measure = "tpr", x.measure = "fpr")  
auc.train.2008 <- performance(pred.2008, measure = "auc")  
auc.train.2008 <- auc.train.2008@y.values  
plot(roc.perf.2008, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train.2008[[1]],3), sep = ""))



# AUC=0.764. Not surprising. Maybe better if we normalize and then weed out some variables  
  
#LDA for TEST data  
lda.test.x.2008 <- test2008[,c(1,6,10,12,13,14,15)]  
lda.test.y.2008 <- test2008$y  
pred.lda1.2008 <- predict(fit.lda.2008, newdata = lda.test.x.2008)  
preds1.2008 <- pred.lda1.2008$posterior  
preds1.2008 <- as.data.frame(preds1.2008)  
pred1.2008 <- prediction(preds1.2008[,2],lda.test.y.2008)  
roc.perf.2008 = performance(pred1.2008, measure = "tpr", x.measure = "fpr")  
auc.train.2008 <- performance(pred1.2008, measure = "auc")  
auc.train.2008 <- auc.train.2008@y.values  
plot(roc.perf.2008, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train.2008[[1]],3), sep = ""))



#AUC=0.764  
  
pred.lda.cm.2008<-predict(fit.lda.2008,newdata=lda.test.x.2008)$class #Predictions can come in many forms, the class form provides the categorical level of your response.  
Truth<-lda.test.y.2008  
x<-table(pred.lda.cm.2008,Truth) # Creating a confusion matrix  
x

## Truth  
## pred.lda.cm.2008 no yes  
## no 2594 626  
## yes 123 153

# Truth  
#pred.lda.cm.2008 no yes  
# no 2594 626  
# yes 123 153  
  
#Missclassification Error  
ME<-(x[2,1]+x[1,2])/3496 #change denom to N  
ME

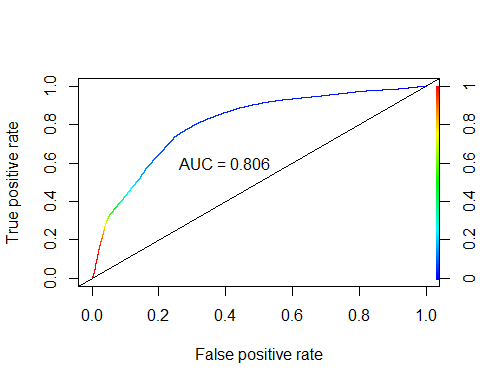
## [1] 0.2142449

#0.2142449  
#Calculating overall accuracy  
1-ME

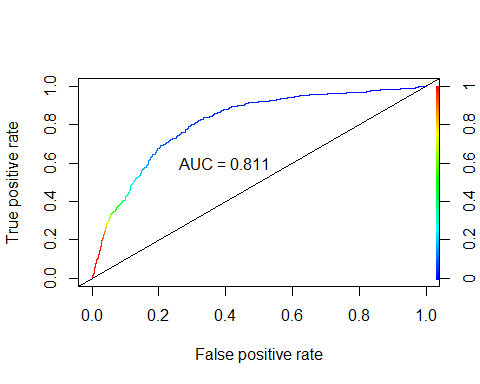
## [1] 0.7857551

#0.7857551

# With all years  
fit.qda <- qda(lda.train.y ~ ., data = lda.train.x)  
pred.qda <- predict(fit.qda, newdata = lda.train.x)  
preds <- pred.qda$posterior  
preds <- as.data.frame(preds)  
# ROC for TRAINING data  
pred <- prediction(preds[,2],lda.train.y)  
roc.perf = performance(pred, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC: 0.806  
  
# QDA for TEST data  
pred.qda1 <- predict(fit.qda, newdata = lda.test.x)  
preds1 <- pred.qda1$posterior  
preds1 <- as.data.frame(preds1)  
pred1 <- prediction(preds1[,2],lda.test.y)  
roc.perf = performance(pred1, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred1, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC = 0.811  
  
pred.lda.cm<-predict(fit.qda,newdata=lda.test.x)$class #Predictions can come in many forms, the class form provides the categorical level of your response.  
Truth<-lda.test.y  
x<-table(pred.lda.cm,Truth) # Creating a confusion matrix  
x

## Truth  
## pred.lda.cm no yes  
## no 2881 250  
## yes 226 139

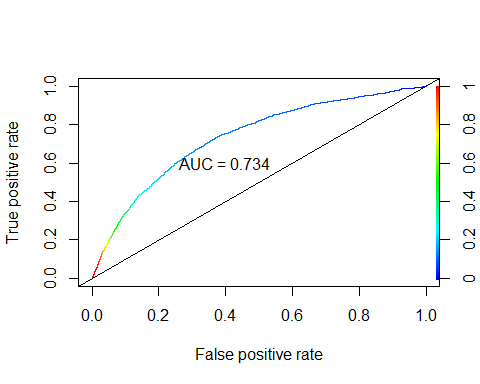
# Truth  
#pred.lda.cm no yes  
# no 2881 250  
# yes 226 139  
  
#Missclassification Error  
ME<-(x[2,1]+x[1,2])/3496 #change denom to N  
ME

## [1] 0.1361556

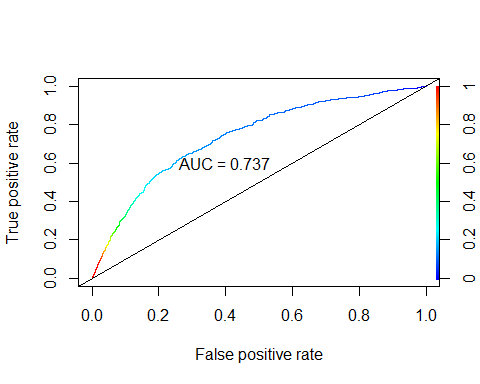
#0.1361556  
#Calculating overall accuracy  
1-ME

## [1] 0.8638444

#0.8638444  
  
#Excluding 2008  
fit.qda.2008 <- qda(lda.train.y.2008 ~ ., data = lda.train.x.2008)  
pred.qda <- predict(fit.qda.2008, newdata = lda.train.x.2008)  
preds <- pred.qda$posterior  
preds <- as.data.frame(preds)  
# ROC for TRAINING data  
pred <- prediction(preds[,2],lda.train.y.2008)  
roc.perf = performance(pred, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC: 0.734  
  
# QDA for TEST data  
pred.qda1.2008 <- predict(fit.qda.2008, newdata = lda.test.x.2008)  
preds1 <- pred.qda1.2008$posterior  
preds1 <- as.data.frame(preds1)  
pred1 <- prediction(preds1[,2],lda.test.y.2008)  
roc.perf = performance(pred1, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred1, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC = 0.737  
  
  
pred.lda.cm.2008<-predict(fit.qda.2008,newdata=lda.test.x.2008)$class #Predictions can come in many forms, the class form provides the categorical level of your response.  
Truth<-lda.test.y.2008  
x<-table(pred.lda.cm.2008,Truth) # Creating a confusion matrix  
x

## Truth  
## pred.lda.cm.2008 no yes  
## no 2509 572  
## yes 208 207

# Truth  
#pred.lda.cm.2008 no yes  
# no 2509 572  
# yes 208 207  
  
#Missclassification Error  
ME<-(x[2,1]+x[1,2])/3496 #change denom to N  
ME

## [1] 0.2231121

#0.2231121  
  
#Calculating overall accuracy  
1-ME

## [1] 0.7768879

#0.7768879

# try some interactions  
#For Year excluding 2008  
#year&day and year&month  
model.complex<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year+year:month+year:day, data=train2008,family = binomial(link="logit"))  
step(Model\_Full\_2008,  
 scope = list(upper=model.complex),  
 direction="forward",  
 test="Chisq",  
 data=train2008)

## Start: AIC=10333.99  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year  
##   
## Df Deviance AIC LRT Pr(>Chi)   
## + month:year 10 9960.5 10068 285.527 < 2e-16 \*\*\*  
## + day:year 1 10242.7 10333 3.269 0.07062 .   
## <none> 10246.0 10334   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=10068.46  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year + month:year  
##   
## Df Deviance AIC LRT Pr(>Chi)   
## + day:year 1 9931.5 10042 28.958 7.398e-08 \*\*\*  
## <none> 9960.5 10068   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=10041.5  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year + month:year + day:year

##   
## Call: glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year +   
## month:year + day:year, family = binomial(link = "logit"),   
## data = train2008)  
##   
## Coefficients:  
## (Intercept) age jobblue-collar jobentrepreneur   
## -7.025e+00 1.454e-03 -2.326e-01 -3.153e-01   
## jobhousemaid jobmanagement jobretired jobself-employed   
## -3.820e-01 3.167e-02 -1.272e-03 -1.483e-01   
## jobservices jobstudent jobtechnician jobunemployed   
## -6.238e-02 1.553e-01 -2.133e-03 -1.372e-01   
## jobunknown maritalmarried maritalsingle educationsecondary   
## -4.772e-02 -1.670e-01 3.126e-03 2.834e-01   
## educationtertiary educationunknown defaultyes balance   
## 4.179e-01 2.520e-01 1.392e-01 -6.724e-07   
## housingyes loanyes contacttelephone contactunknown   
## -5.766e-01 -2.380e-01 -4.927e-01 -1.179e+00   
## day monthfeb monthmar monthapr   
## 4.443e-02 3.144e+00 5.020e+00 3.066e+00   
## monthmay monthjun monthjul monthaug   
## 2.817e+00 4.513e+00 4.360e+00 3.968e+00   
## monthsep monthoct monthnov monthdec   
## 4.376e+00 4.282e+00 4.799e+00 4.329e+00   
## duration campaign pdays previous   
## 3.966e-03 -8.275e-02 4.290e-04 2.090e-02   
## poutcomeother poutcomesuccess poutcomeunknown year2010   
## 1.566e-01 1.793e+00 6.315e-01 5.141e+00   
## monthfeb:year2010 monthmar:year2010 monthapr:year2010 monthmay:year2010   
## -3.092e+00 -4.627e+00 -2.734e+00 -2.609e+00   
## monthjun:year2010 monthjul:year2010 monthaug:year2010 monthsep:year2010   
## -4.355e+00 -4.104e+00 -3.759e+00 -4.321e+00   
## monthoct:year2010 monthnov:year2010 monthdec:year2010 day:year2010   
## -4.591e+00 -5.036e+00 NA -4.060e-02   
##   
## Degrees of Freedom: 13984 Total (i.e. Null); 13930 Residual  
## Null Deviance: 14820   
## Residual Deviance: 9932 AIC: 10040

#Degrees of Freedom: 13984 Total (i.e. Null); 13930 Residual  
#Null Deviance: 14820   
#Residual Deviance: 9932   
#AIC: 10040  
  
#age&balance  
model.complex<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year+age:balance, data=train2008,family = binomial(link="logit"))  
step(Model\_Full\_2008,  
 scope = list(upper=model.complex),  
 direction="forward",  
 test="Chisq",  
 data=train2008)

## Start: AIC=10333.99  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year  
##   
## Df Deviance AIC LRT Pr(>Chi)  
## <none> 10246 10334   
## + age:balance 1 10244 10334 1.7787 0.1823

##   
## Call: glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train2008)  
##   
## Coefficients:  
## (Intercept) age jobblue-collar jobentrepreneur   
## -4.917e+00 3.715e-03 -2.899e-01 -4.291e-01   
## jobhousemaid jobmanagement jobretired jobself-employed   
## -3.580e-01 -1.026e-03 2.264e-02 -1.791e-01   
## jobservices jobstudent jobtechnician jobunemployed   
## -1.231e-01 1.567e-01 -1.579e-02 -2.343e-01   
## jobunknown maritalmarried maritalsingle educationsecondary   
## -6.376e-02 -9.675e-02 6.028e-02 2.818e-01   
## educationtertiary educationunknown defaultyes balance   
## 4.557e-01 2.832e-01 -2.834e-02 2.076e-06   
## housingyes loanyes contacttelephone contactunknown   
## -6.775e-01 -2.774e-01 -4.937e-01 -1.411e+00   
## day monthfeb monthmar monthapr   
## 2.585e-02 1.386e+00 2.734e+00 1.512e+00   
## monthmay monthjun monthjul monthaug   
## 1.207e+00 2.394e+00 2.123e+00 2.116e+00   
## monthsep monthoct monthnov monthdec   
## 2.133e+00 2.135e+00 2.665e+00 2.517e+00   
## duration campaign pdays previous   
## 3.733e-03 -8.143e-02 4.407e-05 1.168e-02   
## poutcomeother poutcomesuccess poutcomeunknown year2010   
## 1.046e-01 1.867e+00 5.075e-01 1.085e+00   
##   
## Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
## Null Deviance: 14820   
## Residual Deviance: 10250 AIC: 10330

#Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
#Null Deviance: 14820   
#Residual Deviance: 10250   
#AIC: 10330  
  
#job&balance  
model.complex<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year+job:balance, data=train2008,family = binomial(link="logit"))  
step(Model\_Full\_2008,  
 scope = list(upper=model.complex),  
 direction="forward",  
 test="Chisq",  
 data=train2008)

## Start: AIC=10333.99  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year  
##   
## Df Deviance AIC LRT Pr(>Chi)  
## <none> 10246 10334   
## + job:balance 11 10236 10346 9.5157 0.5744

##   
## Call: glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train2008)  
##   
## Coefficients:  
## (Intercept) age jobblue-collar jobentrepreneur   
## -4.917e+00 3.715e-03 -2.899e-01 -4.291e-01   
## jobhousemaid jobmanagement jobretired jobself-employed   
## -3.580e-01 -1.026e-03 2.264e-02 -1.791e-01   
## jobservices jobstudent jobtechnician jobunemployed   
## -1.231e-01 1.567e-01 -1.579e-02 -2.343e-01   
## jobunknown maritalmarried maritalsingle educationsecondary   
## -6.376e-02 -9.675e-02 6.028e-02 2.818e-01   
## educationtertiary educationunknown defaultyes balance   
## 4.557e-01 2.832e-01 -2.834e-02 2.076e-06   
## housingyes loanyes contacttelephone contactunknown   
## -6.775e-01 -2.774e-01 -4.937e-01 -1.411e+00   
## day monthfeb monthmar monthapr   
## 2.585e-02 1.386e+00 2.734e+00 1.512e+00   
## monthmay monthjun monthjul monthaug   
## 1.207e+00 2.394e+00 2.123e+00 2.116e+00   
## monthsep monthoct monthnov monthdec   
## 2.133e+00 2.135e+00 2.665e+00 2.517e+00   
## duration campaign pdays previous   
## 3.733e-03 -8.143e-02 4.407e-05 1.168e-02   
## poutcomeother poutcomesuccess poutcomeunknown year2010   
## 1.046e-01 1.867e+00 5.075e-01 1.085e+00   
##   
## Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
## Null Deviance: 14820   
## Residual Deviance: 10250 AIC: 10330

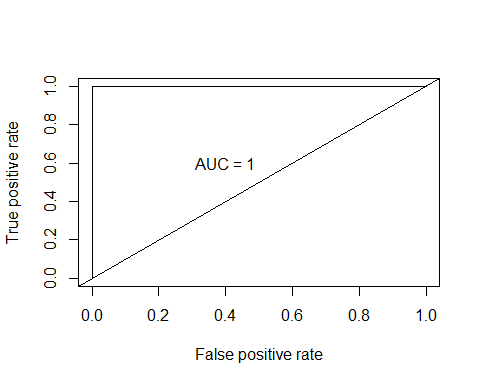
#Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
#Null Deviance: 14820   
#Residual Deviance: 10250 AIC: 10330  
  
#education&balance  
model.complex<-glm(as.factor(y) ~ age+job+marital+education+default+balance+housing+loan+contact+day+month+duration+campaign+pdays+previous+poutcome+year+education:balance, data=train2008,family = binomial(link="logit"))  
step(Model\_Full\_2008,  
 scope = list(upper=model.complex),  
 direction="forward",  
 test="Chisq",  
 data=train2008)

## Start: AIC=10333.99  
## as.factor(y) ~ age + job + marital + education + default + balance +   
## housing + loan + contact + day + month + duration + campaign +   
## pdays + previous + poutcome + year  
##   
## Df Deviance AIC LRT Pr(>Chi)  
## <none> 10246 10334   
## + education:balance 3 10246 10340 0.51645 0.9153

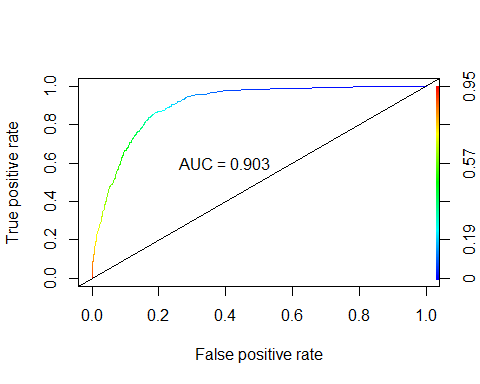
##   
## Call: glm(formula = as.factor(y) ~ age + job + marital + education +   
## default + balance + housing + loan + contact + day + month +   
## duration + campaign + pdays + previous + poutcome + year,   
## family = binomial(link = "logit"), data = train2008)  
##   
## Coefficients:  
## (Intercept) age jobblue-collar jobentrepreneur   
## -4.917e+00 3.715e-03 -2.899e-01 -4.291e-01   
## jobhousemaid jobmanagement jobretired jobself-employed   
## -3.580e-01 -1.026e-03 2.264e-02 -1.791e-01   
## jobservices jobstudent jobtechnician jobunemployed   
## -1.231e-01 1.567e-01 -1.579e-02 -2.343e-01   
## jobunknown maritalmarried maritalsingle educationsecondary   
## -6.376e-02 -9.675e-02 6.028e-02 2.818e-01   
## educationtertiary educationunknown defaultyes balance   
## 4.557e-01 2.832e-01 -2.834e-02 2.076e-06   
## housingyes loanyes contacttelephone contactunknown   
## -6.775e-01 -2.774e-01 -4.937e-01 -1.411e+00   
## day monthfeb monthmar monthapr   
## 2.585e-02 1.386e+00 2.734e+00 1.512e+00   
## monthmay monthjun monthjul monthaug   
## 1.207e+00 2.394e+00 2.123e+00 2.116e+00   
## monthsep monthoct monthnov monthdec   
## 2.133e+00 2.135e+00 2.665e+00 2.517e+00   
## duration campaign pdays previous   
## 3.733e-03 -8.143e-02 4.407e-05 1.168e-02   
## poutcomeother poutcomesuccess poutcomeunknown year2010   
## 1.046e-01 1.867e+00 5.075e-01 1.085e+00   
##   
## Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
## Null Deviance: 14820   
## Residual Deviance: 10250 AIC: 10330

#Degrees of Freedom: 13984 Total (i.e. Null); 13941 Residual  
#Null Deviance: 14820   
#Residual Deviance: 10250 AIC: 10330

# Random Forest (training data)  
# Remove id variable as it's just for reference  
dat.train.rf <- train2008[,-18]  
train.rf<-randomForest(as.factor(y)~.,data=dat.train.rf,mtry=4,ntree=500,importance=T)  
fit.pred<-predict(train.rf,newdata=dat.train.rf,type="prob")  
pred <- prediction(fit.pred[,2], dat.train.rf$y)  
roc.perf = performance(pred, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



#AUC=1  
  
# Random Forest (test data)  
#Predict test set  
dat.val1.rf <- test2008[,-18]  
pred.val1<-predict(train.rf,newdata=dat.val1.rf,type="prob")  
pred <- prediction(pred.val1[,2], dat.val1.rf$y)  
roc.perf = performance(pred, measure = "tpr", x.measure = "fpr")  
auc.train <- performance(pred, measure = "auc")  
auc.train <- auc.train@y.values  
plot(roc.perf, colorize=TRUE)  
abline(a=0, b= 1)  
text(x = .40, y = .6,paste("AUC = ", round(auc.train[[1]],3), sep = ""))



# AUC = 0.904