Untitled

Americo

29 aprile 2016

# Caricamento package

require(gains)  
require(PRROC)  
require(broom)

## Warning: package 'broom' was built under R version 3.1.3

require(ResourceSelection)  
require(MKmisc)

## Warning: package 'MKmisc' was built under R version 3.1.3

require(perturb)  
require(caret)

## Warning: package 'caret' was built under R version 3.1.3

## Warning: package 'ggplot2' was built under R version 3.1.3

require(DAAG)  
library(pROC)

## Warning: package 'pROC' was built under R version 3.1.3

library(ROCR)

## Warning: package 'ROCR' was built under R version 3.1.3

## Warning: package 'gplots' was built under R version 3.1.3

require(MASS)  
library(devtools)

## Warning: package 'devtools' was built under R version 3.1.3

require(GGally)  
library(woe)  
require(ggplot2)  
require(dplyr)  
require(tidyr)  
select <- dplyr::select

# Set up cartelle e creazione dataset

getwd()

## [1] "/Users/Americo/Documents/Education/Unitelma/tesi/data\_analysis/analysis"

setwd("/Users/Americo/Documents/Education/Unitelma/tesi/data\_analysis/dataset")  
bank0\_df <- read.csv(file = "bank\_full.csv", sep = ";")  
bank0 <- tbl\_df(read.csv(file = "bank\_full.csv", sep = ";"))  
bank0\_sm <- read.csv(file = "bank.csv", sep = ";")

# Informazioni sul dataset

## Input variables

### 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")

It should be stressed that due to internal competition and current financial crisis, there are huge pressures for European banks to increase a financial asset. To solve this issue, one adopted strategy is offer attractive long-term deposit applications with good interest rates, in particular by using directed marketing campaigns. Also, the same drivers are pressing for a reduction in costs and time. Thus, there is a need for an improvement in efficiency: lesser contacts should be done, but an approximately number of successes (clients subscribing the deposit) should be kept.

## Missing Attribute Values

None

# Esplorazione del dataset

## Esplorazione dimensioni

str(bank0)

## Classes 'tbl\_df', 'tbl' and 'data.frame': 45211 obs. of 17 variables:  
## $ age : int 58 44 33 47 33 35 28 42 58 43 ...  
## $ job : Factor w/ 12 levels "admin.","blue-collar",..: 5 10 3 2 12 5 5 3 6 10 ...  
## $ marital : Factor w/ 3 levels "divorced","married",..: 2 3 2 2 3 2 3 1 2 3 ...  
## $ education: Factor w/ 4 levels "primary","secondary",..: 3 2 2 4 4 3 3 3 1 2 ...  
## $ default : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 2 1 1 ...  
## $ balance : int 2143 29 2 1506 1 231 447 2 121 593 ...  
## $ housing : Factor w/ 2 levels "no","yes": 2 2 2 2 1 2 2 2 2 2 ...  
## $ loan : Factor w/ 2 levels "no","yes": 1 1 2 1 1 1 2 1 1 1 ...  
## $ contact : Factor w/ 3 levels "cellular","telephone",..: 3 3 3 3 3 3 3 3 3 3 ...  
## $ day : int 5 5 5 5 5 5 5 5 5 5 ...  
## $ month : Factor w/ 12 levels "apr","aug","dec",..: 9 9 9 9 9 9 9 9 9 9 ...  
## $ 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 : Factor w/ 4 levels "failure","other",..: 4 4 4 4 4 4 4 4 4 4 ...  
## $ y : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...

dim(bank0)

## [1] 45211 17

Non ci sono variabili numeriche, solo interi e fattori. Da capire se servirà qualche trasformazione #Esplorazione tabellare distribuzioni univariate

summary(bank0)

## age job marital education   
## Min. :18.00 blue-collar:9732 divorced: 5207 primary : 6851   
## 1st Qu.:33.00 management :9458 married :27214 secondary:23202   
## Median :39.00 technician :7597 single :12790 tertiary :13301   
## Mean :40.94 admin. :5171 unknown : 1857   
## 3rd Qu.:48.00 services :4154   
## Max. :95.00 retired :2264   
## (Other) :6835   
## default balance housing loan contact   
## no :44396 Min. : -8019 no :20081 no :37967 cellular :29285   
## yes: 815 1st Qu.: 72 yes:25130 yes: 7244 telephone: 2906   
## Median : 448 unknown :13020   
## Mean : 1362   
## 3rd Qu.: 1428   
## Max. :102127   
##   
## day month duration campaign   
## Min. : 1.00 may :13766 Min. : 0.0 Min. : 1.000   
## 1st Qu.: 8.00 jul : 6895 1st Qu.: 103.0 1st Qu.: 1.000   
## Median :16.00 aug : 6247 Median : 180.0 Median : 2.000   
## Mean :15.81 jun : 5341 Mean : 258.2 Mean : 2.764   
## 3rd Qu.:21.00 nov : 3970 3rd Qu.: 319.0 3rd Qu.: 3.000   
## Max. :31.00 apr : 2932 Max. :4918.0 Max. :63.000   
## (Other): 6060   
## pdays previous poutcome y   
## Min. : -1.0 Min. : 0.0000 failure: 4901 no :39922   
## 1st Qu.: -1.0 1st Qu.: 0.0000 other : 1840 yes: 5289   
## Median : -1.0 Median : 0.0000 success: 1511   
## Mean : 40.2 Mean : 0.5803 unknown:36959   
## 3rd Qu.: -1.0 3rd Qu.: 0.0000   
## Max. :871.0 Max. :275.0000   
##

## Age

g\_age <- ggplot(bank0, aes(x = factor(age))) + geom\_bar(col = "white")  
g\_age

g\_age\_y <- ggplot(bank0, aes(x = factor(age), fill = y)) +   
 geom\_bar(col = "white", position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_age\_y

Vediamo che dai 30 ai 59 anni l'incidenza di sottoscrizioni è pari a quella complessiva; c'è anche da dire che dopo i 60 non ci sono numerosità grandi:

t\_age\_y <- bank0 %>%  
 group\_by (factor(age)) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_age\_y

## Source: local data frame [77 x 3]  
##   
## factor(age) n perc\_y  
## 1 18 12 0.5833333  
## 2 19 35 0.3142857  
## 3 20 50 0.3000000  
## 4 21 79 0.2784810  
## 5 22 129 0.3100775  
## 6 23 202 0.2178218  
## 7 24 302 0.2251656  
## 8 25 527 0.2144213  
## 9 26 805 0.1664596  
## 10 27 909 0.1551155  
## .. ... ... ...

Una idea potrebbe essere quella di fare un binning 18-30, 31-59, 60 o più. Certo che la variabile sembra predittiva.

bank0$age\_num <- as.numeric(bank0$age)  
bank0$age\_class <- cut(bank0$age\_num, breaks = c(min(bank0$age\_num)-1, 30, 59, max(bank0$age\_num)))  
bank\_age\_na <- bank0[is.na(bank0$age\_class),]  
  
age\_class\_woe <- bank0 %>%  
 select(age\_class, y) %>%  
 group\_by(age\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
age\_class\_woe$woe <- log(age\_class\_woe$perc\_no / age\_class\_woe$perc\_y)  
age\_class\_IV <- sum((age\_class\_woe$perc\_no - age\_class\_woe$perc\_y) \* age\_class\_woe$woe)

bank0 %>%  
 select(age\_class, y) %>%  
 group\_by(age\_class) %>%  
 summarise(n = n(), perc\_y = mean(y =="yes"))

## Source: local data frame [3 x 3]  
##   
## age\_class n perc\_y  
## 1 (17,30] 7030 0.16287340  
## 2 (30,59] 36397 0.09737066  
## 3 (59,95] 1784 0.33632287

## Job

g\_job <- ggplot(bank0, aes(x = job)) +  
 geom\_bar()  
g\_job

g\_job\_y <- ggplot(bank0, aes(x = job, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_job\_y

Non molto predittivo, c'è l'età che fa da counfounding e spiega la stessa variabilità di studenti e pensionati.

t\_job\_y <- bank0 %>%  
 group\_by (job) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_job\_y

## Source: local data frame [12 x 3]  
##   
## job n perc\_y  
## 1 admin. 5171 0.12202669  
## 2 blue-collar 9732 0.07274969  
## 3 entrepreneur 1487 0.08271688  
## 4 housemaid 1240 0.08790323  
## 5 management 9458 0.13755551  
## 6 retired 2264 0.22791519  
## 7 self-employed 1579 0.11842939  
## 8 services 4154 0.08883004  
## 9 student 938 0.28678038  
## 10 technician 7597 0.11056996  
## 11 unemployed 1303 0.15502686  
## 12 unknown 288 0.11805556

Non mi spiego perché i disoccupati tendano a fare più depositi. Forse in questa categoria ci sono giovani a cui il deposito viene regalato dai genitori.

job\_woe <- bank0 %>%  
 select(job, y) %>%  
 group\_by(job) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
job\_woe$woe <- log(job\_woe$perc\_no / job\_woe$perc\_y)  
job\_IV <- sum((job\_woe$perc\_no - job\_woe$perc\_y) \* job\_woe$woe)  
  
#woe\_func <- function(x) {  
 #bank0 %>%  
 #select\_(x, "y") %>%  
 #group\_by\_(x) %>%  
 #summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 #mutate(perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 #select(starts\_with("perc"))  
  
#paste(c("x","\_","IV"), collapse="") <- sum((perc\_no - perc\_y) \* log(perc\_no / perc\_y))   
#}  
#woe\_func("job")#la NSE è troppo complessa

## Marital

g\_marital <- ggplot(bank0, aes(x = marital)) +  
 geom\_bar()  
g\_marital

g\_marital\_y <- ggplot(bank0, aes(x = marital, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_marital\_y

Non me lo aspettavo. Pochissima predittività della variabile

t\_marital\_y <- bank0 %>%  
 group\_by (marital) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_marital\_y

## Source: local data frame [3 x 3]  
##   
## marital n perc\_y  
## 1 divorced 5207 0.1194546  
## 2 married 27214 0.1012347  
## 3 single 12790 0.1494918

I single tendono a farne un po' di più. Anche qui, confounding dell'età giovanile?

marital\_woe <- bank0 %>%  
 select(marital, y) %>%  
 group\_by(marital) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
marital\_woe$woe <- log(marital\_woe$perc\_no / marital\_woe$perc\_y)  
marital\_IV <- sum((marital\_woe$perc\_no - marital\_woe$perc\_y) \* marital\_woe$woe)

## Education

g\_education <- ggplot(bank0, aes(x = education)) +  
 geom\_bar()  
g\_education

g\_education\_y <- ggplot(bank0, aes(x = education, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_education\_y

Sembra che a maggior livello culturale segua maggiore propensione a sottoscrivere depositi. Da capire se qui gli studenti fanno confounding, anche se ci credo meno. Penso (sarà da verificare con qualche analisi trivariata) che anche tra i trentacinquenni ci siano molti laureati.

t\_education\_y <- bank0 %>%  
 group\_by (education) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_education\_y

## Source: local data frame [4 x 3]  
##   
## education n perc\_y  
## 1 primary 6851 0.08626478  
## 2 secondary 23202 0.10559435  
## 3 tertiary 13301 0.15006390  
## 4 unknown 1857 0.13570275

education\_woe <- bank0 %>%  
 select(education, y) %>%  
 group\_by(education) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
education\_woe$woe <- log(education\_woe$perc\_no / education\_woe$perc\_y)  
education\_IV <- sum((education\_woe$perc\_no - education\_woe$perc\_y) \* education\_woe$woe)

## Default

g\_default <- ggplot(bank0, aes(x = default)) +  
 geom\_bar()  
g\_default

g\_default\_y <- ggplot(bank0, aes(x = default, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_default\_y

Pochissimi in default, i no si comportanto come il campione, ma d'altronde sono pressoché tutto il campione. Certo il sì è un po' predittivo, ma poca numerosità.

t\_default\_y <- bank0 %>%  
 group\_by (default) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_default\_y

## Source: local data frame [2 x 3]  
##   
## default n perc\_y  
## 1 no 44396 0.11796108  
## 2 yes 815 0.06380368

default\_woe <- bank0 %>%  
 select(default, y) %>%  
 group\_by(default) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
default\_woe$woe <- log(default\_woe$perc\_no / default\_woe$perc\_y)  
default\_IV <- sum((default\_woe$perc\_no - default\_woe$perc\_y) \* default\_woe$woe)

## Balance

g\_balance <- ggplot(bank0, aes(x = 1, y = balance)) +   
 geom\_boxplot()  
g\_balance

g\_balance\_y <- ggplot(bank0, aes(x = y, y = balance)) +   
 geom\_boxplot()  
g\_balance\_y

è una distribuzione con un range molto ampio ma un range interquartile molto ristretto, sia complessivamente che per i due gruppi. Provo a capire se posso analizzare i dati fino al 95 percentile

quantile(bank0$balance, c(seq(0.1, 1, 0.05)))

## 10% 15% 20% 25% 30% 35% 40% 45% 50% 55%   
## 0 0 22 72 131 198 272 352 448 563   
## 60% 65% 70% 75% 80% 85% 90% 95% 100%   
## 701 883 1126 1428 1859 2539 3574 5768 102127

balance\_p\_95 <- quantile(bank0$balance, 0.95)

bank0\_p95 <- bank0 %>%  
 filter(balance <= balance\_p\_95)  
g\_balance\_p95 <- ggplot(bank0\_p95, aes(x = 1, y = balance)) +   
 geom\_boxplot()  
g\_balance\_p95

g\_balance\_y\_p95 <- ggplot(bank0\_p95, aes(x = y, y = balance)) +   
 geom\_boxplot()  
g\_balance\_y\_p95

Ora comincia a essere evidente che chi sottoscrive depositi tende a avere saldo sul conto più alto; vale per la mediana del gruppo sì, ma anche per il terzo quartile. Quindi per saldi che crescono si tende a sottoscrivere più depositi, il che ha senso. più soldi hai, più ne puoi depositare.

bank0$balance <- as.numeric(bank0$balance)  
  
balance\_quantile <- quantile(bank0$balance, probs = seq(0.1,1,0.1))  
  
bank0$balance\_class <- cut(bank0$balance, breaks = c(min(bank0$balance)-1, -1, balance\_quantile), right = TRUE, labels = c("negative", "0", "(0, 22]", "(22,131]", "(131,272]", "(272,448]", "(448,701]", "(701,1126]", "(1126,1859]", "(1859,3574]", "(3574,102127]"))  
summary(bank0$balance\_class)

## negative 0 (0, 22] (22,131] (131,272]   
## 3766 3514 1773 4544 4516   
## (272,448] (448,701] (701,1126] (1126,1859] (1859,3574]   
## 4495 4522 4526 4513 4521   
## (3574,102127]   
## 4521

bank\_balance\_na <- bank0[is.na(bank0$balance\_class),] #ok  
  
  
t\_balance\_class\_y <- bank0 %>%  
 group\_by (balance\_class) %>%  
 summarise(n(), mean(y == "yes"))

Esatto. al crescere del saldo cresce la quota di coloro che sottoscrivono.

balance\_class\_woe <- bank0 %>%  
 select(balance\_class, y) %>%  
 group\_by(balance\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
balance\_class\_woe$woe <- log(balance\_class\_woe$perc\_no / balance\_class\_woe$perc\_y)  
balance\_class\_IV <- sum((balance\_class\_woe$perc\_no - balance\_class\_woe$perc\_y) \* balance\_class\_woe$woe)

## Housing

g\_housing <- ggplot(bank0, aes(x = housing)) +  
 geom\_bar()  
g\_housing

g\_housing\_y <- ggplot(bank0, aes(x = housing, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_housing\_y

Una buona predittività, peraltro due gruppi di quasi uguale dimensione. Chi ha già un mutuo per la casa non sottoscrive depositi a lungo termine, ovvio.

t\_housing\_y <- bank0 %>%  
 group\_by (housing) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_housing\_y

## Source: local data frame [2 x 3]  
##   
## housing n perc\_y  
## 1 no 20081 0.1670236  
## 2 yes 25130 0.0769996

housing\_woe <- bank0 %>%  
 select(housing, y) %>%  
 group\_by(housing) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
housing\_woe$woe <- log(housing\_woe$perc\_no / housing\_woe$perc\_y)  
housing\_IV <- sum((housing\_woe$perc\_no - housing\_woe$perc\_y) \* housing\_woe$woe)

## Loan

g\_loan <- ggplot(bank0, aes(x = loan)) +  
 geom\_bar()  
g\_loan

g\_loan\_y <- ggplot(bank0, aes(x = loan, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_loan\_y

Una buona predittività, ma minore del mutuo. Chi ha un prestito personale tende a sottoscrivere meno depositi, certo che è un gruppo più ristretto, e infatti chi non li sottoscrive si comporta pressoché come il campione (un pelino più sottoscrivente).

t\_loan\_y <- bank0 %>%  
 group\_by (loan) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_loan\_y

## Source: local data frame [2 x 3]  
##   
## loan n perc\_y  
## 1 no 37967 0.12655727  
## 2 yes 7244 0.06681391

I volumi sono comunque significativi.

loan\_woe <- bank0 %>%  
 select(loan, y) %>%  
 group\_by(loan) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
loan\_woe$woe <- log(loan\_woe$perc\_no / loan\_woe$perc\_y)  
loan\_IV <- sum((loan\_woe$perc\_no - loan\_woe$perc\_y) \* loan\_woe$woe)

# Variabili di Contatto

## Contact

g\_contact <- ggplot(bank0, aes(x = contact)) +  
 geom\_bar()  
g\_contact

g\_contact\_y <- ggplot(bank0, aes(x = contact, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_contact\_y

Però, se la tipologia di contatto è sconosciuta non converte quasi nessuno, con cellulare più della media, e anche con telefono. buono a sapersi, ma che vorrà dire?

t\_contact\_y <- bank0 %>%  
 group\_by (contact) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_contact\_y

## Source: local data frame [3 x 3]  
##   
## contact n perc\_y  
## 1 cellular 29285 0.14918900  
## 2 telephone 2906 0.13420509  
## 3 unknown 13020 0.04070661

contact\_woe <- bank0 %>%  
 select(contact, y) %>%  
 group\_by(contact) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
contact\_woe$woe <- log(contact\_woe$perc\_no / contact\_woe$perc\_y)  
contact\_IV <- sum((contact\_woe$perc\_no - contact\_woe$perc\_y) \* contact\_woe$woe)

Finora la variabile più predittiva!

## Day

La trasformo in fattore, 31 livelli su 45000 osservazioni: ha senso

g\_day <- ggplot(bank0, aes(x = factor(day))) +  
 geom\_bar()  
g\_day

g\_day\_y <- ggplot(bank0, aes(x = factor(day), fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_day\_y

Distribuzione trimodale, abbastanza strana.

t\_day\_y <- bank0 %>%  
 group\_by (day) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_day\_y

## Source: local data frame [31 x 3]  
##   
## day n perc\_y  
## 1 1 322 0.27950311  
## 2 2 1293 0.14075793  
## 3 3 1079 0.16496756  
## 4 4 1445 0.15916955  
## 5 5 1910 0.11256545  
## 6 6 1932 0.09368530  
## 7 7 1817 0.08640616  
## 8 8 1842 0.10912052  
## 9 9 1561 0.11467008  
## 10 10 524 0.23091603  
## .. ... ... ...

I due giorni di maggior conversione sono quelli con i minori contatti. Credo bassa predittività.

day\_woe <- bank0 %>%  
 select(day, y) %>%  
 group\_by(day) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
day\_woe$woe <- log(day\_woe$perc\_no / day\_woe$perc\_y)  
day\_IV <- sum((day\_woe$perc\_no - day\_woe$perc\_y) \* day\_woe$woe)

Bassa ma non bassissima.

## Month

levels(bank0$month)

## [1] "apr" "aug" "dec" "feb" "jan" "jul" "jun" "mar" "may" "nov" "oct"  
## [12] "sep"

class(bank0$month)

## [1] "factor"

bank0$month <- factor(bank0$month, levels = c("jan","feb","mar", "apr","may","jun", "jul","aug", "sep", "oct", "nov","dec"))  
g\_month <- ggplot(bank0, aes(x = month)) +  
 geom\_bar()  
g\_month

g\_month\_y <- ggplot(bank0, aes(x = month, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_month\_y

Distribuzione molto concentrata in estate, con grande conversion nei mesi primaverili e autunnali. Perché? Pochissimi contatti, non più di centinaia. Il mese con più contatti ha la conversion più bassa. Potrebbe dipendendere dal fatto che minori quanttà di telefonate permettono maggiore qualità di vendita (vedi duration)? Forse, ma anche la stagionalità ha la sua parte. Chi pensa al futuro all'avvicinarsi dell'estate?

t\_month\_y <- bank0 %>%  
 group\_by (month) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes")) %>%  
 mutate (n\_rel = n / sum(n)\*100, n\_rel\_ind = 100/12) %>%  
 select(month, n, n\_rel, n\_rel\_ind, perc\_y)  
t\_month\_y

## Source: local data frame [12 x 5]  
##   
## month n n\_rel n\_rel\_ind perc\_y  
## 1 jan 1403 3.1032271 8.333333 0.10121169  
## 2 feb 2649 5.8591936 8.333333 0.16647792  
## 3 mar 477 1.0550530 8.333333 0.51991614  
## 4 apr 2932 6.4851474 8.333333 0.19679400  
## 5 may 13766 30.4483422 8.333333 0.06719454  
## 6 jun 5341 11.8134967 8.333333 0.10222805  
## 7 jul 6895 15.2507133 8.333333 0.09093546  
## 8 aug 6247 13.8174338 8.333333 0.11013286  
## 9 sep 579 1.2806618 8.333333 0.46459413  
## 10 oct 738 1.6323461 8.333333 0.43766938  
## 11 nov 3970 8.7810489 8.333333 0.10151134  
## 12 dec 214 0.4733361 8.333333 0.46728972

I mesi di maggior conversione sono quelli con i minori contatti. Se anche l'information Value e il p.value sono favorevoli, potre non includere month perché non mi spiego l'associazione negativa tra conversion e volumi. Month è il mese di ultima chiamata, quella di chiusura. Il call center potrebbe aver fatto comunque molte chiamate non di chiusura nel mese, non è detto che maggio sia stato mese di maggior lavoro di dicembre. Tuttavia laddove si concentrano più ultime chiamate di chiusura si converte meno. Ci devo tornare a riflettere.

month\_woe <- bank0 %>%  
 select(month, y) %>%  
 group\_by(month) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
month\_woe$woe <- log(month\_woe$perc\_no / month\_woe$perc\_y)  
month\_IV <- sum((month\_woe$perc\_no - month\_woe$perc\_y) \* month\_woe$woe)

Varibile più predittiva finora!

## Duration

g\_duration <- ggplot(bank0, aes(x = 1, y = duration)) +   
 geom\_boxplot()  
g\_duration

g\_duration\_y <- ggplot(bank0, aes(x = y, y = duration)) +   
 geom\_boxplot()  
g\_duration\_y

A chiamate più lunghe corrisponde una maggiore propensione alla conversion! Da capire se si fa riferimento alla penultima o all'ultima chiamata (credo la penultima sennò non avrebbe senso utilizzarla in un modello predittivo). Se è la penultima chiamata, si può immaginare che o un cliente di suo interessato ti tenga dipiù al telefono o che un venditore che lo tiene di più al telefono lo invoglia a comprare. Se l'ultima chiamata, be, c'è da valutae l'aumento dei tempi dovuto all'informativa legale e alla parte sottoscrittiva. Andrà analizzato comunque il rapporto con giorno e mese, perché in quelle varaiabili bassi volumi (poche chiamate, chiamate molto lunghe?) portavano alta conversion.

summary(bank0$duration)

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

duration\_quantile <- quantile(bank0$duration[bank0$duration != 0], probs = seq(0.1,1,0.1))  
bank0$duration\_class <- cut(bank0$duration, breaks = c(-1, 0, duration\_quantile), right = TRUE, labels = c("no call", "(0,58]", "(58,89]", "(89,117]", "(117,147]"," (147,180]", "(180,223]", "(223,280]", "(280,368]", "(368,548]", "(548,4918]"))  
  
t\_duration\_class\_y <- bank0 %>%  
 group\_by (duration\_class) %>%  
 summarise(n(), mean(y == "yes"))

g\_duration\_class\_y <- ggplot(bank0, aes(x = factor(duration\_class), fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_duration\_class\_y

Esatto. al crescere della durata della chiamata cresce la quota di coloro che sottoscrivono, o viceversa.

#devo togliere i tre valori nulli di duration altrimenti la formula va in errore  
duration\_class\_woe <- bank0 %>%  
 filter(duration\_class != "no call") %>%  
 select(duration\_class, y) %>%  
 group\_by(duration\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
duration\_class\_woe$woe <- log(duration\_class\_woe$perc\_no / duration\_class\_woe$perc\_y)  
duration\_class\_IV <- sum((duration\_class\_woe$perc\_no - duration\_class\_woe$perc\_y) \* duration\_class\_woe$woe)

Ampiamente la più predittiva.

## Campaign

g\_campaign <- ggplot(bank0, aes(x = 1, y = campaign)) +   
 geom\_boxplot()  
g\_campaign

g\_campaign\_y <- ggplot(bank0, aes(x = y, y = campaign)) +   
 geom\_boxplot()  
g\_campaign\_y

Non mi sembra discrimini molto il numero di contatti precedenti.

bank0$campaign <- as.numeric(bank0$campaign)  
  
campaign\_quantile <- quantile(bank0$campaign, probs = seq(0.1,1,0.1)) #considerando i quantili il binning lo faccio a mano  
  
bank0$campaign\_class <- cut(bank0$campaign, breaks = c(0, 1, 2, 3, 4, 5, max(bank0$campaign)+1), right = TRUE, labels = c("1", "2", "3", "4", "5", "up 5"))  
summary(bank0$campaign\_class)

## 1 2 3 4 5 up 5   
## 17544 12505 5521 3522 1764 4355

bank\_campaign\_na <- bank0[is.na(bank0$campaign\_class),]   
nrow(bank\_campaign\_na) #ok

## [1] 0

t\_campaign\_class\_y <- bank0 %>%  
 group\_by (campaign\_class) %>%  
 summarise(n(), media = mean(y == "yes"))

g\_campaign\_class\_y <- ggplot(bank0, aes(x = factor(campaign\_class), fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_campaign\_class\_y

Andamento normale. La maggiore conversion si ha per tra il 30 e il quarantesimo percentile di contatti (da capire quanti contatti sono).

campaign\_class\_woe <- bank0 %>%  
 select(campaign\_class, y) %>%  
 group\_by(campaign\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
campaign\_class\_woe$woe <- log(campaign\_class\_woe$perc\_no / campaign\_class\_woe$perc\_y)  
campaign\_class\_IV <- sum((campaign\_class\_woe$perc\_no - campaign\_class\_woe$perc\_y) \* campaign\_class\_woe$woe)  
  
  
############################################  
bank0 <- bank0 %>%  
 mutate(c\_class = ntile(campaign, 10))  
  
c\_class\_woe <- bank0 %>%  
 select(c\_class, y) %>%  
 group\_by(c\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
c\_class\_woe$woe <- log(c\_class\_woe$perc\_no / c\_class\_woe$perc\_y)  
c\_class\_IV <- sum((c\_class\_woe$perc\_no - c\_class\_woe$perc\_y) \* c\_class\_woe$woe)  
c\_class\_IV

## [1] 0.5803398

table(bank0$campaign, bank0$c\_class)

##   
## 1 2 3 4 5 6 7 8 9 10  
## 1 4522 4521 4521 3980 0 0 0 0 0 0  
## 2 0 0 0 541 4521 4521 2922 0 0 0  
## 3 0 0 0 0 0 0 1599 3922 0 0  
## 4 0 0 0 0 0 0 0 599 2923 0  
## 5 0 0 0 0 0 0 0 0 1598 166  
## 6 0 0 0 0 0 0 0 0 0 1291  
## 7 0 0 0 0 0 0 0 0 0 735  
## 8 0 0 0 0 0 0 0 0 0 540  
## 9 0 0 0 0 0 0 0 0 0 327  
## 10 0 0 0 0 0 0 0 0 0 266  
## 11 0 0 0 0 0 0 0 0 0 201  
## 12 0 0 0 0 0 0 0 0 0 155  
## 13 0 0 0 0 0 0 0 0 0 133  
## 14 0 0 0 0 0 0 0 0 0 93  
## 15 0 0 0 0 0 0 0 0 0 84  
## 16 0 0 0 0 0 0 0 0 0 79  
## 17 0 0 0 0 0 0 0 0 0 69  
## 18 0 0 0 0 0 0 0 0 0 51  
## 19 0 0 0 0 0 0 0 0 0 44  
## 20 0 0 0 0 0 0 0 0 0 43  
## 21 0 0 0 0 0 0 0 0 0 35  
## 22 0 0 0 0 0 0 0 0 0 23  
## 23 0 0 0 0 0 0 0 0 0 22  
## 24 0 0 0 0 0 0 0 0 0 20  
## 25 0 0 0 0 0 0 0 0 0 22  
## 26 0 0 0 0 0 0 0 0 0 13  
## 27 0 0 0 0 0 0 0 0 0 10  
## 28 0 0 0 0 0 0 0 0 0 16  
## 29 0 0 0 0 0 0 0 0 0 16  
## 30 0 0 0 0 0 0 0 0 0 8  
## 31 0 0 0 0 0 0 0 0 0 12  
## 32 0 0 0 0 0 0 0 0 0 9  
## 33 0 0 0 0 0 0 0 0 0 6  
## 34 0 0 0 0 0 0 0 0 0 5  
## 35 0 0 0 0 0 0 0 0 0 4  
## 36 0 0 0 0 0 0 0 0 0 4  
## 37 0 0 0 0 0 0 0 0 0 2  
## 38 0 0 0 0 0 0 0 0 0 3  
## 39 0 0 0 0 0 0 0 0 0 1  
## 41 0 0 0 0 0 0 0 0 0 2  
## 43 0 0 0 0 0 0 0 0 0 3  
## 44 0 0 0 0 0 0 0 0 0 1  
## 46 0 0 0 0 0 0 0 0 0 1  
## 50 0 0 0 0 0 0 0 0 0 2  
## 51 0 0 0 0 0 0 0 0 0 1  
## 55 0 0 0 0 0 0 0 0 0 1  
## 58 0 0 0 0 0 0 0 0 0 1  
## 63 0 0 0 0 0 0 0 0 0 1

Wow, che scoperta. Se raggruppo campaign in classi definite da me in base ai quantili, l'information value è 0.08, abbastanza basso. Se uso ntile è 0.58, suspicious. E infatti è sospetto: ntile raggruppa in 10 gruppi di pari dimensioni, ma questo per distribuzioni molto strette comporta che in due gruppi diversi (livelli diversi, valori diversi) ci sia la stessa informazione. Se si vede la cross table, i gruppi 1 2 e 3 sono identici, ma portano (casualmente!) dei woe molto diversi!

## pdays

Considerando che questa variabile include info quantitative (giorni dall'ultima campagna) e qualitativa (contattato si/no) procederò subito con una fattorializzazione.

g\_pdays\_quant <- bank0 %>%   
 filter(pdays != -1) %>%   
 ggplot(aes(x = 1, y = pdays)) + geom\_boxplot()  
  
g\_pdays\_quant\_y <- bank0 %>%   
 filter(pdays != -1) %>%   
 ggplot(aes(x = y, y = pdays)) + geom\_boxplot()

Per chi è stato contattato in passato, è evidente che se è passato meno tempo tende a convertire di pià.

Proviamo un raggruppamento in classi. Potrei fare una classe per i mai contattati e poi 10 classi di pari frequenza.

pdays\_quantile <- quantile(bank0$pdays[bank0$pdays != -1], probs = seq(0.1,1,0.1))  
bank0$pdays\_class <- cut(bank0$pdays, breaks = c(-2, 0, pdays\_quantile), right = TRUE, labels = c("no campaign", "(0,91]", "(91,108]", "(108,159]", "(159,181]"," (181,194]", "(194,258]", "(258,300]", "(300,343]", "(343,362]", "(362,871]"))  
table(bank0$pdays\_class)

##   
## no campaign (0,91] (91,108] (108,159] (159,181] (181,194]   
## 36954 844 817 819 835 814   
## (194,258] (258,300] (300,343] (343,362] (362,871]   
## 833 825 879 765 826

#le classi non sono di uguale numerosità, ma è normale perché ci sono numeri che si ripetono. ntile infatti crea classi di uguale numerosità in cui uno stesso valore può stare in due classi diverse. non va bene.

g\_pdays\_class <- ggplot(bank0, aes(x = pdays\_class)) +  
 geom\_bar()  
g\_pdays\_class#ovvio

g\_pdays\_class\_y <- ggplot(bank0, aes(x = pdays\_class, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_pdays\_class\_y

Variabile molto predittiva con andamento un po' anomalo. Molta conversione per pochi giorni trascorsi, poi meno, poi risale.

pdays\_class\_woe <- bank0 %>%  
 select(pdays\_class, y) %>%  
 group\_by(pdays\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
pdays\_class\_woe$woe <- log(pdays\_class\_woe$perc\_no / pdays\_class\_woe$perc\_y)  
pdays\_class\_IV <- sum((pdays\_class\_woe$perc\_no - pdays\_class\_woe$perc\_y) \* pdays\_class\_woe$woe)

0.34. Buona predittività.

## Previous

g\_previous\_quant <- ggplot(bank0, aes(x = 1, y = previous)) +   
 geom\_boxplot()  
g\_previous\_quant

g\_previous\_quant\_y <- ggplot(bank0, aes(x = y, y = previous)) +   
 geom\_boxplot()  
g\_previous\_quant\_y

Quasi nessuno è stato contattato, è una distribuzione fortemente asimettrica. Provo a fare i medesimi grafici togliendo il valore zero.

g\_previous\_nz\_quant <- bank0 %>%   
 filter(previous > 0) %>%   
 ggplot(aes(x = 1, y = previous)) + geom\_boxplot()  
g\_previous\_nz\_quant

g\_previous\_nz\_quant\_y <- bank0 %>%   
 filter(previous > 0) %>%   
 ggplot(aes(x = y, y = previous)) + geom\_boxplot()  
g\_previous\_nz\_quant\_y

Comunque tutti valori molto molto bassi.

Anche qui provo un raggruppamento in classi, 0 e non 0.

previous\_quantile <- quantile(bank0$previous[bank0$previous > 0], probs = seq(0.1,1,0.1))  
#Considerando i decili, preferisco fare 7 classi di non pari frequenza: da 0 a 6 e maggiore di 6  
bank0$previous\_class <- cut(bank0$previous, breaks = c(0, 1, 2, 3, 4, 5, 7, max(bank0$previous)+1), right = FALSE, labels = c("0 contact", "1 contact", "2 contact", "3 contact", "4 contact","5 or 6 contact", "+ 6 contact"))  
table(bank0$previous\_class)

##   
## 0 contact 1 contact 2 contact 3 contact 4 contact   
## 36954 2772 2106 1142 714   
## 5 or 6 contact + 6 contact   
## 736 787

summary(bank0$previous\_class)#no NA

## 0 contact 1 contact 2 contact 3 contact 4 contact   
## 36954 2772 2106 1142 714   
## 5 or 6 contact + 6 contact   
## 736 787

#le classi non sono di uguale numerosità, ma è normale perché ci sono numeri che si ripetono. ntile infatti crea classi di uguale numerosità in cui uno stesso valore può stare in due classi diverse. non va bene.

g\_previous\_class <- ggplot(bank0, aes(x = previous\_class)) +  
 geom\_bar()  
g\_previous\_class #6 contact è classe davvero poco numerosa, non so se ricondurla a 5 o più => yes

g\_previous\_class\_y <- ggplot(bank0, aes(x = previous\_class, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_previous\_class\_y

Variabile molto predittiva , tendenzialmente più contatti ci sono stati per la precedente campagna più si converte per questa. Chiaro, è gente interessata che forse ha anche convertito per la precedente campagna (da analizzare la correlazione tra previos e poutcome)

previous\_class\_woe <- bank0 %>%  
 select(previous\_class, y) %>%  
 group\_by(previous\_class) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
previous\_class\_woe$woe <- log(previous\_class\_woe$perc\_no / previous\_class\_woe$perc\_y)  
previous\_class\_IV <- sum((previous\_class\_woe$perc\_no - previous\_class\_woe$perc\_y) \* previous\_class\_woe$woe)

0.22 discreta predittività.

## Poutcome

g\_poutcome <- ggplot(bank0, aes(x = poutcome)) +  
 geom\_bar()  
g\_poutcome

g\_poutcome\_y <- ggplot(bank0, aes(x = poutcome, fill = y)) +  
 geom\_bar(position = "fill") +  
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_poutcome\_y

Unknown saranno i zero contact e i no camaign. Lo capirò analizzando la relazione tra queste variabili. Ovviamente per chi ha convertito la scora volta grande conversion anche stavolta.

t\_poutcome\_y <- bank0 %>%  
 group\_by (poutcome) %>%  
 summarise (n = n(), perc\_y = mean(y=="yes"))  
t\_poutcome\_y

## Source: local data frame [4 x 3]  
##   
## poutcome n perc\_y  
## 1 failure 4901 0.12609671  
## 2 other 1840 0.16684783  
## 3 success 1511 0.64725347  
## 4 unknown 36959 0.09161503

poutcome\_woe <- bank0 %>%  
 select(poutcome, y) %>%  
 group\_by(poutcome) %>%  
 summarise(n\_no = sum(y == "no"), n\_y = sum(y == "yes")) %>%  
 mutate (perc\_no = n\_no / sum(n\_no), perc\_y = n\_y / sum(n\_y)) %>%  
 select (starts\_with("perc"))  
poutcome\_woe$woe <- log(poutcome\_woe$perc\_no / poutcome\_woe$perc\_y)  
poutcome\_IV <- sum((poutcome\_woe$perc\_no - poutcome\_woe$perc\_y) \* poutcome\_woe$woe)

0.51, molto predittiva, ovviamente.

## WOE

Creiamo il vettore dell'information value e mettiamolo in un grafico a barre

names(bank0)

## [1] "age" "job" "marital" "education"   
## [5] "default" "balance" "housing" "loan"   
## [9] "contact" "day" "month" "duration"   
## [13] "campaign" "pdays" "previous" "poutcome"   
## [17] "y" "age\_num" "age\_class" "balance\_class"   
## [21] "duration\_class" "campaign\_class" "c\_class" "pdays\_class"   
## [25] "previous\_class"

IV <- c(age\_class\_IV, job\_IV, marital\_IV, education\_IV, default\_IV, balance\_class\_IV, housing\_IV, loan\_IV, contact\_IV, day\_IV, month\_IV, duration\_class\_IV, campaign\_class\_IV, pdays\_class\_IV, previous\_class\_IV, poutcome\_IV)  
Variables <- c("age\_class", "job", "marital", "education", "default", "balance\_class", "housing", "loan", "contact", "day", "month", "duration\_class", "campaign\_class", "pdays\_class", "previous\_class", "poutcome")  
t\_IV\_all <- data.frame(Variables, IV)  
t\_IV\_all <- t\_IV\_all %>%  
 arrange(desc(IV))

Nota su duration: ho scoperto da uno studio su internet che duration fa riferimento alla chiamata di chiusura; perciò non è utilizzabile ai nostri fini, e l'information value sopra lo 0.5, anzi, sopra l'unità la rendeva una variabile assai sospetta. Quindi non la includeremo.

g\_IV\_all <- ggplot(t\_IV\_all, aes(x= reorder(Variables, IV), y= IV)) +  
 geom\_bar(stat='identity') +  
 coord\_flip()  
g\_IV\_all

Duration class è la più sospetta, troppo predittiva. Può darsi che i miei sospetti sul fatto che si riferisca alla chiamata di chiusura siano veri.

# GGpairs

#bank0\_ggpairs <- ggpairs(bank0\_sm) non è venuto bene

# Tre tipologie di relazione tra tre variabili

Forniamo un breve commento ai tre esempi che seguono.

Al punto 1 presentiamo la situazione detta *spiegazione*. La tab. 3 mostra la tabella a doppia entrata originale che mette in relazione il numero delle pompe antincendio presenti sul luogo di un incendio e l’entità dei danni dello stesso. Come si vede tra le due variabili vi è relazione: solo il 30% degli incidenti è caratterizzato da danni superiori a 10.000$ se le autopompe non sono più di 2, sale al 59% se le autopompe sono più di 2. Naturalmente il numero di autopompe che vengono inviate sul luogo dell’incendio sarà legato alle dimensioni dell’incendio stesso e quindi al presumibile danno prodotto. Quindi occorre *controllare la relazione originale* introducendo la variabile “dimensione dell’incendio”. Si vede così in tab 1 che a parità di dimensioni dell’incendio, non vi è alcuna relazione tra numero di autopompe e ammontare del danno: se le dimensioni sono ridotte solo il 5% degli incendi produce un danno superiore a 10.000$, indipendentemente dal numero di autopompe presenti; questa percentuale sale all’80% quando l’incendio è di ampie dimensioni, anche qui indipendentemente dal numero di autopompe coinvolte. E’ ovvio che il numero di autopompe non può determinare le dimensioni dell’incendio, ma ne è una sua conseguenza. Il numero di autopompe non determina neppure l’ammontare dei danni. In realtà le dimensioni causano sia il numero di autopompe che l’ammontare del danno, così che la relazione tra autopompe e danno è fittizia, dovuta esclusivamente alla terza variabile, perciò *spuria*. => Non so se il confondimento implica che la relazione sia spuria.

Al punto 2 presentiamo la situazione detta *interpretazione*. La tab. 6 mostra la tabella a doppia entrata originale che mette in relazione sesso e coinvolgimento in incidenti. Come si vede le donne hanno meno incidenti dei maschi: 32% e 44% rispettivamente. Bisogna però considerare che, al di là della prudenza e delle capacità di guida, il semplice fatto di percorrere mediamente più chilometri espone ad una probabilità maggiore di incorrere in incidenti. Se introduciamo come terza variabile la percorrenza chilometrica annua (tab. 4) vediamo che la relazione tra sesso e incidenti sparisce: se la percorrenza è bassa il 25% dei conducenti è coinvolto in incidenti qualunque sia il suo sesso; se la percorrenza è alta questa percentuale sale al 52% sia tra i maschi che tra le femmine. *A prima vista, la relazione tra sesso e incidenti potrebbe apparire spuria come nell’esempio precedente. Qui però vi è una differenza fondamentale: la terza variabile non è la causa delle due variabili originali (la percorrenza non causa il sesso del conducente)*. *Siamo invece in presenza di una catena causale: il sesso causa la percorrenza che a sua volta causa il coinvolgimento in incidenti. Insomma la relazione tra sesso e incidenti non è fittizia*, appare a prima vista incomprensibile perché è *mediata* da una variabile intermedia, la percorrenza. E’ in questo senso che si dice che la relazione originale è interpretata. Anche qui forniamo la tab. 5, forma compatta della tab. 4.=> l'interpretazione è quando tra la relazione causale tra2 variabili se ne include una terza intermedia che cambia l'interpretazione della relazione, che comunque ha senso, non è spuria come nobel e cioccolato.

Al punto 3 presentiamo la situazione detta *specificazione*. La tab. 9 mostra la tabella a doppia entrata originale che mette in relazione orientamento politico e interesse per la politica. La tab. 7 mostra cosa accade *quando introduciamo come variabile di controllo*, il titolo di studio. In origine tra coloro che si collocano a sinistra il 28% ha un interesse alto, che scende al 15% tra i soggetti di destra. Questa differenza resta anche quando si introduce la terza variabile. Bisogna però notare che tra coloro che hanno un basso grado di istruzione i valori sono pari a 19% e 7%, cioè valori più bassi di quelli della relazione bivariata e con una differenza tra sinistra e destra equivalente. Tra coloro invece che hanno un elevato titolo di studio l’interesse per la politica aumenta: coloro che sono molto interessati crescono rispettivamente al 36% e al 18%, rispettivamente per sinistra e destra, con una differenza tra i due pari a 18 punti percentuali, contro i 13 della relazione originaria. Qui entrambe le variabili “orientamento” e “titolo” influenzano la dipendente ed è per questo che si parla di “specificazione” della relazione originale.=> a me questo sembra effetto di interazione: l'introduzione di una terza variabile porta a un effetto congiunto delle due indipendenti che è piu (perché cambiano le differenze tra i valori) della loro moltiplicazione.

# Link utili

* [epidemiologia1](http://www.quadernodiepidemiologia.it/epi/freq/stn_mis.htm)
* [epidemiologia2](http://www.quadernodiepidemiologia.it/epi/assoc/ass_nc.htm)
* [confounding and interaction](https://www.ctspedia.org/do/view/CTSpedia/InterConfound)
* [stratified analysis - il più importante](http://www.sjsu.edu/faculty/gerstman/StatPrimer/stratified.PDF)

L'ultimo link, un pdf, è semplicemente illuminante. Così come il paragrafo che ho copiato sopra.

# Confounding and interaction: crude vs stratified analysis.

Confounding (from the Latin confundere: to mix together) is a *distortion of an association* between an exposure (E) and disease (D) brought about by extraneous factors (C1, C2, etc). Since confounding is a systematic (not random) error, hypothesis testing cannot be used to detect it. It is a judgement based science. The analyst should start with simple comparisons of means and proportions.

Interaction, as distinct from confounding, is the interdependent operation of two or more factors to produce an unanticipated effect. Interactions is usually addressed by reporting data by subgroups.

Measures of association in the aggregate are called crude measures of association.

Stratification might reveal otherwise hidden confounding and interaction. Example with RR; se il RR crudo è 4.00, e stratificando per i tre livelli di C abbiamo sempre RRC=4.00, allora la stratificazione è superflua. Se RRC = 1.00 sempre, allora c'è confounding. Se RRC = 1.00, 3.00, 25.00, c'è interazione.

Spesso c'è in parte confounding e in parte interaction. The best estimate of association is both valid and precise. If interaction is present, strata-specific measures of association are reported. If interaction is absent but confounding is present, summary (adjusted) measures of association are reported. If neither interaction nor confounding are present, crude (unadjusted) measures of association are reported. In general, the most parsimoniously unconfounded presentation of the data is preferred. If the association between the exposure and disease is not found by scrutinizing the data in the 2-by-2 table, it's hard to support. Simple is better. Qui si parla di RR e di quale va riportato se c'è confounding e/o se c'è interaction.

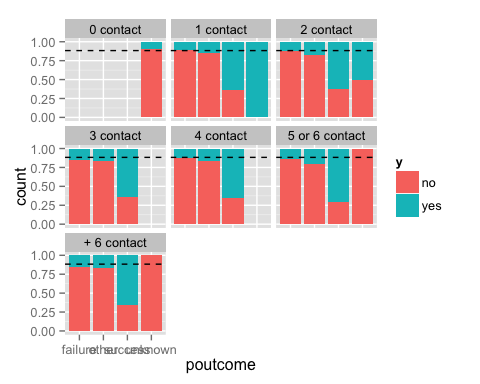
## previous e poutcome

In questo caso sia il numero di contatti della precedente campagna che l'esito della precedente campagna sono predittive. Solo che osservando che al crescere dei contatti della precedente campagna cresce la conversion dell'attuale, mi domando se in realtà non sia stato la predisposizione a convertire della precedente campagna a generare sia tanti contatti (mi faccio chiamare e risponso perché interessato) che l'attuale conversion (ero e sono interessato a sottoscrivere prodotti aggiuntivi).

t\_previous\_y <- bank0 %>%  
 group\_by(previous\_class) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes")) %>%  
 mutate(freq\_rel = n / sum(n)) %>%  
 select(previous\_class, y\_rate)  
  
t\_previous\_poutcome\_y <- bank0 %>%  
 group\_by(previous\_class, poutcome) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes")) %>%  
 mutate(freq\_rel = n / sum(n)) %>%  
 select(previous\_class, poutcome, y\_rate) %>%  
 spread(previous\_class, y\_rate)  
  
g\_previous\_class\_poutcome\_f <- g\_previous\_class + facet\_wrap(~poutcome)  
g\_previous\_class\_poutcome <- ggplot(bank0, aes(x = factor(previous\_class), fill = poutcome)) +  
 geom\_bar(position = "fill")  
g\_previous\_poutcome\_y <- g\_previous\_class\_y + facet\_wrap(~poutcome)  
g\_previous\_class\_y

Ora, per ogni esito della precedente campagna, il numero dei contatti discrimina molto meno che nella crude analysis (tranne che per esito sconosciuto, che però coincide con contati zero, cioè nessuna partecipazione alla precedente campagna). Perciò la mia ipotesi è abbastanza confermata, poutcome ci permette di interpretare abbastanza la relazione tra contatti e conversion. Come controprova, faccio il condizionamento al contrario:

g\_poutcome\_previos\_y <- g\_poutcome\_y + facet\_wrap(~previous\_class)  
g\_poutcome\_previos\_y



Per ogni contatto c'è una bella discriminazione della conversion, perciò la mia ipotesi era esatta. è la predispozione a convertire nella vecchia campagna che contribuisce a determinare il numero di contatti e la nuova conversion.

*La conclusione è che se includeremo nel modello previous andrebbe incluso anche poutcome, come variabile di controllo? Oppure basta poutcome? Mah, alla fine soprattutto per il livello failure, il numero di contatti discrimina, quindi includerei entrambe.*.

## Age e job

g\_age\_class <- ggplot(bank0, aes(x = age\_class)) +   
 geom\_bar()  
g\_age\_class\_y <- ggplot(bank0, aes(x = age\_class, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_age\_class\_y

g\_job

g\_job\_y

g\_job\_age\_class <- ggplot(bank0, aes(x = job, fill = age\_class)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2)  
g\_job\_age\_class

#distribuzione molto sbilanciata della variabile di controllo per i due livelli studenee pensionato, probabile che age aiuti a capire la relazione tra job e conversion.  
  
g\_job\_age\_class\_y <- ggplot(bank0, aes(x = job, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~age\_class) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))  
g\_job\_age\_class\_y

La proporzione delle fasce di età nei job mostra squilibri; ci sono molte professioni dove quasi tutti sono adulti, gli studenti poi sono quasi tutti giovani e i pensionati sono metà adulti e metà anziani. Mentre la relazione tra job e conversion ci dice che i pensionati convertono di più, controllando per l'età vediamo che convertono non se pensionati, ma se pensionati vecchi. Se vai in pensione adulto non converti.

Se ci limitiamo agli anziani, tutti i job convertono molto, ma comunque i pensionati di più. Quindi è chiaro che i pensionati crudamente convertono di più perché in essi la metà è anziana, ma comunque tra gli anziani i pensionati convertono più degli altri. Essere vecchio ti fa convertire, essere vecchio e in pensione ancora di più. Vi è una qualche interazione.

Invece gli studenti convertono anche se adulti; se si osserva la conversion per fasce di età, gli adulti convertono nella media; se invece sei adulto ma studente converti più della media, perciò essere studente è legato al convertire a prescindere (in parte) dall'età.

Il problema andrebbe complicato ragionando sui volumi bassi di alcuni sottogruppi.

## Age e marital

g\_age

g\_age\_y

g\_age\_class

g\_age\_class\_y

g\_marital

g\_marital\_y

Voglio capire se i single convertono in quanto giovani, con l'età a fare da confounder o effect modifier.

g\_marital\_age\_class <- ggplot(bank0, aes(x = marital, fill = age\_class)) +   
 geom\_bar(position = "fill")   
g\_marital\_age\_class

#infatti la proporzione di giovani nei single è altissima, negli altri due casi lo stato maritale e l'età sono quasi indipendenti  
g\_marital\_age\_class\_y <- ggplot(bank0, aes(x = marital, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~age\_class)  
g\_marital\_age\_class\_y

t\_marital\_age\_class <- bank0 %>%  
 group\_by(marital, age\_class) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes")) %>%  
 mutate(freq\_rel = n / sum(n))  
  
t\_marital\_age\_class\_y <- bank0 %>%  
 group\_by(marital, age\_class) %>%  
 summarise(y\_rate = mean(y == "yes")) %>%  
 select(marital, age\_class, y\_rate) %>%  
 spread(marital, y\_rate)  
t\_marital\_y

## Source: local data frame [3 x 3]  
##   
## marital n perc\_y  
## 1 divorced 5207 0.1194546  
## 2 married 27214 0.1012347  
## 3 single 12790 0.1494918

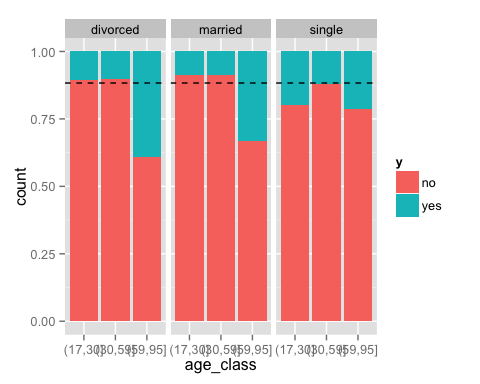
Gli anziani single sono 61, quindi la predittività è poco significativa.

La mia ipotesi di partenza è che i single convertissero in quanto giovani. L'ipotesi sembra confermata; innanzitutto la proporzione di giovani tra i single è molto più alta che negli altri due stati maritali, dove la proporzione è quasi identica (e ci può stare, e dove non è identica è perché in divorced ci sono anche i vedovi). Analizzando poi il comportamento dei tre stati per ogni fascia di età, si vede che se sei giovane converti solo se sei anche single, se sei giovane ma sposato (2060 casi) non converti. Quindi l'età non fa da confounder, ma da effect modifier, perché nella fascia giovanile la conversion dei single è potenziata rispetto a quella complessiva, per quanto il trend sia identico (minor conversion gli sposati, maggiore i single). Ma in realtà abbiamo scoperto che i giovani convertono solo se single. Gli anziani convertono assai, come è ovvio, ma il comportamento dei tre stati maritali (per quanto i numeri siano bassi) si invertono; anche qui c'è interazione! E di sicuro è dovuta allo specifico comportamento degli anziani vedovi.

Nella fascia adulti i tre stati maritali convertono come per tutto il portafoglio, invece.

Ma non sarà che è lo stato maritale a fare da confounding?

g\_age\_class\_marital\_y <- ggplot(bank0, aes(x = age\_class, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~marital)  
g\_age\_class\_marital\_y



Non abbiamo trovato confounding, ma una relazione di interazione, che ci fa dire soprattutto che i giovani che convertono sono i single. *Si potrebbe pensare di includere un effetto di interazione nel modello*.

## Education and job

Convertono di più i laureti. Magari non è questione di job che fai ma di educazione che hai.

g\_education

g\_job

g\_job\_y

g\_education\_y

g\_education\_job <- ggplot(bank0, aes(x = job, fill = education)) +   
 geom\_bar(position = "fill")   
g\_education\_job

g\_education\_job\_y <- ggplot(bank0, aes(x = job, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~education) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))  
g\_education\_job\_y

Vedo una interazione solo nel management, che essendo pieno di laureati marginalmente coverte di più, ma se non laureato non converte. Bisogna capire i volumi, potrebbero essere bassi. Per il resto vedo comportamenti simili alla analisi cruda, con un main effect dell'education. Tuttavia anche imprenditori e liberi professionisti hanno una buona quota di laureati, e tuttavia non convertono marginalmente e gli imprenditori nemmeno solo se laureati. Il lavoro vince sulla job, diciamo, nel senso che non gli fa esercitare nemmeno il main effect (e non so se questa è una interazione, probabilmente sì).

t\_education\_job <- bank0 %>%  
 group\_by(job, education) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes"))  
t\_education\_job

## Source: local data frame [48 x 4]  
## Groups: job  
##   
## job education n y\_rate  
## 1 admin. primary 209 0.05741627  
## 2 admin. secondary 4219 0.11898554  
## 3 admin. tertiary 572 0.17307692  
## 4 admin. unknown 171 0.10526316  
## 5 blue-collar primary 3758 0.05800958  
## 6 blue-collar secondary 5371 0.08061813  
## 7 blue-collar tertiary 149 0.16107383  
## 8 blue-collar unknown 454 0.07268722  
## 9 entrepreneur primary 183 0.06557377  
## 10 entrepreneur secondary 542 0.09594096  
## .. ... ... ... ...

t\_education\_job\_y <- bank0 %>%  
 group\_by(job, education) %>%  
 summarise(y\_rate = mean(y == "yes")) %>%  
 select(job, education, y\_rate) %>%  
 spread(job, y\_rate)  
t\_education\_job\_y

## Source: local data frame [4 x 13]  
##   
## education admin. blue-collar entrepreneur housemaid management  
## 1 primary 0.05741627 0.05800958 0.06557377 0.07814992 0.07482993  
## 2 secondary 0.11898554 0.08061813 0.09594096 0.08607595 0.08652988  
## 3 tertiary 0.17307692 0.16107383 0.07580175 0.12716763 0.14536598  
## 4 unknown 0.10526316 0.07268722 0.09210526 0.08888889 0.19834711  
## Variables not shown: retired (dbl), self-employed (dbl), services (dbl),  
## student (dbl), technician (dbl), unemployed (dbl), unknown (dbl)

Non vedo grande aiuto dell'education a interpretare la relazione tra job e conversion, non includerei un effetto di interazione.

## Housing and loan

g\_loan

g\_housing

g\_loan\_y

g\_housing\_y

g\_loan\_housing <- ggplot(bank0, aes(x = loan, fill = housing)) +   
 geom\_bar(position = "fill")   
g\_loan\_housing

g\_loan\_housing\_y <- ggplot(bank0, aes(x = loan, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~housing) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))  
g\_loan\_housing\_y

Vedo solo un main effect. Se hai il mutuo converti meno; dato che hai un prestito, se hai anche il mutuo converti meno se no converti di più. Magari il noxno accentua la conversion più del semplice main effect, ma non so se vale la pena testare una interazione per questo.

t\_loan <- bank0 %>%  
 group\_by(loan) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes"))  
t\_loan

## Source: local data frame [2 x 3]  
##   
## loan n y\_rate  
## 1 no 37967 0.12655727  
## 2 yes 7244 0.06681391

t\_housing <- bank0 %>%  
 group\_by(housing) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes"))  
t\_housing

## Source: local data frame [2 x 3]  
##   
## housing n y\_rate  
## 1 no 20081 0.1670236  
## 2 yes 25130 0.0769996

t\_loan\_housing <- bank0 %>%  
 group\_by(loan, housing) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes"))  
t\_loan\_housing

## Source: local data frame [4 x 4]  
## Groups: loan  
##   
## loan housing n y\_rate  
## 1 no no 17204 0.18222506  
## 2 no yes 20763 0.08043154  
## 3 yes no 2877 0.07612096  
## 4 yes yes 4367 0.06068239

t\_loan\_housing\_y <- bank0 %>%  
 group\_by(loan, housing) %>%  
 summarise(y\_rate = mean(y == "yes")) %>%  
 select(loan, housing, y\_rate) %>%  
 spread(loan, y\_rate)  
t\_loan\_housing\_y

## Source: local data frame [2 x 3]  
##   
## housing no yes  
## 1 no 0.18222506 0.07612096  
## 2 yes 0.08043154 0.06068239

ggplot(t\_loan\_housing, aes(x = loan, y = y\_rate, col = housing, group = housing)) +   
 geom\_point (size = 4) +   
 geom\_line(col= "black", linetype = 2)

Graficamente l'interaction effect è evidente e logicamente ha senso.

## Month and day

t\_month\_y

## Source: local data frame [12 x 5]  
##   
## month n n\_rel n\_rel\_ind perc\_y  
## 1 jan 1403 3.1032271 8.333333 0.10121169  
## 2 feb 2649 5.8591936 8.333333 0.16647792  
## 3 mar 477 1.0550530 8.333333 0.51991614  
## 4 apr 2932 6.4851474 8.333333 0.19679400  
## 5 may 13766 30.4483422 8.333333 0.06719454  
## 6 jun 5341 11.8134967 8.333333 0.10222805  
## 7 jul 6895 15.2507133 8.333333 0.09093546  
## 8 aug 6247 13.8174338 8.333333 0.11013286  
## 9 sep 579 1.2806618 8.333333 0.46459413  
## 10 oct 738 1.6323461 8.333333 0.43766938  
## 11 nov 3970 8.7810489 8.333333 0.10151134  
## 12 dec 214 0.4733361 8.333333 0.46728972

g\_duration\_class\_y

g\_month

g\_month\_y

t\_month\_duration\_class <- bank0 %>%  
 group\_by(month, duration\_class) %>%  
 summarise(n = n(), y\_rate = mean(y == "yes")) %>%  
 mutate(n\_rel = n / sum(n)) %>%  
 select(duration\_class, month, n\_rel) %>%  
 spread(duration\_class, n\_rel, fill = 0) %>%  
 left\_join(t\_month\_y, by = "month") %>%  
 select(-n, -n\_rel, -n\_rel\_ind) %>%  
 arrange(desc(perc\_y))

I mesi di maggior conversion hanno pochissimi volumi. Posto che volumi così bassi non sono significativi, è curioso che la conversion sia sempre altissima e non è mai, che so, bassissima. Noi non sappiamo il numero di chiamate svolte ogni mese dal call center, perché questa variabile conteggia l'ultima chiamata al cliente, che io immagino (per coerenza su come viene definita duration) sia quella di conversion o abbandono.

Si nota tuttavia che nei mesi di maggior conversion / volumi bassi non c'è una particolare concentrazione di chiamate lunghe, come a dire: minori chiamate, maggiore qualità / lunghezza => conversion. Mi rimane solo l'ausilio grafico per smentire l'associazione tra bassi volumi di ultime chiamate e conversion causa qualità.

g\_month\_duration\_class\_y <- ggplot(bank0, aes(x = month, fill = y)) +   
 geom\_bar(position = "fill") +   
 geom\_hline(yintercept = mean(bank0$y!="yes"), width = 2, col = "black", linetype = 2) +  
 facet\_wrap(~duration\_class) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))  
g\_month\_duration\_class\_y

Mi sembra che a parità di durata delle chiamate l'effetto è che: se la chiamata è breve non si converte in nessun mese (main effect puro), poi il trend tra i mesi è simile per i vari livelli di duration, sempre tenendo conto del main effect.Tendo quindi a pensare che ci sia davvero una stagionalità, e forse il numero basso di chiamate nei mesi in cui la conversion è alta è dovuto al raggiungimento anticipato dei contratti da sottoscrivere. Posso confermare questa ultima tesi facendo un conteggio dei contratti per mese

t\_month\_y2 <- bank0 %>%  
 group\_by (month) %>%  
 summarise(y\_sum = sum(y == "yes"), y\_rate = mean(y == "yes")) %>%  
 arrange(desc(y\_rate))

No, la mia ipotesi era errata. L'ultima cosa che vedo è la distribuzione di duration per mese. Non mi aspetto che sia sbilanciata.

g\_month\_duration\_class <- ggplot(bank0, aes(x = month, fill = duration\_class)) +   
 geom\_bar(position = "fill")   
g\_month\_duration\_class

Non è sbilanciata, marzo è uguale a maggio. Non c'è stata una maggiore dedizione alla telefonata nei mesi di bassi volumi. Io la variabile month la inserisco con riserva, deve esserci della stagionalità.

#set.seed(19121984)  
  
#training\_set <- bank0 %>%  
 # add\_rownames() %>%  
 # sample\_frac(0.8, replace = FALSE)  
  
#test\_set <- bank0 %>%  
 # add\_rownames() %>%  
 # sample\_frac(1, replace = FALSE) %>%  
 #anti\_join(training\_set, by = "rowname")  
#check ok, sono diversi

#set.seed(123)  
#flds <- createFolds(training\_set$y, k = 10, list = TRUE, returnTrain = FALSE)  
#bank\_fold1 <- training\_set[flds[[1]], ]  
#bank\_fold2 <- training\_set[flds[[2]], ]  
#bank\_fold3 <- training\_set[flds[[3]], ]  
#bank\_fold4 <- training\_set[flds[[4]], ]  
#bank\_fold5 <- training\_set[flds[[5]], ]  
#bank\_fold6 <- training\_set[flds[[6]], ]  
#bank\_fold7 <- training\_set[flds[[7]], ]  
#bank\_fold8 <- training\_set[flds[[8]], ]  
#bank\_fold9 <- training\_set[flds[[9]], ]  
#bank\_fold10 <- training\_set[flds[[10]], ]

# Validation set apprach

Creo training\_vsa, validation\_vsa, test\_vsa. Con questa partizione posso attuare il validation approach e modellare e diagnosticare si training, selezionare su validation e testare su test.

set.seed(456)  
training\_set\_vsa <- bank0 %>%  
 add\_rownames() %>%  
 sample\_frac(0.6, replace = FALSE)  
nrow(training\_set\_vsa)

## [1] 27127

test\_and\_val\_set1 <- bank0 %>%  
 add\_rownames() %>%  
 sample\_frac(1, replace = FALSE) %>%  
 anti\_join(training\_set\_vsa, by = "rowname")  
nrow(test\_and\_val\_set1)

## [1] 18084

#check ok, sono diversi  
  
test\_set\_vsa <- test\_and\_val\_set1 %>%  
 sample\_frac(0.5, replace = FALSE)  
nrow(test\_set\_vsa)

## [1] 9042

validation\_set\_vsa <- test\_and\_val\_set1 %>%  
 sample\_frac(1, replace = FALSE) %>%  
 anti\_join(test\_set\_vsa, by = "rowname")  
nrow(validation\_set\_vsa)

## [1] 9042

validation\_set\_vsa[validation\_set\_vsa$rowname == test\_set\_vsa$rowname,] #check ok, tutte righe diverse

## [1] rowname age job marital   
## [5] education default balance housing   
## [9] loan contact day month   
## [13] duration campaign pdays previous   
## [17] poutcome y age\_num age\_class   
## [21] balance\_class duration\_class campaign\_class c\_class   
## [25] pdays\_class previous\_class  
## <0 rows> (or 0-length row.names)

Abbiamo quindi training\_set\_vsa, validation\_set\_vsa, test\_set\_vsa.

Da ora userò l'approccio vsa, perché mi permette di modellare molto meglio il modello testando la significatività, escludendo variabili per p.value, diagnosticando multicollinearità e bontà di adattamento.

I paragoni andranno fatti tra i modelli con questo approccio.

Il modello poco propenso all'overfitting e la quantità di osservazioni che non dovrebbe generare bias mi fanno preferire vsa.

# Modello

Selezione solo le variabili con un IV superiore a 0.02, salvo interazioni incluse (<http://support.sas.com/resources/papers/proceedings13/095-2013.pdf>). Ho comunque sperimentato che escludendo un predittore debole (campaign) l'AUC del validation set passava da 0.746 a 0.740.

formula\_glm1 <- formula\_glm1 <- y~age + job + marital + balance + housing + loan + contact + day + month + campaign + pdays + previous + age\*marital + age\*job + loan\*housing

glm1\_vsa <- glm(formula\_glm1, family = "binomial", data = training\_set\_vsa)  
  
summary(glm1\_vsa) #AIC molto basso, è normale la sovrastima delle performance, che non vuol dire overfitting.

##   
## Call:  
## glm(formula = formula\_glm1, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.2863 -0.5093 -0.3887 -0.2308 3.7185   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.522e+00 3.825e-01 -6.593 4.30e-11 \*\*\*  
## age 1.418e-02 8.028e-03 1.766 0.077374 .   
## jobblue-collar 5.674e-01 3.387e-01 1.675 0.093899 .   
## jobentrepreneur 2.979e-01 6.070e-01 0.491 0.623570   
## jobhousemaid -5.048e-01 6.348e-01 -0.795 0.426499   
## jobmanagement 4.977e-01 3.011e-01 1.653 0.098337 .   
## jobretired -2.406e+00 5.893e-01 -4.083 4.44e-05 \*\*\*  
## jobself-employed 5.101e-01 5.014e-01 1.017 0.308927   
## jobservices 3.450e-01 3.914e-01 0.882 0.377961   
## jobstudent 9.660e-01 6.444e-01 1.499 0.133829   
## jobtechnician 1.674e-01 3.267e-01 0.512 0.608478   
## jobunemployed 5.131e-01 4.879e-01 1.052 0.292935   
## jobunknown 1.005e+00 1.137e+00 0.884 0.376923   
## maritalmarried -1.564e-01 2.824e-01 -0.554 0.579622   
## maritalsingle 9.230e-01 3.247e-01 2.842 0.004482 \*\*   
## balance 2.268e-05 5.788e-06 3.918 8.93e-05 \*\*\*  
## housingyes -6.799e-01 5.154e-02 -13.192 < 2e-16 \*\*\*  
## loanyes -6.613e-01 9.934e-02 -6.657 2.80e-11 \*\*\*  
## contacttelephone -3.450e-01 8.311e-02 -4.152 3.30e-05 \*\*\*  
## contactunknown -1.407e+00 8.200e-02 -17.153 < 2e-16 \*\*\*  
## day 3.895e-03 2.788e-03 1.397 0.162489   
## monthfeb 7.392e-01 1.451e-01 5.093 3.52e-07 \*\*\*  
## monthmar 2.147e+00 1.752e-01 12.255 < 2e-16 \*\*\*  
## monthapr 1.023e+00 1.337e-01 7.648 2.03e-14 \*\*\*  
## monthmay 5.559e-01 1.329e-01 4.182 2.89e-05 \*\*\*  
## monthjun 1.308e+00 1.481e-01 8.834 < 2e-16 \*\*\*  
## monthjul 4.385e-01 1.310e-01 3.347 0.000816 \*\*\*  
## monthaug 2.701e-01 1.331e-01 2.029 0.042412 \*   
## monthsep 1.949e+00 1.671e-01 11.662 < 2e-16 \*\*\*  
## monthoct 1.794e+00 1.540e-01 11.651 < 2e-16 \*\*\*  
## monthnov 3.005e-01 1.364e-01 2.203 0.027583 \*   
## monthdec 1.912e+00 2.139e-01 8.941 < 2e-16 \*\*\*  
## campaign -1.005e-01 1.117e-02 -9.003 < 2e-16 \*\*\*  
## pdays 1.006e-03 2.092e-04 4.808 1.53e-06 \*\*\*  
## previous 4.024e-02 9.105e-03 4.419 9.90e-06 \*\*\*  
## age:maritalmarried -1.053e-04 5.810e-03 -0.018 0.985543   
## age:maritalsingle -2.314e-02 7.614e-03 -3.039 0.002374 \*\*   
## age:jobblue-collar -1.923e-02 8.330e-03 -2.309 0.020955 \*   
## age:jobentrepreneur -1.111e-02 1.407e-02 -0.790 0.429559   
## age:jobhousemaid 2.421e-03 1.321e-02 0.183 0.854614   
## age:jobmanagement -1.009e-02 7.329e-03 -1.377 0.168573   
## age:jobretired 3.898e-02 1.020e-02 3.822 0.000132 \*\*\*  
## age:jobself-employed -1.504e-02 1.222e-02 -1.231 0.218255   
## age:jobservices -9.560e-03 9.765e-03 -0.979 0.327594   
## age:jobstudent -2.328e-02 2.326e-02 -1.001 0.316923   
## age:jobtechnician -3.777e-03 8.028e-03 -0.471 0.637997   
## age:jobunemployed -8.776e-03 1.169e-02 -0.751 0.452775   
## age:jobunknown -2.763e-02 2.370e-02 -1.165 0.243820   
## housingyes:loanyes 3.554e-01 1.350e-01 2.633 0.008457 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 16923 on 27078 degrees of freedom  
## AIC: 17021  
##   
## Number of Fisher Scoring iterations: 6

# Diagnostica

## Multicollinearità

cd <- colldiag(model.matrix(glm1\_vsa))  
print(cd)

## Condition  
## Index Variance Decomposition Proportions  
## intercept (Intercept) age jobblue-collar  
## 1 1.000 0.000 0.000 0.000 0.000   
## 2 2.020 0.000 0.000 0.000 0.000   
## 3 2.119 0.000 0.000 0.000 0.001   
## 4 2.246 0.000 0.000 0.000 0.001   
## 5 2.272 0.000 0.000 0.000 0.000   
## 6 2.277 0.000 0.000 0.000 0.000   
## 7 2.286 0.000 0.000 0.000 0.000   
## 8 2.288 0.000 0.000 0.000 0.000   
## 9 2.292 0.000 0.000 0.000 0.000   
## 10 2.294 0.000 0.000 0.000 0.000   
## 11 2.295 0.000 0.000 0.000 0.000   
## 12 2.566 0.000 0.000 0.000 0.000   
## 13 2.636 0.000 0.000 0.000 0.000   
## 14 2.661 0.000 0.000 0.000 0.000   
## 15 2.813 0.000 0.000 0.000 0.001   
## 16 3.154 0.000 0.000 0.000 0.000   
## 17 3.194 0.000 0.000 0.000 0.000   
## 18 3.200 0.000 0.000 0.000 0.000   
## 19 3.225 0.000 0.000 0.000 0.000   
## 20 3.227 0.000 0.000 0.000 0.000   
## 21 3.231 0.000 0.000 0.000 0.000   
## 22 3.260 0.000 0.000 0.000 0.000   
## 23 3.325 0.000 0.000 0.000 0.000   
## 24 3.486 0.000 0.000 0.000 0.000   
## 25 3.630 0.000 0.000 0.000 0.000   
## 26 3.779 0.000 0.000 0.000 0.000   
## 27 4.488 0.000 0.000 0.000 0.000   
## 28 4.655 0.000 0.000 0.000 0.000   
## 29 5.715 0.000 0.000 0.000 0.000   
## 30 6.743 0.000 0.000 0.000 0.004   
## 31 6.967 0.000 0.000 0.000 0.001   
## 32 7.728 0.000 0.000 0.000 0.000   
## 33 8.130 0.000 0.000 0.000 0.001   
## 34 10.191 0.000 0.000 0.003 0.002   
## 35 12.926 0.000 0.000 0.018 0.014   
## 36 19.409 0.000 0.000 0.000 0.000   
## 37 19.559 0.000 0.000 0.000 0.009   
## 38 19.795 0.000 0.000 0.000 0.029   
## 39 20.254 0.000 0.000 0.000 0.035   
## 40 20.298 0.000 0.000 0.000 0.004   
## 41 20.863 0.000 0.000 0.000 0.154   
## 42 21.130 0.000 0.000 0.000 0.020   
## 43 21.262 0.000 0.000 0.000 0.002   
## 44 21.786 0.000 0.000 0.000 0.016   
## 45 24.405 0.000 0.000 0.006 0.004   
## 46 26.456 0.000 0.000 0.000 0.009   
## 47 29.810 0.000 0.000 0.001 0.006   
## 48 60.118 0.000 0.000 0.001 0.380   
## 49 92.839 0.000 0.000 0.972 0.306   
## 50 47418879277428.367 1.000 1.000 0.000 0.000   
## jobentrepreneur jobhousemaid jobmanagement jobretired jobself-employed  
## 1 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000 0.000   
## 3 0.000 0.000 0.000 0.001 0.000   
## 4 0.000 0.000 0.000 0.001 0.000   
## 5 0.000 0.004 0.000 0.001 0.000   
## 6 0.003 0.000 0.000 0.000 0.001   
## 7 0.000 0.000 0.001 0.000 0.000   
## 8 0.002 0.000 0.000 0.000 0.000   
## 9 0.001 0.004 0.000 0.000 0.000   
## 10 0.002 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.000 0.009   
## 12 0.000 0.000 0.000 0.000 0.000   
## 13 0.000 0.000 0.000 0.000 0.000   
## 14 0.000 0.000 0.000 0.001 0.000   
## 15 0.000 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000 0.000   
## 18 0.000 0.000 0.000 0.000 0.000   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000 0.000   
## 30 0.001 0.001 0.003 0.001 0.001   
## 31 0.000 0.000 0.001 0.000 0.000   
## 32 0.000 0.000 0.000 0.000 0.000   
## 33 0.000 0.000 0.001 0.000 0.000   
## 34 0.001 0.001 0.003 0.001 0.001   
## 35 0.004 0.006 0.017 0.003 0.008   
## 36 0.000 0.001 0.002 0.000 0.588   
## 37 0.003 0.006 0.016 0.000 0.114   
## 38 0.013 0.041 0.006 0.001 0.011   
## 39 0.008 0.020 0.231 0.000 0.004   
## 40 0.006 0.026 0.006 0.000 0.000   
## 41 0.011 0.379 0.002 0.000 0.000   
## 42 0.595 0.134 0.000 0.000 0.000   
## 43 0.028 0.004 0.000 0.000 0.000   
## 44 0.094 0.152 0.007 0.004 0.000   
## 45 0.000 0.000 0.003 0.003 0.000   
## 46 0.006 0.009 0.002 0.036 0.001   
## 47 0.001 0.001 0.010 0.710 0.003   
## 48 0.112 0.096 0.355 0.099 0.138   
## 49 0.107 0.113 0.330 0.135 0.118   
## 50 0.000 0.000 0.000 0.000 0.000   
## jobservices jobstudent jobtechnician jobunemployed jobunknown  
## 1 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.001 0.000 0.000 0.000   
## 3 0.000 0.000 0.000 0.000 0.000   
## 4 0.000 0.002 0.001 0.000 0.000   
## 5 0.000 0.000 0.000 0.000 0.004   
## 6 0.000 0.000 0.001 0.003 0.000   
## 7 0.002 0.000 0.001 0.000 0.000   
## 8 0.000 0.000 0.000 0.006 0.000   
## 9 0.000 0.000 0.000 0.000 0.004   
## 10 0.003 0.000 0.000 0.000 0.001   
## 11 0.000 0.000 0.000 0.000 0.000   
## 12 0.000 0.000 0.000 0.000 0.000   
## 13 0.000 0.003 0.000 0.000 0.000   
## 14 0.000 0.000 0.000 0.000 0.000   
## 15 0.000 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000 0.000   
## 18 0.000 0.000 0.000 0.000 0.000   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000 0.000   
## 30 0.002 0.000 0.002 0.001 0.000   
## 31 0.001 0.000 0.002 0.000 0.000   
## 32 0.000 0.000 0.000 0.000 0.000   
## 33 0.000 0.000 0.000 0.000 0.000   
## 34 0.002 0.000 0.003 0.001 0.000   
## 35 0.011 0.000 0.013 0.007 0.001   
## 36 0.010 0.001 0.004 0.116 0.000   
## 37 0.000 0.001 0.001 0.568 0.001   
## 38 0.196 0.011 0.024 0.059 0.002   
## 39 0.002 0.001 0.077 0.000 0.002   
## 40 0.261 0.004 0.156 0.004 0.001   
## 41 0.002 0.002 0.026 0.001 0.001   
## 42 0.000 0.001 0.001 0.000 0.013   
## 43 0.000 0.000 0.000 0.000 0.903   
## 44 0.028 0.065 0.051 0.006 0.012   
## 45 0.001 0.001 0.006 0.000 0.001   
## 46 0.000 0.741 0.000 0.000 0.001   
## 47 0.008 0.056 0.012 0.004 0.000   
## 48 0.238 0.071 0.326 0.115 0.023   
## 49 0.230 0.036 0.294 0.107 0.026   
## 50 0.000 0.000 0.000 0.000 0.000   
## maritalmarried maritalsingle balance housingyes loanyes  
## 1 0.000 0.000 0.001 0.001 0.001   
## 2 0.000 0.001 0.000 0.000 0.000   
## 3 0.000 0.000 0.005 0.004 0.002   
## 4 0.000 0.000 0.000 0.000 0.001   
## 5 0.000 0.000 0.000 0.000 0.001   
## 6 0.000 0.000 0.000 0.000 0.000   
## 7 0.000 0.000 0.001 0.000 0.001   
## 8 0.000 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.000 0.000   
## 12 0.000 0.000 0.002 0.000 0.015   
## 13 0.000 0.001 0.001 0.001 0.008   
## 14 0.000 0.000 0.007 0.000 0.055   
## 15 0.000 0.000 0.000 0.003 0.002   
## 16 0.000 0.000 0.083 0.000 0.000   
## 17 0.000 0.000 0.000 0.000 0.000   
## 18 0.000 0.000 0.024 0.006 0.003   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.003 0.000 0.000   
## 22 0.000 0.000 0.000 0.001 0.000   
## 23 0.000 0.000 0.002 0.010 0.008   
## 24 0.000 0.000 0.049 0.000 0.000   
## 25 0.000 0.000 0.009 0.001 0.001   
## 26 0.000 0.000 0.800 0.000 0.001   
## 27 0.000 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.004 0.007 0.003   
## 29 0.000 0.000 0.000 0.452 0.158   
## 30 0.007 0.004 0.000 0.002 0.000   
## 31 0.000 0.000 0.001 0.009 0.001   
## 32 0.000 0.000 0.000 0.054 0.118   
## 33 0.000 0.000 0.001 0.198 0.480   
## 34 0.001 0.001 0.003 0.202 0.129   
## 35 0.013 0.007 0.001 0.034 0.002   
## 36 0.001 0.004 0.000 0.000 0.000   
## 37 0.001 0.005 0.000 0.002 0.000   
## 38 0.010 0.046 0.000 0.002 0.000   
## 39 0.001 0.004 0.000 0.001 0.000   
## 40 0.003 0.014 0.000 0.001 0.000   
## 41 0.000 0.004 0.000 0.000 0.000   
## 42 0.000 0.002 0.001 0.000 0.000   
## 43 0.000 0.000 0.000 0.001 0.000   
## 44 0.016 0.079 0.000 0.001 0.000   
## 45 0.006 0.001 0.000 0.004 0.003   
## 46 0.013 0.033 0.000 0.000 0.000   
## 47 0.001 0.017 0.001 0.000 0.005   
## 48 0.481 0.359 0.000 0.000 0.000   
## 49 0.443 0.417 0.000 0.002 0.000   
## 50 0.000 0.000 0.000 0.000 0.000   
## contacttelephone contactunknown day monthfeb monthmar monthapr  
## 1 0.001 0.001 0.001 0.000 0.000 0.000   
## 2 0.001 0.000 0.000 0.000 0.000 0.000   
## 3 0.008 0.005 0.000 0.001 0.002 0.000   
## 4 0.003 0.000 0.000 0.000 0.001 0.000   
## 5 0.000 0.001 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.000 0.001 0.000 0.000   
## 7 0.001 0.000 0.000 0.000 0.000 0.000   
## 8 0.000 0.000 0.000 0.001 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.000 0.000 0.000   
## 12 0.012 0.033 0.000 0.003 0.000 0.008   
## 13 0.000 0.000 0.000 0.002 0.000 0.003   
## 14 0.000 0.001 0.000 0.000 0.000 0.001   
## 15 0.047 0.021 0.001 0.000 0.000 0.000   
## 16 0.010 0.000 0.001 0.039 0.001 0.000   
## 17 0.083 0.001 0.001 0.039 0.007 0.136   
## 18 0.001 0.000 0.002 0.069 0.006 0.004   
## 19 0.000 0.000 0.000 0.006 0.003 0.001   
## 20 0.000 0.000 0.000 0.008 0.504 0.006   
## 21 0.001 0.000 0.000 0.000 0.049 0.017   
## 22 0.014 0.001 0.000 0.031 0.083 0.022   
## 23 0.052 0.002 0.000 0.016 0.032 0.038   
## 24 0.019 0.000 0.000 0.036 0.025 0.025   
## 25 0.694 0.002 0.001 0.015 0.002 0.001   
## 26 0.000 0.000 0.001 0.002 0.006 0.003   
## 27 0.000 0.014 0.002 0.005 0.003 0.021   
## 28 0.005 0.002 0.000 0.000 0.000 0.000   
## 29 0.002 0.026 0.008 0.000 0.001 0.007   
## 30 0.001 0.006 0.004 0.001 0.000 0.001   
## 31 0.021 0.494 0.096 0.030 0.005 0.004   
## 32 0.002 0.143 0.559 0.002 0.004 0.021   
## 33 0.000 0.197 0.029 0.001 0.001 0.001   
## 34 0.001 0.027 0.103 0.012 0.006 0.051   
## 35 0.010 0.000 0.003 0.009 0.004 0.013   
## 36 0.000 0.000 0.000 0.000 0.000 0.000   
## 37 0.000 0.000 0.000 0.001 0.000 0.001   
## 38 0.001 0.000 0.000 0.001 0.001 0.001   
## 39 0.000 0.000 0.000 0.001 0.001 0.000   
## 40 0.000 0.000 0.000 0.000 0.000 0.000   
## 41 0.000 0.000 0.000 0.000 0.000 0.000   
## 42 0.000 0.000 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000 0.000 0.000   
## 44 0.000 0.001 0.000 0.002 0.002 0.002   
## 45 0.000 0.019 0.182 0.654 0.246 0.603   
## 46 0.000 0.000 0.000 0.001 0.000 0.003   
## 47 0.006 0.000 0.002 0.006 0.000 0.004   
## 48 0.000 0.000 0.000 0.000 0.000 0.000   
## 49 0.000 0.000 0.001 0.004 0.001 0.003   
## 50 0.000 0.000 0.000 0.000 0.000 0.000   
## monthmay monthjun monthjul monthaug monthsep monthoct monthnov monthdec  
## 1 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 3 0.002 0.000 0.000 0.002 0.003 0.004 0.000 0.002   
## 4 0.000 0.000 0.000 0.002 0.001 0.001 0.001 0.001   
## 5 0.000 0.002 0.000 0.000 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.000 0.001 0.000 0.000 0.001 0.000   
## 7 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 8 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 12 0.001 0.013 0.001 0.000 0.001 0.001 0.003 0.001   
## 13 0.000 0.000 0.002 0.002 0.001 0.002 0.000 0.001   
## 14 0.000 0.000 0.004 0.002 0.001 0.000 0.000 0.001   
## 15 0.004 0.006 0.023 0.006 0.004 0.000 0.000 0.001   
## 16 0.000 0.000 0.002 0.011 0.013 0.048 0.116 0.001   
## 17 0.001 0.000 0.005 0.011 0.005 0.071 0.000 0.015   
## 18 0.016 0.041 0.004 0.000 0.028 0.014 0.006 0.003   
## 19 0.000 0.001 0.000 0.000 0.058 0.032 0.000 0.681   
## 20 0.001 0.001 0.000 0.000 0.149 0.006 0.000 0.009   
## 21 0.000 0.000 0.000 0.001 0.253 0.248 0.019 0.001   
## 22 0.005 0.021 0.016 0.015 0.033 0.034 0.026 0.029   
## 23 0.002 0.001 0.010 0.049 0.027 0.077 0.000 0.010   
## 24 0.000 0.009 0.022 0.000 0.075 0.035 0.006 0.058   
## 25 0.001 0.005 0.023 0.010 0.005 0.021 0.002 0.008   
## 26 0.002 0.001 0.006 0.000 0.006 0.020 0.036 0.010   
## 27 0.000 0.012 0.002 0.001 0.008 0.003 0.001 0.003   
## 28 0.000 0.000 0.004 0.006 0.001 0.004 0.003 0.000   
## 29 0.009 0.009 0.003 0.016 0.003 0.003 0.001 0.002   
## 30 0.000 0.000 0.000 0.001 0.001 0.001 0.000 0.000   
## 31 0.023 0.022 0.017 0.047 0.003 0.002 0.011 0.002   
## 32 0.001 0.017 0.020 0.024 0.001 0.014 0.028 0.002   
## 33 0.008 0.041 0.014 0.002 0.003 0.002 0.003 0.001   
## 34 0.088 0.040 0.063 0.028 0.008 0.015 0.052 0.005   
## 35 0.019 0.016 0.017 0.014 0.005 0.009 0.020 0.003   
## 36 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 37 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.000   
## 38 0.001 0.000 0.000 0.000 0.001 0.001 0.000 0.000   
## 39 0.000 0.000 0.000 0.001 0.001 0.000 0.000 0.000   
## 40 0.000 0.000 0.001 0.000 0.000 0.000 0.001 0.000   
## 41 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 42 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000 0.001 0.001 0.000 0.000   
## 44 0.003 0.002 0.002 0.001 0.001 0.002 0.001 0.000   
## 45 0.795 0.721 0.722 0.727 0.298 0.326 0.646 0.145   
## 46 0.003 0.003 0.004 0.004 0.001 0.001 0.003 0.000   
## 47 0.008 0.010 0.009 0.008 0.000 0.000 0.007 0.000   
## 48 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 49 0.004 0.004 0.004 0.004 0.001 0.002 0.003 0.000   
## 50 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## campaign pdays previous age:maritalmarried age:maritalsingle  
## 1 0.002 0.001 0.000 0.000 0.000   
## 2 0.000 0.001 0.001 0.000 0.001   
## 3 0.000 0.000 0.001 0.000 0.000   
## 4 0.000 0.002 0.001 0.000 0.000   
## 5 0.000 0.003 0.003 0.000 0.000   
## 6 0.000 0.000 0.000 0.000 0.000   
## 7 0.000 0.000 0.000 0.000 0.000   
## 8 0.000 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.000 0.000   
## 12 0.001 0.049 0.056 0.000 0.000   
## 13 0.002 0.036 0.037 0.000 0.001   
## 14 0.000 0.012 0.016 0.000 0.001   
## 15 0.010 0.029 0.045 0.000 0.000   
## 16 0.011 0.005 0.007 0.000 0.000   
## 17 0.000 0.000 0.009 0.000 0.000   
## 18 0.000 0.000 0.004 0.000 0.000   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000 0.000   
## 22 0.000 0.002 0.006 0.000 0.000   
## 23 0.002 0.001 0.019 0.000 0.000   
## 24 0.008 0.021 0.223 0.000 0.000   
## 25 0.003 0.000 0.008 0.000 0.000   
## 26 0.000 0.000 0.021 0.000 0.000   
## 27 0.049 0.556 0.487 0.000 0.000   
## 28 0.804 0.051 0.047 0.000 0.000   
## 29 0.019 0.035 0.003 0.000 0.000   
## 30 0.002 0.001 0.000 0.008 0.007   
## 31 0.032 0.124 0.001 0.001 0.001   
## 32 0.032 0.008 0.000 0.000 0.000   
## 33 0.010 0.050 0.001 0.001 0.002   
## 34 0.002 0.010 0.000 0.003 0.003   
## 35 0.001 0.000 0.000 0.008 0.013   
## 36 0.000 0.000 0.000 0.001 0.007   
## 37 0.000 0.000 0.000 0.001 0.007   
## 38 0.000 0.000 0.000 0.011 0.071   
## 39 0.000 0.000 0.000 0.001 0.007   
## 40 0.000 0.000 0.000 0.003 0.022   
## 41 0.000 0.000 0.000 0.001 0.005   
## 42 0.000 0.000 0.000 0.001 0.002   
## 43 0.000 0.000 0.000 0.000 0.000   
## 44 0.000 0.000 0.000 0.018 0.107   
## 45 0.008 0.001 0.000 0.010 0.003   
## 46 0.000 0.000 0.000 0.014 0.051   
## 47 0.000 0.000 0.000 0.001 0.022   
## 48 0.000 0.000 0.000 0.478 0.307   
## 49 0.000 0.000 0.000 0.438 0.358   
## 50 0.000 0.000 0.000 0.000 0.000   
## age:jobblue-collar age:jobentrepreneur age:jobhousemaid  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.000   
## 3 0.001 0.000 0.000   
## 4 0.001 0.000 0.000   
## 5 0.000 0.000 0.003   
## 6 0.000 0.003 0.000   
## 7 0.000 0.000 0.000   
## 8 0.000 0.002 0.000   
## 9 0.000 0.001 0.003   
## 10 0.000 0.002 0.000   
## 11 0.000 0.000 0.000   
## 12 0.000 0.000 0.000   
## 13 0.000 0.000 0.000   
## 14 0.000 0.000 0.000   
## 15 0.001 0.000 0.000   
## 16 0.000 0.000 0.000   
## 17 0.000 0.000 0.000   
## 18 0.000 0.000 0.000   
## 19 0.000 0.000 0.000   
## 20 0.000 0.000 0.000   
## 21 0.000 0.000 0.000   
## 22 0.000 0.000 0.000   
## 23 0.000 0.000 0.000   
## 24 0.000 0.000 0.000   
## 25 0.000 0.000 0.000   
## 26 0.000 0.000 0.000   
## 27 0.000 0.000 0.000   
## 28 0.000 0.000 0.000   
## 29 0.000 0.000 0.000   
## 30 0.004 0.001 0.001   
## 31 0.002 0.001 0.001   
## 32 0.000 0.000 0.000   
## 33 0.001 0.000 0.000   
## 34 0.002 0.001 0.001   
## 35 0.005 0.001 0.002   
## 36 0.000 0.000 0.001   
## 37 0.009 0.002 0.005   
## 38 0.028 0.012 0.037   
## 39 0.033 0.007 0.018   
## 40 0.004 0.006 0.024   
## 41 0.152 0.011 0.354   
## 42 0.020 0.579 0.126   
## 43 0.002 0.027 0.004   
## 44 0.016 0.093 0.144   
## 45 0.007 0.001 0.001   
## 46 0.009 0.006 0.009   
## 47 0.006 0.001 0.001   
## 48 0.379 0.122 0.119   
## 49 0.317 0.120 0.144   
## 50 0.000 0.000 0.000   
## age:jobmanagement age:jobretired age:jobself-employed age:jobservices  
## 1 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000   
## 3 0.000 0.001 0.000 0.000   
## 4 0.000 0.001 0.000 0.000   
## 5 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.001 0.000   
## 7 0.001 0.000 0.000 0.002   
## 8 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.003   
## 11 0.000 0.000 0.009 0.000   
## 12 0.000 0.000 0.000 0.000   
## 13 0.000 0.000 0.000 0.000   
## 14 0.000 0.000 0.000 0.000   
## 15 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000   
## 18 0.000 0.000 0.000 0.000   
## 19 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000   
## 30 0.004 0.001 0.001 0.002   
## 31 0.002 0.000 0.001 0.001   
## 32 0.000 0.000 0.000 0.000   
## 33 0.001 0.000 0.000 0.001   
## 34 0.003 0.001 0.001 0.002   
## 35 0.006 0.000 0.004 0.004   
## 36 0.002 0.000 0.578 0.010   
## 37 0.016 0.000 0.113 0.000   
## 38 0.006 0.000 0.011 0.201   
## 39 0.224 0.000 0.004 0.002   
## 40 0.005 0.000 0.000 0.264   
## 41 0.002 0.000 0.000 0.002   
## 42 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000   
## 44 0.006 0.002 0.000 0.027   
## 45 0.006 0.001 0.001 0.003   
## 46 0.003 0.024 0.001 0.000   
## 47 0.009 0.575 0.003 0.008   
## 48 0.358 0.163 0.144 0.235   
## 49 0.344 0.229 0.126 0.229   
## 50 0.000 0.000 0.000 0.000   
## age:jobstudent age:jobtechnician age:jobunemployed age:jobunknown  
## 1 0.000 0.000 0.000 0.000   
## 2 0.001 0.000 0.000 0.000   
## 3 0.000 0.000 0.000 0.000   
## 4 0.002 0.001 0.000 0.000   
## 5 0.000 0.000 0.000 0.004   
## 6 0.000 0.001 0.003 0.000   
## 7 0.000 0.001 0.000 0.000   
## 8 0.000 0.000 0.006 0.000   
## 9 0.000 0.000 0.000 0.004   
## 10 0.000 0.000 0.000 0.001   
## 11 0.000 0.000 0.000 0.000   
## 12 0.000 0.000 0.000 0.000   
## 13 0.003 0.000 0.000 0.000   
## 14 0.000 0.000 0.000 0.000   
## 15 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000   
## 18 0.000 0.000 0.000 0.000   
## 19 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000   
## 30 0.000 0.003 0.001 0.000   
## 31 0.000 0.002 0.001 0.000   
## 32 0.000 0.000 0.000 0.000   
## 33 0.000 0.001 0.000 0.000   
## 34 0.001 0.002 0.001 0.000   
## 35 0.001 0.004 0.003 0.000   
## 36 0.002 0.004 0.114 0.000   
## 37 0.001 0.001 0.560 0.001   
## 38 0.016 0.024 0.058 0.002   
## 39 0.002 0.078 0.000 0.002   
## 40 0.006 0.155 0.004 0.001   
## 41 0.002 0.026 0.001 0.001   
## 42 0.001 0.001 0.000 0.013   
## 43 0.000 0.000 0.000 0.885   
## 44 0.086 0.049 0.006 0.012   
## 45 0.001 0.009 0.001 0.002   
## 46 0.767 0.001 0.000 0.001   
## 47 0.049 0.011 0.003 0.000   
## 48 0.039 0.325 0.121 0.032   
## 49 0.018 0.299 0.114 0.036   
## 50 0.000 0.000 0.000 0.000   
## housingyes:loanyes  
## 1 0.000   
## 2 0.000   
## 3 0.004   
## 4 0.001   
## 5 0.001   
## 6 0.000   
## 7 0.001   
## 8 0.000   
## 9 0.000   
## 10 0.000   
## 11 0.000   
## 12 0.017   
## 13 0.007   
## 14 0.054   
## 15 0.012   
## 16 0.000   
## 17 0.001   
## 18 0.001   
## 19 0.000   
## 20 0.000   
## 21 0.000   
## 22 0.001   
## 23 0.007   
## 24 0.005   
## 25 0.005   
## 26 0.002   
## 27 0.000   
## 28 0.002   
## 29 0.086   
## 30 0.000   
## 31 0.005   
## 32 0.132   
## 33 0.528   
## 34 0.121   
## 35 0.002   
## 36 0.000   
## 37 0.000   
## 38 0.000   
## 39 0.000   
## 40 0.000   
## 41 0.000   
## 42 0.000   
## 43 0.000   
## 44 0.000   
## 45 0.001   
## 46 0.000   
## 47 0.003   
## 48 0.000   
## 49 0.000   
## 50 0.000

Ci sono due indici (il 48 e il 49) maggiori di 30, ma nessuna vdp maggiore di 0.5 (anche se un paio ci vanno vicino, e qualche problema di multicollinearità non mi stupisce).

## Bontà adattamento modello

hl\_glm1 <- hoslem.test(training\_set\_vsa$y =="yes", fitted(glm1\_vsa), g=10)  
hl\_glm1

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: training\_set\_vsa$y == "yes", fitted(glm1\_vsa)  
## X-squared = 109.5292, df = 8, p-value < 2.2e-16

Il test, se rigettato, rigetta l'ipotesi di buon adattamento. Qui non si può rigettare. è vero che c'è il problema che il numero di gruppi non dovrebbe essere inferiore a p+1 (<http://thestatsgeek.com/2014/02/16/the-hosmer-lemeshow-goodness-of-fit-test-for-logistic-regression/>), e io con le variabili dummy ho molte p. Ma anche se i gruppi sono 100, il p.value è significativo, quindi il buon adattamento va rigettato.

# Stima performance predittive

glm1\_vsa\_predictions <- predict(glm1\_vsa, validation\_set\_vsa, type="response")  
  
AUC\_glm1\_vsa <- roc(validation\_set\_vsa$y, glm1\_vsa\_predictions, levels=c("no", "yes"))  
AUC\_glm1\_vsa$auc

## Area under the curve: 0.746

AUC del modello sul validation set è 0.746. Con quella competerà con gli altri modelli, se la diagnostica non me lo fa cambiare.

Voglio sperimentare anche la AUCPR, che forse è miglior indicatore per classi molto sbilanciate (<http://stats.stackexchange.com/questions/7207/roc-vs-precision-and-recall-curves>)

AUCPR\_glm1\_vsa <- pr.curve(glm1\_vsa\_predictions, weights.class0 = validation\_set\_vsa$y == "yes", curve=T)  
AUCPR\_glm1\_vsa

##   
## Precision-recall curve  
##   
## Area under curve (Integral):  
## 0.3200601   
##   
## Area under curve (Davis & Goadrich):  
## 0.3200454   
##   
## Curve for scores from 6.851199e-05 to 0.8358319   
## ( can be plotted with plot(x) )

plot(AUCPR\_glm1\_vsa)

Prima stepwise selection su tutto il training

glm2\_beforebackstep\_training <- glm(y~age + job + marital + education + default + balance + housing + loan + contact + pdays + day + month + campaign + previous, family = "binomial", data = training\_set\_vsa)  
  
glm2\_backstep\_selection <- stepAIC(object = glm2\_beforebackstep\_training, direction = "backward", scope = c(upper = ~age + job + marital + education + default + balance + housing + loan + contact + day + month + campaign + pdays + previous, lower = ~1))

## Start: AIC=17050.53  
## y ~ age + job + marital + education + default + balance + housing +   
## loan + contact + pdays + day + month + campaign + previous  
##   
## Df Deviance AIC  
## - default 1 16973 17049  
## - day 1 16974 17050  
## <none> 16972 17050  
## - age 1 16978 17054  
## - balance 1 16986 17062  
## - pdays 1 16993 17069  
## - previous 1 16996 17072  
## - education 3 17000 17072  
## - marital 2 17001 17075  
## - job 11 17025 17081  
## - loan 1 17029 17105  
## - campaign 1 17078 17154  
## - housing 1 17156 17232  
## - contact 2 17306 17380  
## - month 11 17628 17684  
##   
## Step: AIC=17048.71  
## y ~ age + job + marital + education + balance + housing + loan +   
## contact + pdays + day + month + campaign + previous  
##   
## Df Deviance AIC  
## - day 1 16975 17049  
## <none> 16973 17049  
## - age 1 16978 17052  
## - balance 1 16986 17060  
## - pdays 1 16993 17067  
## - previous 1 16996 17070  
## - education 3 17000 17070  
## - marital 2 17001 17073  
## - job 11 17025 17079  
## - loan 1 17030 17104  
## - campaign 1 17078 17152  
## - housing 1 17156 17230  
## - contact 2 17307 17379  
## - month 11 17630 17684  
##   
## Step: AIC=17048.57  
## y ~ age + job + marital + education + balance + housing + loan +   
## contact + pdays + month + campaign + previous  
##   
## Df Deviance AIC  
## <none> 16975 17049  
## - age 1 16980 17052  
## - balance 1 16988 17060  
## - pdays 1 16994 17066  
## - previous 1 16998 17070  
## - education 3 17002 17070  
## - marital 2 17003 17073  
## - job 11 17027 17079  
## - loan 1 17032 17104  
## - campaign 1 17078 17150  
## - housing 1 17160 17232  
## - contact 2 17307 17377  
## - month 11 17636 17688

glm2\_backstep\_selection$anova

## Stepwise Model Path   
## Analysis of Deviance Table  
##   
## Initial Model:  
## y ~ age + job + marital + education + default + balance + housing +   
## loan + contact + pdays + day + month + campaign + previous  
##   
## Final Model:  
## y ~ age + job + marital + education + balance + housing + loan +   
## contact + pdays + month + campaign + previous  
##   
##   
## Step Df Deviance Resid. Df Resid. Dev AIC  
## 1 27088 16972.53 17050.53  
## 2 - default 1 0.1882962 27089 16972.71 17048.71  
## 3 - day 1 1.8527483 27090 16974.57 17048.57

glm2\_beforeforwardstep\_training <- glm(y~1, family = "binomial", data = training\_set\_vsa)  
  
glm2\_forwardstep\_selection <- stepAIC(object = glm2\_beforeforwardstep\_training, direction = "forward", scope = c(upper = ~age + job + marital + education + default + balance + housing + loan + contact + day + month + campaign + pdays + previous, lower = ~1))

## Start: AIC=19591.18  
## y ~ 1  
##   
## Df Deviance AIC  
## + month 11 18206 18230  
## + contact 2 18831 18837  
## + housing 1 19017 19021  
## + job 11 19152 19176  
## + pdays 1 19346 19350  
## + campaign 1 19379 19383  
## + previous 1 19385 19389  
## + loan 1 19439 19443  
## + education 3 19443 19451  
## + marital 2 19493 19499  
## + balance 1 19521 19525  
## + age 1 19559 19563  
## + day 1 19568 19572  
## + default 1 19576 19580  
## <none> 19589 19591  
##   
## Step: AIC=18229.93  
## y ~ month  
##   
## Df Deviance AIC  
## + contact 2 17647 17675  
## + housing 1 17919 17945  
## + job 11 18014 18060  
## + campaign 1 18067 18093  
## + pdays 1 18080 18106  
## + previous 1 18104 18130  
## + loan 1 18109 18135  
## + marital 2 18128 18156  
## + education 3 18128 18158  
## + balance 1 18173 18199  
## + day 1 18201 18227  
## + default 1 18202 18228  
## + age 1 18202 18228  
## <none> 18206 18230  
##   
## Step: AIC=17675.22  
## y ~ month + contact  
##   
## Df Deviance AIC  
## + housing 1 17390 17420  
## + job 11 17493 17543  
## + campaign 1 17529 17559  
## + loan 1 17563 17593  
## + marital 2 17593 17625  
## + education 3 17594 17628  
## + previous 1 17606 17636  
## + balance 1 17615 17645  
## + pdays 1 17620 17650  
## + age 1 17638 17668  
## + default 1 17645 17675  
## <none> 17647 17675  
## + day 1 17646 17676  
##   
## Step: AIC=17419.59  
## y ~ month + contact + housing  
##   
## Df Deviance AIC  
## + campaign 1 17279 17311  
## + job 11 17295 17347  
## + loan 1 17319 17351  
## + pdays 1 17336 17368  
## + previous 1 17336 17368  
## + education 3 17345 17381  
## + marital 2 17349 17383  
## + balance 1 17366 17398  
## + default 1 17387 17419  
## <none> 17390 17420  
## + age 1 17388 17420  
## + day 1 17390 17422  
##   
## Step: AIC=17310.76  
## y ~ month + contact + housing + campaign  
##   
## Df Deviance AIC  
## + loan 1 17207 17241  
## + job 11 17190 17244  
## + previous 1 17222 17256  
## + pdays 1 17230 17264  
## + education 3 17233 17271  
## + marital 2 17239 17275  
## + balance 1 17256 17290  
## + day 1 17276 17310  
## + default 1 17277 17311  
## <none> 17279 17311  
## + age 1 17278 17312  
##   
## Step: AIC=17241.1  
## y ~ month + contact + housing + campaign + loan  
##   
## Df Deviance AIC  
## + job 11 17125 17181  
## + previous 1 17149 17185  
## + pdays 1 17157 17193  
## + education 3 17165 17205  
## + marital 2 17170 17208  
## + balance 1 17189 17225  
## + day 1 17205 17241  
## <none> 17207 17241  
## + age 1 17206 17242  
## + default 1 17206 17242  
##   
## Step: AIC=17181.18  
## y ~ month + contact + housing + campaign + loan + job  
##   
## Df Deviance AIC  
## + previous 1 17070 17128  
## + pdays 1 17074 17132  
## + education 3 17093 17155  
## + marital 2 17098 17158  
## + balance 1 17109 17167  
## <none> 17125 17181  
## + day 1 17124 17182  
## + default 1 17125 17183  
## + age 1 17125 17183  
##   
## Step: AIC=17127.7  
## y ~ month + contact + housing + campaign + loan + job + previous  
##   
## Df Deviance AIC  
## + education 3 17037 17101  
## + marital 2 17042 17104  
## + pdays 1 17050 17110  
## + balance 1 17054 17114  
## <none> 17070 17128  
## + day 1 17068 17128  
## + default 1 17069 17129  
## + age 1 17070 17130  
##   
## Step: AIC=17101.15  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education  
##   
## Df Deviance AIC  
## + marital 2 17015 17083  
## + pdays 1 17017 17083  
## + balance 1 17023 17089  
## <none> 17037 17101  
## + day 1 17035 17101  
## + age 1 17036 17102  
## + default 1 17037 17103  
##   
## Step: AIC=17082.94  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education + marital  
##   
## Df Deviance AIC  
## + pdays 1 16995 17065  
## + balance 1 17000 17070  
## + age 1 17008 17078  
## <none> 17015 17083  
## + day 1 17013 17083  
## + default 1 17014 17084  
##   
## Step: AIC=17064.98  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education + marital + pdays  
##   
## Df Deviance AIC  
## + balance 1 16980 17052  
## + age 1 16988 17060  
## <none> 16995 17065  
## + day 1 16993 17065  
## + default 1 16995 17067  
##   
## Step: AIC=17051.77  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education + marital + pdays + balance  
##   
## Df Deviance AIC  
## + age 1 16975 17049  
## <none> 16980 17052  
## + day 1 16978 17052  
## + default 1 16980 17054  
##   
## Step: AIC=17048.57  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education + marital + pdays + balance + age  
##   
## Df Deviance AIC  
## <none> 16975 17049  
## + day 1 16973 17049  
## + default 1 16974 17050

glm2\_forwardstep\_selection$anova

## Stepwise Model Path   
## Analysis of Deviance Table  
##   
## Initial Model:  
## y ~ 1  
##   
## Final Model:  
## y ~ month + contact + housing + campaign + loan + job + previous +   
## education + marital + pdays + balance + age  
##   
##   
## Step Df Deviance Resid. Df Resid. Dev AIC  
## 1 27126 19589.18 19591.18  
## 2 + month 11 1383.247932 27115 18205.93 18229.93  
## 3 + contact 2 558.713170 27113 17647.22 17675.22  
## 4 + housing 1 257.632941 27112 17389.59 17419.59  
## 5 + campaign 1 110.829372 27111 17278.76 17310.76  
## 6 + loan 1 71.662214 27110 17207.10 17241.10  
## 7 + job 11 81.914125 27099 17125.18 17181.18  
## 8 + previous 1 55.476363 27098 17069.70 17127.70  
## 9 + education 3 32.559086 27095 17037.15 17101.15  
## 10 + marital 2 22.200738 27093 17014.94 17082.94  
## 11 + pdays 1 19.962529 27092 16994.98 17064.98  
## 12 + balance 1 15.207435 27091 16979.77 17051.77  
## 13 + age 1 5.207865 27090 16974.57 17048.57

Allora, la backward come final model presenta y ~ age + job + marital + education + balance + housing + loan + contact + pdays + month + campaign + previous e un AIC di 17048.57.

La forward y ~ month + contact + housing + campaign + loan + job + previous + education + marital + pdays + balance + age con AIC di 17048.57. Identico!

# Diagnostica

step\_formula <- y ~ age + job + marital + education + balance + housing + loan + contact + pdays + month + campaign + previous  
  
glm2\_vsa <- glm(step\_formula, family ="binomial", data = training\_set\_vsa)

## Multicollinearità

cd <- colldiag(model.matrix(glm2\_vsa))  
print(cd)

## Condition  
## Index Variance Decomposition Proportions  
## intercept (Intercept) age jobblue-collar  
## 1 1.000 0.000 0.000 0.000 0.001   
## 2 2.130 0.000 0.000 0.000 0.014   
## 3 2.326 0.000 0.000 0.000 0.002   
## 4 2.423 0.000 0.000 0.000 0.000   
## 5 2.536 0.000 0.000 0.000 0.000   
## 6 2.587 0.000 0.000 0.000 0.008   
## 7 2.721 0.000 0.000 0.000 0.000   
## 8 2.771 0.000 0.000 0.000 0.017   
## 9 2.815 0.000 0.000 0.000 0.000   
## 10 2.848 0.000 0.000 0.000 0.006   
## 11 2.879 0.000 0.000 0.000 0.010   
## 12 2.902 0.000 0.000 0.000 0.029   
## 13 2.925 0.000 0.000 0.000 0.005   
## 14 2.930 0.000 0.000 0.000 0.002   
## 15 2.933 0.000 0.000 0.000 0.000   
## 16 2.940 0.000 0.000 0.000 0.002   
## 17 2.951 0.000 0.000 0.000 0.003   
## 18 2.965 0.000 0.000 0.000 0.000   
## 19 3.047 0.000 0.000 0.000 0.003   
## 20 3.079 0.000 0.000 0.000 0.014   
## 21 3.171 0.000 0.000 0.000 0.011   
## 22 3.206 0.000 0.000 0.000 0.023   
## 23 3.280 0.000 0.000 0.000 0.002   
## 24 3.287 0.000 0.000 0.000 0.028   
## 25 3.402 0.000 0.000 0.000 0.002   
## 26 3.511 0.000 0.000 0.000 0.034   
## 27 3.588 0.000 0.000 0.000 0.000   
## 28 4.089 0.000 0.000 0.000 0.000   
## 29 4.323 0.000 0.000 0.001 0.007   
## 30 4.517 0.000 0.000 0.000 0.083   
## 31 5.704 0.000 0.000 0.001 0.012   
## 32 6.639 0.000 0.000 0.001 0.009   
## 33 8.636 0.000 0.000 0.003 0.053   
## 34 8.843 0.000 0.000 0.013 0.140   
## 35 10.729 0.000 0.000 0.123 0.419   
## 36 16.454 0.000 0.000 0.465 0.006   
## 37 24.971 0.000 0.000 0.392 0.055   
## 38 43115449283739.594 1.000 1.000 0.000 0.000   
## jobentrepreneur jobhousemaid jobmanagement jobretired jobself-employed  
## 1 0.000 0.000 0.001 0.000 0.000   
## 2 0.000 0.000 0.019 0.001 0.002   
## 3 0.003 0.004 0.009 0.000 0.001   
## 4 0.000 0.019 0.014 0.035 0.000   
## 5 0.000 0.005 0.001 0.036 0.001   
## 6 0.015 0.000 0.005 0.011 0.001   
## 7 0.049 0.001 0.002 0.024 0.004   
## 8 0.004 0.000 0.004 0.006 0.014   
## 9 0.006 0.000 0.003 0.063 0.010   
## 10 0.056 0.026 0.008 0.001 0.031   
## 11 0.000 0.187 0.000 0.012 0.014   
## 12 0.039 0.030 0.001 0.002 0.068   
## 13 0.167 0.097 0.004 0.012 0.058   
## 14 0.000 0.039 0.000 0.004 0.005   
## 15 0.000 0.006 0.000 0.003 0.011   
## 16 0.050 0.000 0.002 0.000 0.344   
## 17 0.001 0.021 0.001 0.004 0.020   
## 18 0.146 0.128 0.006 0.027 0.067   
## 19 0.065 0.010 0.000 0.010 0.006   
## 20 0.068 0.022 0.004 0.019 0.033   
## 21 0.000 0.033 0.000 0.002 0.000   
## 22 0.021 0.023 0.000 0.025 0.001   
## 23 0.001 0.044 0.000 0.013 0.000   
## 24 0.000 0.011 0.000 0.156 0.000   
## 25 0.000 0.003 0.003 0.028 0.000   
## 26 0.020 0.009 0.001 0.029 0.000   
## 27 0.000 0.000 0.001 0.009 0.002   
## 28 0.000 0.000 0.000 0.000 0.000   
## 29 0.004 0.000 0.007 0.002 0.002   
## 30 0.003 0.002 0.131 0.007 0.005   
## 31 0.001 0.000 0.003 0.006 0.000   
## 32 0.001 0.003 0.003 0.001 0.001   
## 33 0.108 0.014 0.395 0.034 0.124   
## 34 0.030 0.034 0.069 0.016 0.038   
## 35 0.130 0.213 0.285 0.350 0.129   
## 36 0.000 0.000 0.000 0.047 0.000   
## 37 0.008 0.015 0.016 0.003 0.010   
## 38 0.000 0.000 0.000 0.000 0.000   
## jobservices jobstudent jobtechnician jobunemployed jobunknown  
## 1 0.001 0.000 0.001 0.000 0.000   
## 2 0.008 0.002 0.000 0.000 0.000   
## 3 0.004 0.011 0.000 0.001 0.002   
## 4 0.001 0.009 0.000 0.001 0.003   
## 5 0.005 0.037 0.007 0.000 0.080   
## 6 0.006 0.115 0.037 0.003 0.004   
## 7 0.006 0.032 0.052 0.009 0.010   
## 8 0.006 0.002 0.000 0.187 0.048   
## 9 0.005 0.025 0.011 0.023 0.178   
## 10 0.007 0.012 0.012 0.089 0.000   
## 11 0.044 0.002 0.003 0.034 0.004   
## 12 0.073 0.028 0.004 0.010 0.038   
## 13 0.119 0.032 0.012 0.000 0.067   
## 14 0.000 0.001 0.002 0.013 0.001   
## 15 0.004 0.000 0.000 0.004 0.000   
## 16 0.009 0.002 0.001 0.004 0.003   
## 17 0.001 0.014 0.003 0.033 0.020   
## 18 0.033 0.000 0.006 0.009 0.002   
## 19 0.073 0.028 0.036 0.031 0.057   
## 20 0.004 0.018 0.001 0.182 0.026   
## 21 0.005 0.076 0.038 0.063 0.033   
## 22 0.001 0.010 0.031 0.010 0.142   
## 23 0.008 0.027 0.014 0.014 0.136   
## 24 0.014 0.030 0.002 0.030 0.059   
## 25 0.001 0.012 0.003 0.000 0.007   
## 26 0.001 0.268 0.025 0.000 0.008   
## 27 0.001 0.004 0.006 0.007 0.004   
## 28 0.000 0.002 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000 0.000   
## 30 0.126 0.012 0.041 0.010 0.003   
## 31 0.005 0.001 0.007 0.000 0.001   
## 32 0.002 0.000 0.007 0.001 0.000   
## 33 0.130 0.049 0.263 0.052 0.014   
## 34 0.056 0.041 0.075 0.030 0.010   
## 35 0.208 0.046 0.274 0.128 0.038   
## 36 0.002 0.016 0.001 0.000 0.000   
## 37 0.029 0.032 0.026 0.022 0.002   
## 38 0.000 0.000 0.000 0.000 0.000   
## maritalmarried maritalsingle educationsecondary educationtertiary  
## 1 0.001 0.001 0.001 0.001   
## 2 0.000 0.001 0.005 0.019   
## 3 0.001 0.001 0.003 0.008   
## 4 0.004 0.012 0.003 0.010   
## 5 0.000 0.000 0.002 0.001   
## 6 0.008 0.040 0.005 0.001   
## 7 0.001 0.005 0.002 0.001   
## 8 0.000 0.001 0.003 0.000   
## 9 0.001 0.002 0.000 0.001   
## 10 0.000 0.001 0.002 0.001   
## 11 0.000 0.000 0.000 0.000   
## 12 0.000 0.001 0.000 0.000   
## 13 0.000 0.000 0.000 0.000   
## 14 0.001 0.001 0.000 0.000   
## 15 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000   
## 17 0.000 0.002 0.000 0.000   
## 18 0.000 0.000 0.000 0.000   
## 19 0.001 0.002 0.000 0.000   
## 20 0.000 0.000 0.000 0.000   
## 21 0.003 0.007 0.000 0.002   
## 22 0.000 0.002 0.004 0.000   
## 23 0.001 0.000 0.002 0.000   
## 24 0.002 0.003 0.000 0.001   
## 25 0.004 0.009 0.000 0.001   
## 26 0.023 0.095 0.000 0.001   
## 27 0.003 0.002 0.002 0.001   
## 28 0.000 0.000 0.000 0.000   
## 29 0.002 0.004 0.010 0.000   
## 30 0.001 0.000 0.125 0.100   
## 31 0.004 0.001 0.003 0.002   
## 32 0.016 0.003 0.014 0.006   
## 33 0.002 0.018 0.406 0.625   
## 34 0.767 0.533 0.044 0.020   
## 35 0.061 0.015 0.250 0.108   
## 36 0.020 0.078 0.021 0.019   
## 37 0.070 0.160 0.092 0.071   
## 38 0.000 0.000 0.000 0.000   
## educationunknown balance housingyes loanyes contacttelephone  
## 1 0.000 0.002 0.003 0.002 0.001   
## 2 0.000 0.008 0.004 0.002 0.006   
## 3 0.000 0.002 0.002 0.000 0.005   
## 4 0.000 0.000 0.005 0.013 0.074   
## 5 0.099 0.014 0.004 0.041 0.011   
## 6 0.007 0.006 0.002 0.000 0.000   
## 7 0.030 0.001 0.001 0.022 0.021   
## 8 0.065 0.018 0.001 0.001 0.001   
## 9 0.049 0.001 0.000 0.003 0.008   
## 10 0.001 0.043 0.000 0.000 0.000   
## 11 0.006 0.001 0.001 0.002 0.022   
## 12 0.002 0.008 0.000 0.014 0.008   
## 13 0.000 0.000 0.000 0.000 0.002   
## 14 0.006 0.002 0.000 0.005 0.000   
## 15 0.000 0.002 0.000 0.001 0.001   
## 16 0.000 0.007 0.000 0.000 0.004   
## 17 0.003 0.013 0.000 0.004 0.000   
## 18 0.000 0.014 0.001 0.017 0.017   
## 19 0.013 0.001 0.000 0.059 0.004   
## 20 0.000 0.002 0.001 0.028 0.003   
## 21 0.019 0.016 0.005 0.020 0.145   
## 22 0.325 0.003 0.002 0.032 0.094   
## 23 0.048 0.002 0.000 0.108 0.267   
## 24 0.022 0.077 0.000 0.018 0.014   
## 25 0.001 0.447 0.000 0.041 0.154   
## 26 0.025 0.077 0.001 0.001 0.002   
## 27 0.008 0.213 0.005 0.546 0.087   
## 28 0.000 0.000 0.000 0.000 0.001   
## 29 0.002 0.006 0.001 0.012 0.007   
## 30 0.019 0.002 0.000 0.000 0.000   
## 31 0.001 0.004 0.899 0.001 0.003   
## 32 0.004 0.001 0.024 0.001 0.018   
## 33 0.156 0.000 0.001 0.001 0.004   
## 34 0.008 0.000 0.000 0.000 0.000   
## 35 0.067 0.002 0.004 0.001 0.007   
## 36 0.000 0.001 0.022 0.000 0.008   
## 37 0.013 0.002 0.011 0.002 0.002   
## 38 0.000 0.000 0.000 0.000 0.000   
## contactunknown pdays monthfeb monthmar monthapr monthmay monthjun  
## 1 0.001 0.001 0.000 0.000 0.000 0.000 0.000   
## 2 0.022 0.005 0.003 0.004 0.001 0.003 0.003   
## 3 0.012 0.088 0.006 0.001 0.012 0.000 0.010   
## 4 0.013 0.008 0.000 0.000 0.000 0.005 0.001   
## 5 0.007 0.001 0.000 0.007 0.000 0.000 0.015   
## 6 0.001 0.014 0.003 0.002 0.001 0.000 0.000   
## 7 0.000 0.004 0.002 0.000 0.000 0.000 0.000   
## 8 0.003 0.002 0.040 0.001 0.010 0.001 0.010   
## 9 0.002 0.002 0.018 0.064 0.004 0.001 0.000   
## 10 0.000 0.002 0.060 0.001 0.000 0.000 0.000   
## 11 0.000 0.009 0.006 0.000 0.027 0.005 0.022   
## 12 0.001 0.003 0.002 0.072 0.034 0.000 0.000   
## 13 0.000 0.001 0.000 0.007 0.016 0.000 0.000   
## 14 0.001 0.002 0.000 0.089 0.001 0.000 0.002   
## 15 0.000 0.000 0.004 0.045 0.000 0.000 0.001   
## 16 0.000 0.000 0.000 0.062 0.031 0.001 0.002   
## 17 0.000 0.007 0.008 0.289 0.077 0.001 0.000   
## 18 0.000 0.001 0.008 0.007 0.023 0.002 0.006   
## 19 0.000 0.002 0.037 0.020 0.000 0.003 0.010   
## 20 0.000 0.000 0.053 0.020 0.007 0.004 0.013   
## 21 0.002 0.004 0.020 0.000 0.019 0.003 0.003   
## 22 0.000 0.000 0.000 0.000 0.004 0.000 0.004   
## 23 0.001 0.003 0.027 0.005 0.001 0.000 0.011   
## 24 0.002 0.003 0.028 0.039 0.000 0.005 0.000   
## 25 0.001 0.001 0.001 0.009 0.008 0.000 0.003   
## 26 0.002 0.001 0.000 0.002 0.000 0.001 0.000   
## 27 0.001 0.000 0.002 0.000 0.000 0.001 0.000   
## 28 0.014 0.585 0.004 0.003 0.024 0.000 0.015   
## 29 0.001 0.031 0.000 0.000 0.000 0.000 0.001   
## 30 0.000 0.004 0.000 0.000 0.000 0.000 0.000   
## 31 0.057 0.037 0.000 0.002 0.012 0.011 0.015   
## 32 0.816 0.172 0.017 0.006 0.009 0.044 0.074   
## 33 0.003 0.000 0.000 0.000 0.000 0.000 0.000   
## 34 0.002 0.001 0.001 0.000 0.001 0.001 0.001   
## 35 0.023 0.005 0.022 0.006 0.025 0.047 0.036   
## 36 0.011 0.000 0.340 0.130 0.352 0.474 0.419   
## 37 0.002 0.001 0.289 0.107 0.300 0.383 0.322   
## 38 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## monthjul monthaug monthsep monthoct monthnov monthdec campaign previous  
## 1 0.000 0.000 0.000 0.000 0.000 0.000 0.003 0.001   
## 2 0.000 0.005 0.004 0.004 0.003 0.002 0.000 0.011   
## 3 0.001 0.005 0.004 0.002 0.000 0.004 0.005 0.088   
## 4 0.016 0.002 0.000 0.001 0.000 0.001 0.003 0.007   
## 5 0.007 0.001 0.019 0.021 0.002 0.012 0.000 0.003   
## 6 0.002 0.007 0.000 0.000 0.006 0.000 0.005 0.020   
## 7 0.022 0.034 0.000 0.003 0.005 0.002 0.000 0.007   
## 8 0.005 0.005 0.000 0.001 0.012 0.002 0.003 0.000   
## 9 0.004 0.003 0.014 0.029 0.015 0.033 0.000 0.005   
## 10 0.002 0.000 0.005 0.014 0.066 0.003 0.003 0.002   
## 11 0.000 0.000 0.007 0.058 0.001 0.010 0.000 0.024   
## 12 0.000 0.000 0.063 0.000 0.002 0.057 0.000 0.019   
## 13 0.000 0.000 0.022 0.012 0.001 0.008 0.000 0.006   
## 14 0.001 0.002 0.001 0.271 0.003 0.225 0.000 0.012   
## 15 0.000 0.000 0.292 0.035 0.004 0.316 0.000 0.005   
## 16 0.000 0.000 0.106 0.011 0.000 0.022 0.001 0.001   
## 17 0.000 0.000 0.034 0.000 0.000 0.022 0.000 0.045   
## 18 0.001 0.000 0.000 0.005 0.038 0.006 0.001 0.001   
## 19 0.008 0.012 0.020 0.005 0.003 0.033 0.001 0.014   
## 20 0.000 0.002 0.008 0.033 0.020 0.018 0.000 0.005   
## 21 0.001 0.022 0.007 0.013 0.002 0.004 0.008 0.063   
## 22 0.001 0.006 0.000 0.000 0.002 0.000 0.012 0.007   
## 23 0.003 0.000 0.012 0.023 0.004 0.015 0.011 0.066   
## 24 0.000 0.000 0.063 0.067 0.004 0.043 0.001 0.029   
## 25 0.032 0.000 0.010 0.018 0.010 0.011 0.000 0.023   
## 26 0.003 0.007 0.002 0.002 0.010 0.002 0.005 0.003   
## 27 0.021 0.011 0.000 0.002 0.016 0.000 0.007 0.001   
## 28 0.001 0.001 0.008 0.003 0.002 0.003 0.028 0.502   
## 29 0.008 0.013 0.001 0.002 0.002 0.000 0.877 0.025   
## 30 0.000 0.000 0.000 0.000 0.000 0.000 0.012 0.002   
## 31 0.002 0.014 0.004 0.003 0.005 0.002 0.003 0.002   
## 32 0.033 0.052 0.002 0.004 0.022 0.002 0.001 0.001   
## 33 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 34 0.000 0.000 0.001 0.001 0.000 0.000 0.000 0.000   
## 35 0.027 0.026 0.008 0.009 0.023 0.005 0.004 0.000   
## 36 0.422 0.427 0.157 0.195 0.399 0.079 0.003 0.000   
## 37 0.377 0.340 0.125 0.149 0.317 0.058 0.000 0.000   
## 38 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000

Nessun indice, eccetto intercetta, supera 30. Non ci sono problemi di multicollinearità.

## Bontà adattamento modello

hl\_glm2 <- hoslem.test(training\_set\_vsa$y =="yes", fitted(glm2\_vsa), g=100)  
hl\_glm2

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: training\_set\_vsa$y == "yes", fitted(glm2\_vsa)  
## X-squared = 291.9953, df = 98, p-value < 2.2e-16

Anche qua test rigettato, l'ipotesi nulla di buon adattamento va rigettata. Anche con 100 gruppi il p.value è bassissimo.

# Stima performance predittive

Bene, ora vediamo l'AUC sul validation set

glm2\_vsa\_predictions <- predict(glm2\_vsa, validation\_set\_vsa, type="response")  
  
AUC\_glm2\_vsa <- roc(validation\_set\_vsa$y, glm2\_vsa\_predictions, levels=c("no", "yes"))  
AUC\_glm2\_vsa$auc

## Area under the curve: 0.7463

Però. glm2 ha AUC di 0.7463 contro 0.746 di glm1. Per ora vince la step contro di me.

Vediamo la AUCPR

AUCPR\_glm2\_vsa <- pr.curve(glm2\_vsa\_predictions, weights.class0 = validation\_set\_vsa$y == "yes", curve=T)  
AUCPR\_glm2\_vsa

##   
## Precision-recall curve  
##   
## Area under curve (Integral):  
## 0.3228931   
##   
## Area under curve (Davis & Goadrich):  
## 0.3228801   
##   
## Curve for scores from 6.33801e-05 to 0.7474767   
## ( can be plotted with plot(x) )

plot(AUCPR\_glm2\_vsa)

Modello simile a glm1, ma usando i fattori invece delle variabili quantitative quando possibile

formula\_glm3 <- y ~ age\_class + job + marital + balance\_class + housing + loan + contact + day + month + campaign\_class + pdays\_class + age\_class\*marital + age\_class\*job + loan\*housing

glm3\_vsa <- glm(formula\_glm3, family = "binomial", data = training\_set\_vsa)  
  
summary(glm3\_vsa) #AIC molto basso, è normale la sovrastima delle performance, che non vuol dire overfitting.

##   
## Call:  
## glm(formula = formula\_glm3, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1687 -0.4810 -0.3656 -0.2391 3.0072   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.500080 0.451897 -5.532 3.16e-08  
## age\_class(30,59] 0.092068 0.428430 0.215 0.829848  
## age\_class(59,95] 0.526675 0.590778 0.891 0.372664  
## jobblue-collar 0.006063 0.178650 0.034 0.972926  
## jobentrepreneur 0.062842 0.387935 0.162 0.871312  
## jobhousemaid 0.106918 0.481816 0.222 0.824387  
## jobmanagement 0.122353 0.164247 0.745 0.456314  
## jobretired -10.231499 93.844429 -0.109 0.913182  
## jobself-employed 0.129211 0.268424 0.481 0.630253  
## jobservices 0.070667 0.198524 0.356 0.721871  
## jobstudent 0.164979 0.172464 0.957 0.338769  
## jobtechnician 0.057886 0.173687 0.333 0.738924  
## jobunemployed 0.056318 0.273420 0.206 0.836809  
## jobunknown 0.268962 0.886603 0.303 0.761614  
## maritalmarried -0.029380 0.414953 -0.071 0.943555  
## maritalsingle 0.369989 0.406510 0.910 0.362739  
## balance\_class0 0.009133 0.127519 0.072 0.942905  
## balance\_class(0, 22] -0.090903 0.157046 -0.579 0.562703  
## balance\_class(22,131] 0.075669 0.120813 0.626 0.531095  
## balance\_class(131,272] 0.274397 0.117516 2.335 0.019544  
## balance\_class(272,448] 0.354354 0.116856 3.032 0.002426  
## balance\_class(448,701] 0.280988 0.117285 2.396 0.016585  
## balance\_class(701,1126] 0.383203 0.116197 3.298 0.000974  
## balance\_class(1126,1859] 0.377881 0.115773 3.264 0.001099  
## balance\_class(1859,3574] 0.593122 0.113791 5.212 1.86e-07  
## balance\_class(3574,102127] 0.564231 0.114875 4.912 9.03e-07  
## housingyes -0.586780 0.052034 -11.277 < 2e-16  
## loanyes -0.550209 0.101243 -5.435 5.49e-08  
## contacttelephone -0.308210 0.083170 -3.706 0.000211  
## contactunknown -1.283894 0.083019 -15.465 < 2e-16  
## day 0.002347 0.002804 0.837 0.402590  
## monthfeb 0.718895 0.147235 4.883 1.05e-06  
## monthmar 2.143796 0.179165 11.965 < 2e-16  
## monthapr 1.150920 0.137452 8.373 < 2e-16  
## monthmay 0.584910 0.136655 4.280 1.87e-05  
## monthjun 1.231421 0.151019 8.154 3.52e-16  
## monthjul 0.514489 0.134080 3.837 0.000124  
## monthaug 0.348577 0.136463 2.554 0.010638  
## monthsep 1.746246 0.171984 10.154 < 2e-16  
## monthoct 1.622463 0.158571 10.232 < 2e-16  
## monthnov 0.208321 0.140448 1.483 0.138007  
## monthdec 1.939394 0.218544 8.874 < 2e-16  
## campaign\_class2 -0.218022 0.050740 -4.297 1.73e-05  
## campaign\_class3 -0.041808 0.066185 -0.632 0.527590  
## campaign\_class4 -0.355367 0.088196 -4.029 5.59e-05  
## campaign\_class5 -0.516713 0.123476 -4.185 2.85e-05  
## campaign\_classup 5 -0.729599 0.095520 -7.638 2.20e-14  
## pdays\_class(0,91] 1.045691 0.101408 10.312 < 2e-16  
## pdays\_class(91,108] 1.328945 0.105291 12.622 < 2e-16  
## pdays\_class(108,159] -0.003548 0.137247 -0.026 0.979379  
## pdays\_class(159,181] 0.803065 0.121545 6.607 3.92e-11  
## pdays\_class (181,194] 1.070988 0.111692 9.589 < 2e-16  
## pdays\_class(194,258] -0.084672 0.150489 -0.563 0.573674  
## pdays\_class(258,300] 0.219745 0.137347 1.600 0.109615  
## pdays\_class(300,343] -0.344559 0.158714 -2.171 0.029936  
## pdays\_class(343,362] -0.121708 0.174408 -0.698 0.485281  
## pdays\_class(362,871] 0.859966 0.117542 7.316 2.55e-13  
## age\_class(30,59]:maritalmarried -0.187301 0.421075 -0.445 0.656453  
## age\_class(59,95]:maritalmarried -0.423333 0.457321 -0.926 0.354611  
## age\_class(30,59]:maritalsingle -0.435756 0.414457 -1.051 0.293079  
## age\_class(59,95]:maritalsingle -1.008681 0.647257 -1.558 0.119140  
## age\_class(30,59]:jobblue-collar -0.193497 0.200684 -0.964 0.334952  
## age\_class(59,95]:jobblue-collar 0.227028 0.580469 0.391 0.695715  
## age\_class(30,59]:jobentrepreneur -0.229958 0.417375 -0.551 0.581659  
## age\_class(59,95]:jobentrepreneur 0.800050 0.813006 0.984 0.325084  
## age\_class(30,59]:jobhousemaid -0.491352 0.509042 -0.965 0.334421  
## age\_class(59,95]:jobhousemaid -0.055588 0.702190 -0.079 0.936902  
## age\_class(30,59]:jobmanagement -0.074559 0.184360 -0.404 0.685905  
## age\_class(59,95]:jobmanagement 0.869773 0.473654 1.836 0.066312  
## age\_class(30,59]:jobretired 9.874420 93.844571 0.105 0.916200  
## age\_class(59,95]:jobretired 10.767000 93.845290 0.115 0.908658  
## age\_class(30,59]:jobself-employed -0.397719 0.308222 -1.290 0.196924  
## age\_class(59,95]:jobself-employed 1.199187 0.654842 1.831 0.067061  
## age\_class(30,59]:jobservices -0.066928 0.225592 -0.297 0.766711  
## age\_class(59,95]:jobservices -0.022954 0.825470 -0.028 0.977816  
## age\_class(30,59]:jobstudent -0.057097 0.338063 -0.169 0.865879  
## age\_class(59,95]:jobstudent NA NA NA NA  
## age\_class(30,59]:jobtechnician -0.037721 0.195406 -0.193 0.846929  
## age\_class(59,95]:jobtechnician 0.439834 0.555580 0.792 0.428555  
## age\_class(30,59]:jobunemployed 0.165930 0.304908 0.544 0.586305  
## age\_class(59,95]:jobunemployed 0.292111 0.868208 0.336 0.736530  
## age\_class(30,59]:jobunknown -0.459836 0.934210 -0.492 0.622565  
## age\_class(59,95]:jobunknown -1.052628 1.249830 -0.842 0.399667  
## housingyes:loanyes 0.304603 0.136331 2.234 0.025464  
##   
## (Intercept) \*\*\*  
## age\_class(30,59]   
## age\_class(59,95]   
## jobblue-collar   
## jobentrepreneur   
## jobhousemaid   
## jobmanagement   
## jobretired   
## jobself-employed   
## jobservices   
## jobstudent   
## jobtechnician   
## jobunemployed   
## jobunknown   
## maritalmarried   
## maritalsingle   
## balance\_class0   
## balance\_class(0, 22]   
## balance\_class(22,131]   
## balance\_class(131,272] \*   
## balance\_class(272,448] \*\*   
## balance\_class(448,701] \*   
## balance\_class(701,1126] \*\*\*  
## balance\_class(1126,1859] \*\*   
## balance\_class(1859,3574] \*\*\*  
## balance\_class(3574,102127] \*\*\*  
## housingyes \*\*\*  
## loanyes \*\*\*  
## contacttelephone \*\*\*  
## contactunknown \*\*\*  
## day   
## monthfeb \*\*\*  
## monthmar \*\*\*  
## monthapr \*\*\*  
## monthmay \*\*\*  
## monthjun \*\*\*  
## monthjul \*\*\*  
## monthaug \*   
## monthsep \*\*\*  
## monthoct \*\*\*  
## monthnov   
## monthdec \*\*\*  
## campaign\_class2 \*\*\*  
## campaign\_class3   
## campaign\_class4 \*\*\*  
## campaign\_class5 \*\*\*  
## campaign\_classup 5 \*\*\*  
## pdays\_class(0,91] \*\*\*  
## pdays\_class(91,108] \*\*\*  
## pdays\_class(108,159]   
## pdays\_class(159,181] \*\*\*  
## pdays\_class (181,194] \*\*\*  
## pdays\_class(194,258]   
## pdays\_class(258,300]   
## pdays\_class(300,343] \*   
## pdays\_class(343,362]   
## pdays\_class(362,871] \*\*\*  
## age\_class(30,59]:maritalmarried   
## age\_class(59,95]:maritalmarried   
## age\_class(30,59]:maritalsingle   
## age\_class(59,95]:maritalsingle   
## age\_class(30,59]:jobblue-collar   
## age\_class(59,95]:jobblue-collar   
## age\_class(30,59]:jobentrepreneur   
## age\_class(59,95]:jobentrepreneur   
## age\_class(30,59]:jobhousemaid   
## age\_class(59,95]:jobhousemaid   
## age\_class(30,59]:jobmanagement   
## age\_class(59,95]:jobmanagement .   
## age\_class(30,59]:jobretired   
## age\_class(59,95]:jobretired   
## age\_class(30,59]:jobself-employed   
## age\_class(59,95]:jobself-employed .   
## age\_class(30,59]:jobservices   
## age\_class(59,95]:jobservices   
## age\_class(30,59]:jobstudent   
## age\_class(59,95]:jobstudent   
## age\_class(30,59]:jobtechnician   
## age\_class(59,95]:jobtechnician   
## age\_class(30,59]:jobunemployed   
## age\_class(59,95]:jobunemployed   
## age\_class(30,59]:jobunknown   
## age\_class(59,95]:jobunknown   
## housingyes:loanyes \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 16431 on 27044 degrees of freedom  
## AIC: 16597  
##   
## Number of Fisher Scoring iterations: 10

AIC più bassa sinora.

# Diagnostica

## Multicollinearità

matrix\_glm3 <- model.matrix(glm3\_vsa)  
matrix\_glm3 <- matrix\_glm3[,colSums(matrix\_glm3 != 0) != 0]  
  
cd\_glm3 <- colldiag(matrix\_glm3)  
print(cd\_glm3)

## Condition  
## Index Variance Decomposition Proportions  
## intercept (Intercept) age\_class(30,59]  
## 1 1.000 0.000 0.000 0.000   
## 2 1.790 0.000 0.000 0.000   
## 3 2.182 0.000 0.000 0.000   
## 4 2.296 0.000 0.000 0.000   
## 5 2.355 0.000 0.000 0.000   
## 6 2.364 0.000 0.000 0.000   
## 7 2.377 0.000 0.000 0.000   
## 8 2.382 0.000 0.000 0.000   
## 9 2.392 0.000 0.000 0.000   
## 10 2.400 0.000 0.000 0.000   
## 11 2.469 0.000 0.000 0.000   
## 12 2.618 0.000 0.000 0.000   
## 13 2.683 0.000 0.000 0.000   
## 14 2.732 0.000 0.000 0.000   
## 15 2.871 0.000 0.000 0.000   
## 16 2.890 0.000 0.000 0.000   
## 17 2.957 0.000 0.000 0.000   
## 18 3.002 0.000 0.000 0.000   
## 19 3.057 0.000 0.000 0.000   
## 20 3.107 0.000 0.000 0.000   
## 21 3.135 0.000 0.000 0.000   
## 22 3.171 0.000 0.000 0.000   
## 23 3.204 0.000 0.000 0.000   
## 24 3.222 0.000 0.000 0.000   
## 25 3.246 0.000 0.000 0.000   
## 26 3.257 0.000 0.000 0.000   
## 27 3.268 0.000 0.000 0.000   
## 28 3.277 0.000 0.000 0.000   
## 29 3.282 0.000 0.000 0.000   
## 30 3.295 0.000 0.000 0.000   
## 31 3.300 0.000 0.000 0.000   
## 32 3.304 0.000 0.000 0.000   
## 33 3.306 0.000 0.000 0.000   
## 34 3.323 0.000 0.000 0.000   
## 35 3.329 0.000 0.000 0.000   
## 36 3.334 0.000 0.000 0.000   
## 37 3.342 0.000 0.000 0.000   
## 38 3.350 0.000 0.000 0.000   
## 39 3.352 0.000 0.000 0.000   
## 40 3.357 0.000 0.000 0.000   
## 41 3.370 0.000 0.000 0.000   
## 42 3.372 0.000 0.000 0.000   
## 43 3.378 0.000 0.000 0.000   
## 44 3.387 0.000 0.000 0.000   
## 45 3.391 0.000 0.000 0.000   
## 46 3.417 0.000 0.000 0.000   
## 47 3.418 0.000 0.000 0.000   
## 48 3.462 0.000 0.000 0.000   
## 49 3.493 0.000 0.000 0.000   
## 50 3.528 0.000 0.000 0.000   
## 51 3.561 0.000 0.000 0.000   
## 52 3.606 0.000 0.000 0.000   
## 53 3.630 0.000 0.000 0.000   
## 54 3.682 0.000 0.000 0.000   
## 55 3.774 0.000 0.000 0.000   
## 56 3.845 0.000 0.000 0.000   
## 57 3.936 0.000 0.000 0.000   
## 58 4.155 0.000 0.000 0.000   
## 59 4.422 0.000 0.000 0.000   
## 60 5.383 0.000 0.000 0.001   
## 61 5.940 0.000 0.000 0.000   
## 62 6.141 0.000 0.000 0.000   
## 63 7.088 0.000 0.000 0.000   
## 64 7.647 0.000 0.000 0.000   
## 65 8.061 0.000 0.000 0.000   
## 66 8.775 0.000 0.000 0.001   
## 67 9.307 0.000 0.000 0.000   
## 68 9.860 0.000 0.000 0.002   
## 69 10.648 0.000 0.000 0.000   
## 70 11.517 0.000 0.000 0.003   
## 71 11.559 0.000 0.000 0.000   
## 72 12.167 0.000 0.000 0.000   
## 73 12.516 0.000 0.000 0.000   
## 74 13.358 0.000 0.000 0.006   
## 75 14.087 0.000 0.000 0.000   
## 76 14.756 0.000 0.000 0.000   
## 77 19.623 0.000 0.000 0.000   
## 78 20.008 0.000 0.000 0.000   
## 79 21.561 0.000 0.000 0.003   
## 80 25.780 0.000 0.000 0.014   
## 81 37.010 0.000 0.000 0.033   
## 82 87.115 0.000 0.000 0.001   
## 83 98.820 0.000 0.000 0.934   
## 84 49006905062698.305 1.000 1.000 0.000   
## age\_class(59,95] jobblue-collar jobentrepreneur jobhousemaid  
## 1 0.000 0.000 0.000 0.000   
## 2 0.002 0.000 0.000 0.000   
## 3 0.000 0.003 0.000 0.000   
## 4 0.000 0.001 0.001 0.005   
## 5 0.000 0.000 0.004 0.000   
## 6 0.000 0.000 0.005 0.004   
## 7 0.000 0.001 0.007 0.000   
## 8 0.000 0.000 0.000 0.001   
## 9 0.000 0.000 0.001 0.000   
## 10 0.000 0.001 0.000 0.000   
## 11 0.000 0.001 0.003 0.002   
## 12 0.000 0.000 0.000 0.000   
## 13 0.000 0.000 0.000 0.000   
## 14 0.000 0.001 0.000 0.000   
## 15 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000   
## 18 0.000 0.001 0.000 0.000   
## 19 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000   
## 30 0.000 0.000 0.000 0.000   
## 31 0.000 0.000 0.000 0.000   
## 32 0.000 0.000 0.000 0.000   
## 33 0.000 0.000 0.000 0.000   
## 34 0.000 0.000 0.000 0.000   
## 35 0.000 0.000 0.000 0.000   
## 36 0.000 0.000 0.000 0.000   
## 37 0.000 0.000 0.000 0.000   
## 38 0.000 0.000 0.000 0.000   
## 39 0.000 0.000 0.000 0.000   
## 40 0.000 0.000 0.000 0.000   
## 41 0.000 0.000 0.000 0.000   
## 42 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000   
## 44 0.000 0.000 0.000 0.000   
## 45 0.000 0.000 0.000 0.000   
## 46 0.000 0.000 0.000 0.000   
## 47 0.000 0.000 0.000 0.000   
## 48 0.000 0.000 0.000 0.000   
## 49 0.000 0.000 0.000 0.000   
## 50 0.000 0.000 0.000 0.000   
## 51 0.000 0.000 0.000 0.000   
## 52 0.000 0.000 0.000 0.000   
## 53 0.000 0.000 0.000 0.000   
## 54 0.000 0.000 0.000 0.000   
## 55 0.000 0.000 0.000 0.000   
## 56 0.000 0.000 0.000 0.000   
## 57 0.000 0.000 0.000 0.000   
## 58 0.000 0.000 0.000 0.000   
## 59 0.000 0.000 0.000 0.000   
## 60 0.000 0.000 0.000 0.000   
## 61 0.000 0.000 0.000 0.000   
## 62 0.000 0.001 0.000 0.000   
## 63 0.000 0.014 0.003 0.002   
## 64 0.000 0.003 0.000 0.000   
## 65 0.000 0.000 0.000 0.000   
## 66 0.001 0.006 0.001 0.000   
## 67 0.012 0.005 0.001 0.000   
## 68 0.003 0.003 0.001 0.001   
## 69 0.000 0.008 0.000 0.000   
## 70 0.000 0.188 0.004 0.001   
## 71 0.000 0.040 0.001 0.000   
## 72 0.000 0.006 0.000 0.000   
## 73 0.000 0.001 0.000 0.000   
## 74 0.000 0.043 0.037 0.002   
## 75 0.000 0.027 0.091 0.000   
## 76 0.000 0.000 0.671 0.000   
## 77 0.012 0.010 0.004 0.668   
## 78 0.001 0.000 0.000 0.178   
## 79 0.609 0.003 0.001 0.004   
## 80 0.014 0.017 0.004 0.010   
## 81 0.014 0.593 0.154 0.118   
## 82 0.001 0.004 0.001 0.001   
## 83 0.329 0.016 0.004 0.002   
## 84 0.000 0.000 0.000 0.000   
## jobmanagement jobretired jobself-employed jobservices jobstudent  
## 1 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000 0.000   
## 3 0.002 0.000 0.000 0.000 0.003   
## 4 0.000 0.000 0.001 0.001 0.002   
## 5 0.000 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.001 0.000 0.000   
## 7 0.001 0.000 0.001 0.000 0.000   
## 8 0.004 0.000 0.000 0.002 0.000   
## 9 0.001 0.000 0.025 0.000 0.000   
## 10 0.000 0.000 0.000 0.018 0.000   
## 11 0.000 0.000 0.000 0.001 0.018   
## 12 0.000 0.000 0.000 0.000 0.006   
## 13 0.000 0.000 0.000 0.001 0.009   
## 14 0.001 0.000 0.000 0.001 0.012   
## 15 0.000 0.000 0.000 0.000 0.047   
## 16 0.000 0.000 0.000 0.000 0.022   
## 17 0.000 0.000 0.000 0.000 0.001   
## 18 0.000 0.000 0.000 0.000 0.020   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.002   
## 21 0.000 0.000 0.000 0.000 0.002   
## 22 0.000 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000 0.000   
## 30 0.000 0.000 0.000 0.000 0.000   
## 31 0.000 0.000 0.000 0.000 0.001   
## 32 0.000 0.000 0.000 0.000 0.000   
## 33 0.000 0.000 0.000 0.000 0.000   
## 34 0.000 0.000 0.000 0.000 0.000   
## 35 0.000 0.000 0.000 0.000 0.000   
## 36 0.000 0.000 0.000 0.000 0.000   
## 37 0.000 0.000 0.000 0.000 0.000   
## 38 0.000 0.000 0.000 0.000 0.000   
## 39 0.000 0.000 0.000 0.000 0.000   
## 40 0.000 0.000 0.000 0.000 0.000   
## 41 0.000 0.000 0.000 0.000 0.000   
## 42 0.000 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000 0.001   
## 44 0.000 0.000 0.000 0.000 0.000   
## 45 0.000 0.000 0.000 0.000 0.000   
## 46 0.000 0.000 0.000 0.000 0.000   
## 47 0.000 0.000 0.000 0.000 0.000   
## 48 0.000 0.000 0.000 0.000 0.000   
## 49 0.000 0.000 0.000 0.000 0.000   
## 50 0.000 0.000 0.000 0.000 0.000   
## 51 0.000 0.000 0.000 0.000 0.000   
## 52 0.000 0.000 0.000 0.000 0.000   
## 53 0.000 0.000 0.000 0.000 0.001   
## 54 0.000 0.000 0.000 0.000 0.000   
## 55 0.000 0.000 0.000 0.000 0.000   
## 56 0.000 0.000 0.000 0.000 0.000   
## 57 0.000 0.000 0.000 0.000 0.012   
## 58 0.000 0.000 0.000 0.000 0.205   
## 59 0.000 0.000 0.000 0.000 0.004   
## 60 0.000 0.000 0.000 0.000 0.122   
## 61 0.000 0.000 0.000 0.000 0.001   
## 62 0.001 0.000 0.000 0.003 0.073   
## 63 0.012 0.000 0.004 0.007 0.020   
## 64 0.004 0.000 0.001 0.004 0.003   
## 65 0.000 0.000 0.000 0.001 0.000   
## 66 0.009 0.000 0.003 0.015 0.003   
## 67 0.002 0.000 0.001 0.007 0.005   
## 68 0.009 0.000 0.003 0.020 0.003   
## 69 0.006 0.000 0.004 0.449 0.002   
## 70 0.000 0.000 0.000 0.000 0.009   
## 71 0.001 0.000 0.004 0.001 0.000   
## 72 0.003 0.000 0.630 0.001 0.000   
## 73 0.169 0.000 0.108 0.006 0.005   
## 74 0.085 0.000 0.003 0.002 0.018   
## 75 0.033 0.000 0.001 0.000 0.003   
## 76 0.044 0.000 0.006 0.003 0.008   
## 77 0.014 0.000 0.004 0.006 0.004   
## 78 0.001 0.000 0.000 0.000 0.001   
## 79 0.002 0.005 0.001 0.002 0.001   
## 80 0.018 0.000 0.005 0.010 0.010   
## 81 0.553 0.000 0.187 0.405 0.323   
## 82 0.004 0.990 0.002 0.003 0.004   
## 83 0.017 0.003 0.005 0.030 0.010   
## 84 0.000 0.000 0.000 0.000 0.000   
## jobtechnician jobunemployed jobunknown maritalmarried maritalsingle  
## 1 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000 0.000   
## 3 0.002 0.000 0.000 0.000 0.001   
## 4 0.000 0.000 0.001 0.000 0.000   
## 5 0.000 0.000 0.009 0.000 0.000   
## 6 0.003 0.003 0.002 0.000 0.000   
## 7 0.004 0.007 0.001 0.000 0.000   
## 8 0.000 0.017 0.000 0.000 0.000   
## 9 0.000 0.001 0.000 0.000 0.000   
## 10 0.002 0.002 0.000 0.000 0.000   
## 11 0.001 0.001 0.000 0.000 0.001   
## 12 0.000 0.000 0.000 0.000 0.000   
## 13 0.000 0.000 0.000 0.000 0.000   
## 14 0.000 0.000 0.000 0.000 0.000   
## 15 0.000 0.000 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.000 0.000   
## 18 0.000 0.001 0.000 0.000 0.000   
## 19 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000 0.000   
## 26 0.000 0.000 0.000 0.000 0.000   
## 27 0.000 0.000 0.000 0.000 0.000   
## 28 0.000 0.000 0.000 0.000 0.000   
## 29 0.000 0.000 0.000 0.000 0.000   
## 30 0.000 0.000 0.000 0.000 0.000   
## 31 0.000 0.000 0.000 0.000 0.000   
## 32 0.000 0.000 0.000 0.000 0.000   
## 33 0.000 0.000 0.000 0.000 0.000   
## 34 0.000 0.000 0.000 0.000 0.000   
## 35 0.000 0.000 0.000 0.000 0.000   
## 36 0.000 0.000 0.000 0.000 0.000   
## 37 0.000 0.000 0.000 0.000 0.000   
## 38 0.000 0.000 0.000 0.000 0.000   
## 39 0.000 0.000 0.000 0.000 0.000   
## 40 0.000 0.000 0.000 0.000 0.000   
## 41 0.000 0.000 0.000 0.000 0.000   
## 42 0.000 0.000 0.000 0.000 0.000   
## 43 0.000 0.000 0.000 0.000 0.000   
## 44 0.000 0.000 0.000 0.000 0.000   
## 45 0.000 0.000 0.000 0.000 0.000   
## 46 0.000 0.000 0.000 0.000 0.000   
## 47 0.000 0.000 0.000 0.000 0.000   
## 48 0.000 0.000 0.000 0.000 0.000   
## 49 0.000 0.000 0.000 0.000 0.000   
## 50 0.000 0.000 0.000 0.000 0.000   
## 51 0.000 0.000 0.000 0.000 0.000   
## 52 0.000 0.000 0.000 0.000 0.000   
## 53 0.000 0.000 0.000 0.000 0.000   
## 54 0.000 0.000 0.000 0.000 0.000   
## 55 0.000 0.000 0.000 0.000 0.000   
## 56 0.000 0.000 0.000 0.000 0.000   
## 57 0.000 0.000 0.000 0.000 0.000   
## 58 0.000 0.000 0.000 0.000 0.000   
## 59 0.000 0.000 0.000 0.000 0.000   
## 60 0.000 0.000 0.000 0.000 0.000   
## 61 0.000 0.000 0.000 0.000 0.000   
## 62 0.002 0.001 0.000 0.000 0.002   
## 63 0.011 0.003 0.000 0.005 0.004   
## 64 0.006 0.002 0.000 0.000 0.003   
## 65 0.000 0.000 0.000 0.000 0.000   
## 66 0.012 0.006 0.000 0.000 0.006   
## 67 0.003 0.001 0.000 0.001 0.000   
## 68 0.013 0.001 0.000 0.000 0.009   
## 69 0.024 0.013 0.000 0.000 0.001   
## 70 0.046 0.072 0.000 0.006 0.004   
## 71 0.006 0.664 0.000 0.001 0.000   
## 72 0.073 0.001 0.000 0.000 0.000   
## 73 0.173 0.010 0.000 0.001 0.003   
## 74 0.028 0.002 0.000 0.006 0.008   
## 75 0.006 0.000 0.000 0.004 0.005   
## 76 0.020 0.003 0.000 0.002 0.005   
## 77 0.010 0.003 0.173 0.000 0.000   
## 78 0.001 0.000 0.781 0.000 0.000   
## 79 0.003 0.001 0.005 0.000 0.000   
## 80 0.015 0.002 0.001 0.001 0.000   
## 81 0.512 0.174 0.023 0.052 0.034   
## 82 0.004 0.001 0.000 0.004 0.005   
## 83 0.018 0.007 0.000 0.914 0.905   
## 84 0.000 0.000 0.000 0.000 0.000   
## balance\_class0 balance\_class(0, 22] balance\_class(22,131]  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.000   
## 3 0.000 0.000 0.000   
## 4 0.001 0.000 0.000   
## 5 0.000 0.000 0.000   
## 6 0.000 0.000 0.000   
## 7 0.000 0.000 0.000   
## 8 0.000 0.000 0.000   
## 9 0.000 0.000 0.000   
## 10 0.000 0.000 0.000   
## 11 0.001 0.000 0.000   
## 12 0.000 0.000 0.000   
## 13 0.000 0.000 0.000   
## 14 0.000 0.003 0.001   
## 15 0.006 0.003 0.004   
## 16 0.004 0.001 0.004   
## 17 0.002 0.001 0.000   
## 18 0.000 0.000 0.000   
## 19 0.003 0.001 0.004   
## 20 0.000 0.004 0.000   
## 21 0.001 0.000 0.000   
## 22 0.000 0.000 0.007   
## 23 0.004 0.002 0.008   
## 24 0.001 0.007 0.005   
## 25 0.007 0.000 0.017   
## 26 0.001 0.026 0.020   
## 27 0.000 0.041 0.011   
## 28 0.033 0.050 0.003   
## 29 0.005 0.001 0.017   
## 30 0.013 0.024 0.000   
## 31 0.008 0.024 0.000   
## 32 0.001 0.061 0.030   
## 33 0.025 0.009 0.000   
## 34 0.033 0.055 0.006   
## 35 0.004 0.002 0.036   
## 36 0.000 0.023 0.019   
## 37 0.002 0.009 0.000   
## 38 0.042 0.001 0.001   
## 39 0.004 0.026 0.020   
## 40 0.008 0.051 0.052   
## 41 0.007 0.055 0.013   
## 42 0.036 0.008 0.003   
## 43 0.003 0.056 0.000   
## 44 0.052 0.006 0.002   
## 45 0.068 0.003 0.040   
## 46 0.000 0.000 0.010   
## 47 0.011 0.058 0.000   
## 48 0.022 0.003 0.001   
## 49 0.021 0.011 0.000   
## 50 0.000 0.000 0.005   
## 51 0.009 0.002 0.025   
## 52 0.005 0.000 0.003   
## 53 0.000 0.000 0.004   
## 54 0.007 0.015 0.015   
## 55 0.000 0.002 0.001   
## 56 0.001 0.002 0.009   
## 57 0.004 0.000 0.000   
## 58 0.001 0.001 0.000   
## 59 0.000 0.000 0.001   
## 60 0.001 0.000 0.000   
## 61 0.000 0.000 0.001   
## 62 0.007 0.001 0.003   
## 63 0.000 0.000 0.000   
## 64 0.005 0.003 0.005   
## 65 0.000 0.000 0.000   
## 66 0.000 0.000 0.000   
## 67 0.016 0.009 0.021   
## 68 0.084 0.048 0.095   
## 69 0.000 0.000 0.000   
## 70 0.000 0.000 0.000   
## 71 0.000 0.000 0.001   
## 72 0.000 0.000 0.000   
## 73 0.000 0.000 0.000   
## 74 0.156 0.095 0.161   
## 75 0.239 0.167 0.271   
## 76 0.008 0.005 0.006   
## 77 0.000 0.001 0.000   
## 78 0.000 0.000 0.000   
## 79 0.000 0.000 0.000   
## 80 0.024 0.020 0.030   
## 81 0.002 0.002 0.004   
## 82 0.000 0.000 0.000   
## 83 0.001 0.001 0.001   
## 84 0.000 0.000 0.000   
## balance\_class(131,272] balance\_class(272,448] balance\_class(448,701]  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.000   
## 3 0.000 0.000 0.000   
## 4 0.000 0.000 0.000   
## 5 0.000 0.000 0.000   
## 6 0.000 0.000 0.000   
## 7 0.000 0.000 0.000   
## 8 0.000 0.000 0.000   
## 9 0.000 0.000 0.000   
## 10 0.000 0.000 0.000   
## 11 0.000 0.000 0.000   
## 12 0.000 0.000 0.000   
## 13 0.000 0.000 0.000   
## 14 0.000 0.000 0.000   
## 15 0.001 0.000 0.001   
## 16 0.000 0.001 0.000   
## 17 0.001 0.005 0.008   
## 18 0.001 0.000 0.000   
## 19 0.004 0.001 0.001   
## 20 0.000 0.000 0.000   
## 21 0.001 0.005 0.001   
## 22 0.004 0.003 0.001   
## 23 0.004 0.000 0.006   
## 24 0.001 0.004 0.000   
## 25 0.014 0.015 0.000   
## 26 0.000 0.025 0.002   
## 27 0.023 0.019 0.010   
## 28 0.005 0.003 0.003   
## 29 0.019 0.002 0.005   
## 30 0.031 0.012 0.001   
## 31 0.001 0.022 0.021   
## 32 0.003 0.001 0.057   
## 33 0.001 0.000 0.006   
## 34 0.001 0.020 0.012   
## 35 0.026 0.018 0.013   
## 36 0.095 0.042 0.002   
## 37 0.000 0.020 0.075   
## 38 0.000 0.002 0.005   
## 39 0.014 0.048 0.000   
## 40 0.034 0.001 0.029   
## 41 0.002 0.047 0.027   
## 42 0.003 0.006 0.021   
## 43 0.043 0.009 0.026   
## 44 0.002 0.001 0.006   
## 45 0.005 0.003 0.000   
## 46 0.022 0.001 0.000   
## 47 0.005 0.007 0.035   
## 48 0.006 0.044 0.014   
## 49 0.001 0.004 0.000   
## 50 0.000 0.000 0.000   
## 51 0.001 0.005 0.000   
## 52 0.000 0.000 0.000   
## 53 0.001 0.002 0.000   
## 54 0.016 0.001 0.000   
## 55 0.000 0.001 0.000   
## 56 0.002 0.003 0.001   
## 57 0.001 0.000 0.001   
## 58 0.000 0.000 0.000   
## 59 0.001 0.000 0.001   
## 60 0.000 0.000 0.000   
## 61 0.000 0.000 0.000   
## 62 0.005 0.006 0.007   
## 63 0.000 0.000 0.000   
## 64 0.003 0.004 0.003   
## 65 0.000 0.000 0.000   
## 66 0.000 0.000 0.001   
## 67 0.022 0.021 0.022   
## 68 0.096 0.101 0.108   
## 69 0.000 0.001 0.001   
## 70 0.000 0.000 0.000   
## 71 0.000 0.001 0.001   
## 72 0.000 0.000 0.000   
## 73 0.000 0.000 0.000   
## 74 0.163 0.161 0.156   
## 75 0.263 0.257 0.259   
## 76 0.009 0.008 0.010   
## 77 0.000 0.000 0.000   
## 78 0.000 0.000 0.000   
## 79 0.000 0.000 0.000   
## 80 0.034 0.033 0.031   
## 81 0.004 0.004 0.003   
## 82 0.000 0.000 0.000   
## 83 0.001 0.001 0.001   
## 84 0.000 0.000 0.000   
## balance\_class(701,1126] balance\_class(1126,1859]  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.000 0.000   
## 13 0.000 0.000   
## 14 0.000 0.001   
## 15 0.000 0.001   
## 16 0.000 0.000   
## 17 0.001 0.000   
## 18 0.000 0.000   
## 19 0.000 0.001   
## 20 0.001 0.005   
## 21 0.000 0.006   
## 22 0.000 0.001   
## 23 0.013 0.001   
## 24 0.008 0.003   
## 25 0.000 0.000   
## 26 0.004 0.007   
## 27 0.007 0.021   
## 28 0.012 0.001   
## 29 0.036 0.000   
## 30 0.011 0.047   
## 31 0.009 0.000   
## 32 0.004 0.005   
## 33 0.003 0.043   
## 34 0.074 0.023   
## 35 0.034 0.053   
## 36 0.053 0.009   
## 37 0.006 0.003   
## 38 0.014 0.033   
## 39 0.000 0.008   
## 40 0.017 0.023   
## 41 0.013 0.001   
## 42 0.001 0.025   
## 43 0.001 0.000   
## 44 0.020 0.008   
## 45 0.025 0.017   
## 46 0.003 0.001   
## 47 0.003 0.005   
## 48 0.006 0.008   
## 49 0.003 0.001   
## 50 0.002 0.000   
## 51 0.001 0.012   
## 52 0.000 0.000   
## 53 0.003 0.000   
## 54 0.002 0.008   
## 55 0.000 0.000   
## 56 0.000 0.001   
## 57 0.000 0.000   
## 58 0.000 0.001   
## 59 0.000 0.000   
## 60 0.000 0.000   
## 61 0.000 0.000   
## 62 0.006 0.008   
## 63 0.000 0.000   
## 64 0.004 0.004   
## 65 0.000 0.001   
## 66 0.000 0.000   
## 67 0.022 0.021   
## 68 0.104 0.109   
## 69 0.001 0.001   
## 70 0.000 0.000   
## 71 0.000 0.001   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.160 0.161   
## 75 0.262 0.266   
## 76 0.010 0.010   
## 77 0.000 0.000   
## 78 0.000 0.000   
## 79 0.000 0.000   
## 80 0.031 0.027   
## 81 0.003 0.002   
## 82 0.000 0.000   
## 83 0.001 0.001   
## 84 0.000 0.000   
## balance\_class(1859,3574] balance\_class(3574,102127] housingyes loanyes  
## 1 0.000 0.000 0.001 0.000   
## 2 0.000 0.000 0.001 0.000   
## 3 0.000 0.001 0.001 0.001   
## 4 0.000 0.001 0.002 0.000   
## 5 0.000 0.000 0.000 0.001   
## 6 0.000 0.000 0.000 0.000   
## 7 0.000 0.000 0.000 0.002   
## 8 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.000 0.003   
## 12 0.000 0.000 0.000 0.005   
## 13 0.000 0.001 0.000 0.033   
## 14 0.003 0.002 0.001 0.029   
## 15 0.001 0.005 0.002 0.001   
## 16 0.006 0.005 0.000 0.005   
## 17 0.004 0.006 0.001 0.001   
## 18 0.000 0.005 0.000 0.000   
## 19 0.001 0.002 0.003 0.001   
## 20 0.001 0.006 0.000 0.000   
## 21 0.001 0.004 0.001 0.001   
## 22 0.005 0.000 0.000 0.001   
## 23 0.000 0.000 0.000 0.002   
## 24 0.012 0.007 0.000 0.000   
## 25 0.010 0.015 0.000 0.000   
## 26 0.013 0.006 0.000 0.001   
## 27 0.000 0.021 0.000 0.000   
## 28 0.001 0.003 0.000 0.000   
## 29 0.000 0.019 0.000 0.000   
## 30 0.032 0.010 0.000 0.000   
## 31 0.013 0.004 0.000 0.000   
## 32 0.003 0.000 0.000 0.000   
## 33 0.000 0.001 0.000 0.000   
## 34 0.012 0.014 0.000 0.000   
## 35 0.011 0.000 0.000 0.000   
## 36 0.030 0.012 0.000 0.000   
## 37 0.016 0.001 0.000 0.000   
## 38 0.033 0.018 0.000 0.000   
## 39 0.000 0.001 0.000 0.000   
## 40 0.029 0.000 0.000 0.000   
## 41 0.004 0.000 0.000 0.000   
## 42 0.000 0.003 0.000 0.000   
## 43 0.004 0.000 0.000 0.000   
## 44 0.042 0.018 0.000 0.000   
## 45 0.001 0.028 0.000 0.000   
## 46 0.009 0.003 0.000 0.001   
## 47 0.000 0.002 0.000 0.000   
## 48 0.058 0.043 0.000 0.000   
## 49 0.001 0.015 0.001 0.000   
## 50 0.000 0.017 0.001 0.002   
## 51 0.009 0.011 0.001 0.001   
## 52 0.001 0.007 0.000 0.000   
## 53 0.000 0.020 0.004 0.001   
## 54 0.008 0.019 0.003 0.001   
## 55 0.001 0.000 0.002 0.002   
## 56 0.007 0.023 0.000 0.000   
## 57 0.000 0.001 0.002 0.000   
## 58 0.004 0.001 0.000 0.000   
## 59 0.000 0.001 0.000 0.000   
## 60 0.001 0.001 0.033 0.007   
## 61 0.000 0.000 0.409 0.143   
## 62 0.007 0.010 0.012 0.020   
## 63 0.000 0.000 0.001 0.000   
## 64 0.004 0.004 0.008 0.004   
## 65 0.001 0.001 0.093 0.255   
## 66 0.000 0.000 0.223 0.343   
## 67 0.022 0.022 0.048 0.040   
## 68 0.101 0.104 0.073 0.037   
## 69 0.000 0.001 0.001 0.001   
## 70 0.001 0.000 0.002 0.000   
## 71 0.001 0.000 0.005 0.000   
## 72 0.000 0.000 0.001 0.000   
## 73 0.000 0.001 0.001 0.000   
## 74 0.170 0.166 0.018 0.022   
## 75 0.271 0.269 0.037 0.020   
## 76 0.008 0.010 0.002 0.001   
## 77 0.000 0.000 0.000 0.000   
## 78 0.000 0.000 0.000 0.000   
## 79 0.000 0.000 0.000 0.000   
## 80 0.023 0.022 0.002 0.005   
## 81 0.002 0.002 0.001 0.001   
## 82 0.000 0.000 0.000 0.000   
## 83 0.001 0.001 0.001 0.000   
## 84 0.000 0.000 0.000 0.000   
## contacttelephone contactunknown day monthfeb monthmar monthapr  
## 1 0.001 0.001 0.001 0.000 0.000 0.000   
## 2 0.004 0.000 0.000 0.000 0.001 0.000   
## 3 0.000 0.003 0.000 0.001 0.001 0.000   
## 4 0.001 0.001 0.000 0.000 0.000 0.001   
## 5 0.000 0.001 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.000 0.000 0.000 0.000   
## 7 0.000 0.000 0.000 0.000 0.000 0.000   
## 8 0.001 0.000 0.000 0.002 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000 0.000   
## 11 0.000 0.001 0.000 0.000 0.000 0.000   
## 12 0.001 0.005 0.000 0.001 0.000 0.001   
## 13 0.014 0.019 0.000 0.003 0.000 0.007   
## 14 0.011 0.013 0.000 0.001 0.001 0.008   
## 15 0.008 0.000 0.000 0.001 0.000 0.026   
## 16 0.019 0.012 0.000 0.000 0.001 0.003   
## 17 0.008 0.000 0.001 0.048 0.000 0.006   
## 18 0.006 0.000 0.000 0.032 0.001 0.037   
## 19 0.001 0.003 0.000 0.000 0.001 0.009   
## 20 0.001 0.000 0.000 0.003 0.000 0.000   
## 21 0.000 0.000 0.000 0.001 0.021 0.001   
## 22 0.009 0.001 0.000 0.007 0.003 0.000   
## 23 0.038 0.000 0.000 0.001 0.006 0.000   
## 24 0.001 0.000 0.000 0.001 0.028 0.000   
## 25 0.001 0.000 0.000 0.000 0.052 0.000   
## 26 0.000 0.000 0.000 0.001 0.059 0.001   
## 27 0.003 0.000 0.000 0.001 0.012 0.000   
## 28 0.000 0.000 0.000 0.000 0.007 0.000   
## 29 0.009 0.000 0.000 0.001 0.048 0.001   
## 30 0.005 0.000 0.000 0.002 0.025 0.000   
## 31 0.000 0.000 0.000 0.001 0.002 0.000   
## 32 0.002 0.000 0.000 0.001 0.000 0.000   
## 33 0.007 0.000 0.000 0.000 0.000 0.000   
## 34 0.003 0.000 0.000 0.000 0.003 0.000   
## 35 0.000 0.000 0.000 0.000 0.027 0.000   
## 36 0.003 0.000 0.000 0.000 0.002 0.000   
## 37 0.000 0.000 0.000 0.000 0.001 0.000   
## 38 0.004 0.000 0.000 0.000 0.045 0.000   
## 39 0.001 0.001 0.000 0.000 0.006 0.000   
## 40 0.000 0.000 0.000 0.000 0.008 0.000   
## 41 0.004 0.000 0.000 0.000 0.034 0.000   
## 42 0.007 0.000 0.000 0.000 0.006 0.000   
## 43 0.001 0.000 0.000 0.000 0.007 0.001   
## 44 0.000 0.000 0.000 0.000 0.094 0.000   
## 45 0.001 0.000 0.000 0.001 0.012 0.000   
## 46 0.005 0.001 0.000 0.000 0.000 0.001   
## 47 0.015 0.000 0.000 0.001 0.126 0.000   
## 48 0.030 0.000 0.000 0.001 0.000 0.000   
## 49 0.061 0.001 0.000 0.005 0.035 0.000   
## 50 0.003 0.002 0.000 0.010 0.008 0.000   
## 51 0.003 0.002 0.000 0.001 0.013 0.000   
## 52 0.010 0.000 0.000 0.002 0.000 0.001   
## 53 0.017 0.001 0.000 0.000 0.000 0.012   
## 54 0.097 0.001 0.003 0.017 0.000 0.001   
## 55 0.407 0.002 0.001 0.002 0.000 0.013   
## 56 0.062 0.000 0.001 0.100 0.000 0.005   
## 57 0.063 0.004 0.001 0.000 0.001 0.111   
## 58 0.007 0.000 0.001 0.000 0.008 0.012   
## 59 0.019 0.000 0.000 0.040 0.014 0.030   
## 60 0.000 0.000 0.008 0.002 0.001 0.001   
## 61 0.002 0.024 0.011 0.000 0.001 0.008   
## 62 0.005 0.023 0.029 0.004 0.000 0.000   
## 63 0.001 0.000 0.000 0.000 0.000 0.000   
## 64 0.013 0.686 0.012 0.020 0.006 0.007   
## 65 0.001 0.027 0.500 0.004 0.003 0.018   
## 66 0.000 0.101 0.156 0.003 0.000 0.006   
## 67 0.001 0.000 0.038 0.000 0.000 0.002   
## 68 0.001 0.010 0.050 0.000 0.001 0.008   
## 69 0.000 0.001 0.000 0.000 0.000 0.000   
## 70 0.000 0.000 0.000 0.000 0.000 0.000   
## 71 0.000 0.000 0.000 0.001 0.000 0.000   
## 72 0.000 0.000 0.000 0.000 0.000 0.000   
## 73 0.000 0.001 0.000 0.000 0.001 0.000   
## 74 0.000 0.012 0.001 0.011 0.006 0.019   
## 75 0.000 0.016 0.003 0.039 0.016 0.060   
## 76 0.000 0.001 0.000 0.002 0.001 0.003   
## 77 0.000 0.000 0.000 0.000 0.001 0.001   
## 78 0.000 0.000 0.000 0.000 0.000 0.000   
## 79 0.000 0.000 0.000 0.003 0.001 0.003   
## 80 0.001 0.016 0.169 0.594 0.229 0.553   
## 81 0.000 0.001 0.007 0.021 0.008 0.017   
## 82 0.000 0.000 0.000 0.000 0.000 0.000   
## 83 0.000 0.000 0.001 0.003 0.001 0.002   
## 84 0.000 0.000 0.000 0.000 0.000 0.000   
## monthmay monthjun monthjul monthaug monthsep monthoct monthnov monthdec  
## 1 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 2 0.000 0.000 0.000 0.000 0.001 0.002 0.000 0.001   
## 3 0.001 0.000 0.000 0.002 0.001 0.001 0.001 0.000   
## 4 0.001 0.000 0.000 0.001 0.000 0.000 0.001 0.000   
## 5 0.000 0.001 0.000 0.000 0.000 0.000 0.001 0.000   
## 6 0.000 0.000 0.000 0.001 0.000 0.000 0.001 0.000   
## 7 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 8 0.000 0.000 0.000 0.001 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 11 0.001 0.000 0.000 0.002 0.001 0.000 0.000 0.000   
## 12 0.000 0.004 0.000 0.000 0.000 0.000 0.001 0.000   
## 13 0.001 0.007 0.004 0.001 0.000 0.000 0.002 0.002   
## 14 0.001 0.003 0.000 0.001 0.002 0.002 0.005 0.003   
## 15 0.000 0.000 0.008 0.004 0.001 0.001 0.006 0.000   
## 16 0.000 0.007 0.010 0.006 0.007 0.004 0.009 0.003   
## 17 0.000 0.000 0.004 0.000 0.000 0.006 0.022 0.006   
## 18 0.000 0.000 0.000 0.001 0.000 0.000 0.008 0.019   
## 19 0.008 0.023 0.001 0.000 0.024 0.002 0.000 0.005   
## 20 0.001 0.002 0.000 0.000 0.006 0.003 0.003 0.005   
## 21 0.001 0.001 0.000 0.000 0.147 0.023 0.004 0.001   
## 22 0.000 0.000 0.002 0.003 0.001 0.003 0.002 0.206   
## 23 0.000 0.000 0.005 0.015 0.001 0.018 0.000 0.044   
## 24 0.000 0.000 0.000 0.000 0.058 0.069 0.000 0.006   
## 25 0.000 0.000 0.001 0.000 0.001 0.020 0.001 0.003   
## 26 0.000 0.000 0.001 0.001 0.001 0.000 0.001 0.009   
## 27 0.000 0.000 0.000 0.001 0.002 0.003 0.000 0.040   
## 28 0.000 0.000 0.001 0.000 0.002 0.020 0.000 0.002   
## 29 0.000 0.000 0.000 0.001 0.000 0.001 0.000 0.009   
## 30 0.000 0.000 0.001 0.000 0.003 0.003 0.001 0.012   
## 31 0.000 0.000 0.001 0.001 0.001 0.014 0.000 0.002   
## 32 0.000 0.000 0.001 0.000 0.000 0.021 0.001 0.001   
## 33 0.000 0.000 0.000 0.001 0.000 0.002 0.000 0.009   
## 34 0.000 0.000 0.000 0.000 0.003 0.012 0.000 0.000   
## 35 0.000 0.000 0.000 0.000 0.000 0.007 0.000 0.004   
## 36 0.000 0.000 0.001 0.001 0.011 0.009 0.000 0.000   
## 37 0.000 0.000 0.000 0.000 0.003 0.006 0.001 0.008   
## 38 0.000 0.000 0.000 0.000 0.005 0.001 0.000 0.000   
## 39 0.000 0.000 0.000 0.000 0.019 0.035 0.002 0.016   
## 40 0.000 0.000 0.002 0.001 0.001 0.003 0.001 0.000   
## 41 0.000 0.000 0.001 0.000 0.011 0.009 0.000 0.004   
## 42 0.000 0.000 0.001 0.000 0.003 0.003 0.000 0.002   
## 43 0.000 0.000 0.000 0.000 0.011 0.017 0.001 0.002   
## 44 0.000 0.000 0.001 0.000 0.007 0.035 0.000 0.000   
## 45 0.000 0.000 0.002 0.002 0.000 0.004 0.001 0.007   
## 46 0.000 0.000 0.001 0.000 0.037 0.071 0.000 0.007   
## 47 0.000 0.001 0.005 0.002 0.002 0.000 0.000 0.009   
## 48 0.000 0.000 0.003 0.001 0.000 0.021 0.001 0.000   
## 49 0.000 0.001 0.019 0.017 0.010 0.012 0.001 0.064   
## 50 0.000 0.000 0.002 0.009 0.006 0.001 0.003 0.223   
## 51 0.001 0.001 0.001 0.007 0.065 0.006 0.000 0.006   
## 52 0.000 0.000 0.000 0.000 0.006 0.029 0.002 0.003   
## 53 0.004 0.009 0.001 0.002 0.072 0.036 0.042 0.003   
## 54 0.004 0.033 0.004 0.001 0.027 0.007 0.002 0.024   
## 55 0.001 0.001 0.017 0.015 0.034 0.000 0.007 0.000   
## 56 0.001 0.005 0.008 0.001 0.005 0.008 0.037 0.009   
## 57 0.001 0.003 0.002 0.000 0.029 0.024 0.014 0.007   
## 58 0.000 0.002 0.001 0.002 0.000 0.001 0.009 0.002   
## 59 0.002 0.006 0.006 0.004 0.044 0.042 0.030 0.056   
## 60 0.001 0.001 0.000 0.000 0.005 0.005 0.005 0.001   
## 61 0.008 0.008 0.004 0.020 0.001 0.001 0.001 0.002   
## 62 0.000 0.000 0.002 0.021 0.000 0.001 0.001 0.000   
## 63 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 64 0.037 0.050 0.025 0.037 0.002 0.003 0.015 0.003   
## 65 0.001 0.002 0.009 0.019 0.001 0.013 0.020 0.001   
## 66 0.000 0.018 0.021 0.000 0.001 0.000 0.012 0.000   
## 67 0.007 0.000 0.007 0.001 0.000 0.000 0.003 0.000   
## 68 0.021 0.005 0.014 0.003 0.001 0.001 0.007 0.001   
## 69 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 70 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 71 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 72 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 73 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 74 0.027 0.021 0.019 0.016 0.008 0.009 0.018 0.003   
## 75 0.080 0.066 0.056 0.054 0.021 0.027 0.064 0.010   
## 76 0.005 0.003 0.003 0.003 0.001 0.001 0.003 0.000   
## 77 0.001 0.001 0.000 0.001 0.001 0.001 0.001 0.000   
## 78 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 79 0.004 0.004 0.004 0.003 0.000 0.001 0.003 0.000   
## 80 0.747 0.680 0.690 0.682 0.271 0.299 0.599 0.125   
## 81 0.025 0.025 0.023 0.023 0.009 0.013 0.020 0.004   
## 82 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 83 0.004 0.003 0.004 0.004 0.001 0.001 0.003 0.000   
## 84 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## campaign\_class2 campaign\_class3 campaign\_class4 campaign\_class5  
## 1 0.001 0.001 0.001 0.000   
## 2 0.000 0.000 0.000 0.000   
## 3 0.000 0.000 0.000 0.000   
## 4 0.000 0.000 0.000 0.000   
## 5 0.000 0.000 0.000 0.000   
## 6 0.000 0.000 0.000 0.000   
## 7 0.000 0.000 0.000 0.000   
## 8 0.000 0.000 0.000 0.000   
## 9 0.000 0.000 0.000 0.000   
## 10 0.000 0.000 0.000 0.000   
## 11 0.000 0.000 0.001 0.000   
## 12 0.000 0.000 0.000 0.000   
## 13 0.000 0.000 0.001 0.000   
## 14 0.001 0.000 0.000 0.000   
## 15 0.004 0.000 0.005 0.003   
## 16 0.001 0.001 0.010 0.001   
## 17 0.000 0.001 0.000 0.000   
## 18 0.000 0.000 0.003 0.000   
## 19 0.002 0.000 0.000 0.001   
## 20 0.001 0.002 0.001 0.003   
## 21 0.000 0.002 0.001 0.001   
## 22 0.014 0.064 0.005 0.002   
## 23 0.007 0.023 0.082 0.029   
## 24 0.000 0.003 0.005 0.000   
## 25 0.004 0.005 0.002 0.098   
## 26 0.003 0.008 0.010 0.008   
## 27 0.008 0.004 0.013 0.003   
## 28 0.010 0.016 0.001 0.107   
## 29 0.001 0.001 0.045 0.012   
## 30 0.049 0.018 0.078 0.005   
## 31 0.001 0.012 0.075 0.000   
## 32 0.013 0.116 0.001 0.034   
## 33 0.001 0.001 0.002 0.002   
## 34 0.000 0.013 0.003 0.003   
## 35 0.001 0.006 0.003 0.001   
## 36 0.002 0.010 0.000 0.024   
## 37 0.000 0.000 0.008 0.022   
## 38 0.000 0.006 0.011 0.065   
## 39 0.000 0.081 0.001 0.146   
## 40 0.033 0.033 0.101 0.004   
## 41 0.014 0.013 0.001 0.024   
## 42 0.009 0.007 0.011 0.006   
## 43 0.025 0.002 0.005 0.124   
## 44 0.000 0.001 0.034 0.001   
## 45 0.000 0.002 0.030 0.008   
## 46 0.003 0.008 0.010 0.002   
## 47 0.000 0.000 0.011 0.042   
## 48 0.034 0.006 0.020 0.002   
## 49 0.034 0.092 0.045 0.026   
## 50 0.001 0.032 0.014 0.010   
## 51 0.048 0.006 0.015 0.003   
## 52 0.008 0.003 0.002 0.003   
## 53 0.006 0.005 0.000 0.000   
## 54 0.001 0.003 0.002 0.001   
## 55 0.019 0.020 0.006 0.000   
## 56 0.002 0.000 0.000 0.000   
## 57 0.000 0.000 0.000 0.000   
## 58 0.003 0.003 0.004 0.002   
## 59 0.001 0.001 0.000 0.000   
## 60 0.250 0.149 0.108 0.058   
## 61 0.069 0.035 0.045 0.022   
## 62 0.295 0.173 0.154 0.077   
## 63 0.000 0.000 0.001 0.000   
## 64 0.001 0.000 0.000 0.000   
## 65 0.002 0.002 0.004 0.003   
## 66 0.003 0.004 0.005 0.002   
## 67 0.001 0.000 0.000 0.000   
## 68 0.003 0.001 0.001 0.000   
## 69 0.000 0.000 0.000 0.000   
## 70 0.000 0.000 0.000 0.000   
## 71 0.000 0.000 0.000 0.000   
## 72 0.000 0.000 0.000 0.000   
## 73 0.000 0.000 0.000 0.000   
## 74 0.001 0.001 0.001 0.001   
## 75 0.006 0.003 0.003 0.002   
## 76 0.000 0.000 0.000 0.000   
## 77 0.000 0.000 0.000 0.000   
## 78 0.000 0.000 0.000 0.000   
## 79 0.000 0.000 0.000 0.000   
## 80 0.000 0.000 0.001 0.002   
## 81 0.000 0.000 0.000 0.000   
## 82 0.000 0.000 0.000 0.000   
## 83 0.000 0.000 0.000 0.000   
## 84 0.000 0.000 0.000 0.000   
## campaign\_classup 5 pdays\_class(0,91] pdays\_class(91,108]  
## 1 0.001 0.000 0.000   
## 2 0.000 0.001 0.001   
## 3 0.000 0.002 0.003   
## 4 0.001 0.000 0.000   
## 5 0.000 0.000 0.000   
## 6 0.000 0.000 0.000   
## 7 0.000 0.000 0.000   
## 8 0.000 0.000 0.000   
## 9 0.000 0.000 0.000   
## 10 0.000 0.000 0.000   
## 11 0.001 0.000 0.000   
## 12 0.000 0.000 0.000   
## 13 0.001 0.000 0.000   
## 14 0.000 0.002 0.007   
## 15 0.031 0.000 0.001   
## 16 0.027 0.001 0.022   
## 17 0.004 0.007 0.010   
## 18 0.000 0.005 0.001   
## 19 0.025 0.000 0.018   
## 20 0.005 0.014 0.006   
## 21 0.000 0.005 0.120   
## 22 0.000 0.014 0.050   
## 23 0.002 0.048 0.003   
## 24 0.015 0.008 0.015   
## 25 0.000 0.111 0.019   
## 26 0.025 0.000 0.018   
## 27 0.003 0.004 0.012   
## 28 0.005 0.000 0.002   
## 29 0.007 0.002 0.005   
## 30 0.001 0.002 0.001   
## 31 0.007 0.071 0.015   
## 32 0.000 0.046 0.014   
## 33 0.000 0.005 0.005   
## 34 0.021 0.007 0.008   
## 35 0.005 0.036 0.004   
## 36 0.002 0.001 0.000   
## 37 0.001 0.082 0.017   
## 38 0.030 0.003 0.006   
## 39 0.008 0.003 0.008   
## 40 0.012 0.001 0.001   
## 41 0.029 0.001 0.023   
## 42 0.017 0.097 0.018   
## 43 0.002 0.046 0.021   
## 44 0.010 0.001 0.001   
## 45 0.084 0.000 0.004   
## 46 0.049 0.050 0.019   
## 47 0.002 0.039 0.070   
## 48 0.034 0.003 0.004   
## 49 0.014 0.014 0.002   
## 50 0.002 0.078 0.061   
## 51 0.054 0.013 0.015   
## 52 0.009 0.002 0.003   
## 53 0.010 0.052 0.054   
## 54 0.003 0.010 0.050   
## 55 0.021 0.009 0.053   
## 56 0.000 0.001 0.025   
## 57 0.000 0.000 0.061   
## 58 0.003 0.000 0.015   
## 59 0.001 0.067 0.089   
## 60 0.128 0.003 0.000   
## 61 0.053 0.001 0.001   
## 62 0.188 0.002 0.003   
## 63 0.000 0.000 0.000   
## 64 0.001 0.022 0.011   
## 65 0.018 0.001 0.000   
## 66 0.011 0.001 0.000   
## 67 0.000 0.000 0.001   
## 68 0.000 0.003 0.001   
## 69 0.000 0.000 0.000   
## 70 0.000 0.000 0.000   
## 71 0.000 0.000 0.000   
## 72 0.000 0.000 0.000   
## 73 0.000 0.000 0.000   
## 74 0.002 0.000 0.000   
## 75 0.003 0.000 0.000   
## 76 0.000 0.000 0.000   
## 77 0.000 0.000 0.000   
## 78 0.000 0.000 0.000   
## 79 0.000 0.000 0.000   
## 80 0.008 0.001 0.000   
## 81 0.000 0.000 0.000   
## 82 0.000 0.000 0.000   
## 83 0.000 0.000 0.000   
## 84 0.000 0.000 0.000   
## pdays\_class(108,159] pdays\_class(159,181] pdays\_class (181,194]  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.001   
## 3 0.001 0.000 0.001   
## 4 0.000 0.000 0.000   
## 5 0.001 0.000 0.000   
## 6 0.001 0.001 0.001   
## 7 0.000 0.000 0.000   
## 8 0.000 0.000 0.000   
## 9 0.000 0.000 0.000   
## 10 0.000 0.000 0.000   
## 11 0.000 0.000 0.000   
## 12 0.001 0.001 0.002   
## 13 0.012 0.001 0.002   
## 14 0.017 0.002 0.009   
## 15 0.031 0.006 0.001   
## 16 0.015 0.002 0.015   
## 17 0.029 0.010 0.002   
## 18 0.001 0.006 0.027   
## 19 0.016 0.016 0.000   
## 20 0.005 0.000 0.012   
## 21 0.000 0.119 0.003   
## 22 0.000 0.016 0.065   
## 23 0.000 0.020 0.062   
## 24 0.000 0.002 0.037   
## 25 0.005 0.014 0.038   
## 26 0.096 0.078 0.005   
## 27 0.002 0.005 0.000   
## 28 0.002 0.000 0.011   
## 29 0.016 0.032 0.015   
## 30 0.001 0.002 0.008   
## 31 0.016 0.039 0.010   
## 32 0.013 0.002 0.004   
## 33 0.017 0.020 0.002   
## 34 0.018 0.000 0.002   
## 35 0.000 0.003 0.018   
## 36 0.016 0.006 0.016   
## 37 0.017 0.033 0.006   
## 38 0.040 0.004 0.039   
## 39 0.003 0.016 0.030   
## 40 0.007 0.001 0.019   
## 41 0.053 0.001 0.002   
## 42 0.018 0.017 0.027   
## 43 0.007 0.004 0.002   
## 44 0.083 0.072 0.048   
## 45 0.002 0.001 0.002   
## 46 0.007 0.017 0.049   
## 47 0.004 0.060 0.004   
## 48 0.030 0.003 0.015   
## 49 0.006 0.001 0.046   
## 50 0.005 0.010 0.090   
## 51 0.051 0.042 0.004   
## 52 0.011 0.039 0.001   
## 53 0.003 0.164 0.003   
## 54 0.000 0.028 0.002   
## 55 0.007 0.000 0.001   
## 56 0.127 0.003 0.002   
## 57 0.000 0.000 0.040   
## 58 0.002 0.000 0.004   
## 59 0.177 0.047 0.185   
## 60 0.000 0.001 0.000   
## 61 0.000 0.003 0.001   
## 62 0.000 0.001 0.000   
## 63 0.000 0.000 0.000   
## 64 0.003 0.024 0.004   
## 65 0.001 0.000 0.001   
## 66 0.001 0.001 0.001   
## 67 0.000 0.000 0.000   
## 68 0.001 0.002 0.000   
## 69 0.000 0.000 0.000   
## 70 0.000 0.000 0.000   
## 71 0.000 0.000 0.000   
## 72 0.000 0.000 0.000   
## 73 0.000 0.000 0.000   
## 74 0.000 0.000 0.000   
## 75 0.000 0.000 0.000   
## 76 0.000 0.000 0.000   
## 77 0.000 0.000 0.000   
## 78 0.000 0.000 0.000   
## 79 0.000 0.000 0.000   
## 80 0.000 0.000 0.001   
## 81 0.000 0.000 0.000   
## 82 0.000 0.000 0.000   
## 83 0.000 0.000 0.000   
## 84 0.000 0.000 0.000   
## pdays\_class(194,258] pdays\_class(258,300] pdays\_class(300,343]  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.000   
## 3 0.001 0.000 0.000   
## 4 0.000 0.001 0.002   
## 5 0.000 0.000 0.000   
## 6 0.000 0.000 0.000   
## 7 0.000 0.000 0.000   
## 8 0.000 0.000 0.000   
## 9 0.000 0.000 0.000   
## 10 0.000 0.000 0.000   
## 11 0.000 0.000 0.000   
## 12 0.001 0.002 0.002   
## 13 0.011 0.002 0.008   
## 14 0.002 0.001 0.008   
## 15 0.000 0.002 0.059   
## 16 0.003 0.001 0.015   
## 17 0.106 0.072 0.019   
## 18 0.088 0.005 0.123   
## 19 0.006 0.003 0.011   
## 20 0.004 0.000 0.000   
## 21 0.000 0.005 0.007   
## 22 0.014 0.075 0.009   
## 23 0.005 0.007 0.003   
## 24 0.017 0.027 0.000   
## 25 0.021 0.019 0.007   
## 26 0.002 0.046 0.021   
## 27 0.001 0.008 0.002   
## 28 0.000 0.001 0.000   
## 29 0.004 0.019 0.003   
## 30 0.012 0.004 0.000   
## 31 0.002 0.001 0.012   
## 32 0.018 0.060 0.004   
## 33 0.001 0.001 0.002   
## 34 0.000 0.001 0.001   
## 35 0.000 0.005 0.001   
## 36 0.016 0.009 0.003   
## 37 0.000 0.060 0.002   
## 38 0.024 0.022 0.006   
## 39 0.013 0.040 0.000   
## 40 0.004 0.003 0.000   
## 41 0.037 0.049 0.027   
## 42 0.003 0.022 0.008   
## 43 0.006 0.038 0.003   
## 44 0.012 0.003 0.005   
## 45 0.003 0.017 0.002   
## 46 0.022 0.018 0.004   
## 47 0.002 0.001 0.000   
## 48 0.045 0.001 0.038   
## 49 0.001 0.012 0.000   
## 50 0.009 0.061 0.004   
## 51 0.001 0.008 0.011   
## 52 0.014 0.001 0.003   
## 53 0.001 0.004 0.034   
## 54 0.075 0.004 0.008   
## 55 0.014 0.012 0.021   
## 56 0.161 0.089 0.013   
## 57 0.003 0.020 0.335   
## 58 0.001 0.019 0.032   
## 59 0.163 0.066 0.068   
## 60 0.001 0.000 0.000   
## 61 0.006 0.010 0.005   
## 62 0.000 0.003 0.002   
## 63 0.000 0.000 0.000   
## 64 0.005 0.029 0.035   
## 65 0.003 0.000 0.002   
## 66 0.008 0.006 0.003   
## 67 0.001 0.000 0.000   
## 68 0.001 0.001 0.001   
## 69 0.000 0.000 0.001   
## 70 0.000 0.000 0.000   
## 71 0.000 0.000 0.000   
## 72 0.000 0.000 0.000   
## 73 0.000 0.000 0.000   
## 74 0.001 0.000 0.001   
## 75 0.002 0.000 0.001   
## 76 0.000 0.000 0.000   
## 77 0.000 0.000 0.000   
## 78 0.000 0.000 0.000   
## 79 0.000 0.000 0.000   
## 80 0.023 0.003 0.000   
## 81 0.001 0.000 0.000   
## 82 0.000 0.001 0.000   
## 83 0.000 0.000 0.000   
## 84 0.000 0.000 0.000   
## pdays\_class(343,362] pdays\_class(362,871]  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.003 0.001   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.003 0.001   
## 12 0.000 0.000   
## 13 0.000 0.001   
## 14 0.000 0.000   
## 15 0.001 0.000   
## 16 0.001 0.004   
## 17 0.000 0.000   
## 18 0.003 0.000   
## 19 0.106 0.078   
## 20 0.011 0.000   
## 21 0.024 0.040   
## 22 0.007 0.020   
## 23 0.000 0.000   
## 24 0.000 0.003   
## 25 0.008 0.003   
## 26 0.001 0.029   
## 27 0.033 0.016   
## 28 0.025 0.008   
## 29 0.041 0.040   
## 30 0.055 0.081   
## 31 0.013 0.000   
## 32 0.013 0.002   
## 33 0.017 0.044   
## 34 0.015 0.091   
## 35 0.006 0.007   
## 36 0.009 0.000   
## 37 0.024 0.039   
## 38 0.056 0.000   
## 39 0.012 0.000   
## 40 0.006 0.000   
## 41 0.001 0.014   
## 42 0.004 0.047   
## 43 0.008 0.003   
## 44 0.004 0.012   
## 45 0.003 0.012   
## 46 0.044 0.013   
## 47 0.059 0.008   
## 48 0.006 0.061   
## 49 0.002 0.012   
## 50 0.001 0.000   
## 51 0.004 0.089   
## 52 0.000 0.013   
## 53 0.091 0.000   
## 54 0.078 0.066   
## 55 0.030 0.049   
## 56 0.003 0.001   
## 57 0.035 0.000   
## 58 0.001 0.001   
## 59 0.001 0.012   
## 60 0.000 0.001   
## 61 0.004 0.010   
## 62 0.005 0.001   
## 63 0.000 0.000   
## 64 0.101 0.050   
## 65 0.000 0.000   
## 66 0.008 0.006   
## 67 0.001 0.001   
## 68 0.003 0.003   
## 69 0.000 0.000   
## 70 0.000 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.003 0.001   
## 75 0.004 0.001   
## 76 0.000 0.000   
## 77 0.000 0.000   
## 78 0.000 0.000   
## 79 0.000 0.000   
## 80 0.000 0.000   
## 81 0.000 0.000   
## 82 0.000 0.000   
## 83 0.000 0.000   
## 84 0.000 0.000   
## age\_class(30,59]:maritalmarried age\_class(59,95]:maritalmarried  
## 1 0.000 0.000   
## 2 0.000 0.004   
## 3 0.000 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.000 0.001   
## 13 0.000 0.000   
## 14 0.000 0.000   
## 15 0.000 0.000   
## 16 0.000 0.000   
## 17 0.000 0.000   
## 18 0.000 0.000   
## 19 0.000 0.000   
## 20 0.000 0.001   
## 21 0.000 0.000   
## 22 0.000 0.000   
## 23 0.000 0.000   
## 24 0.000 0.000   
## 25 0.000 0.000   
## 26 0.000 0.000   
## 27 0.000 0.000   
## 28 0.000 0.000   
## 29 0.000 0.000   
## 30 0.000 0.000   
## 31 0.000 0.000   
## 32 0.000 0.000   
## 33 0.000 0.000   
## 34 0.000 0.000   
## 35 0.000 0.000   
## 36 0.000 0.000   
## 37 0.000 0.000   
## 38 0.000 0.000   
## 39 0.000 0.000   
## 40 0.000 0.000   
## 41 0.000 0.000   
## 42 0.000 0.000   
## 43 0.000 0.000   
## 44 0.000 0.000   
## 45 0.000 0.000   
## 46 0.000 0.000   
## 47 0.000 0.000   
## 48 0.000 0.000   
## 49 0.000 0.000   
## 50 0.000 0.000   
## 51 0.000 0.000   
## 52 0.000 0.001   
## 53 0.000 0.000   
## 54 0.000 0.000   
## 55 0.000 0.000   
## 56 0.000 0.000   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.001 0.001   
## 61 0.000 0.000   
## 62 0.000 0.000   
## 63 0.005 0.004   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.001 0.007   
## 67 0.001 0.240   
## 68 0.005 0.057   
## 69 0.000 0.000   
## 70 0.003 0.016   
## 71 0.000 0.001   
## 72 0.000 0.000   
## 73 0.001 0.003   
## 74 0.003 0.008   
## 75 0.006 0.004   
## 76 0.002 0.002   
## 77 0.000 0.000   
## 78 0.000 0.000   
## 79 0.000 0.023   
## 80 0.003 0.003   
## 81 0.052 0.033   
## 82 0.004 0.002   
## 83 0.913 0.582   
## 84 0.000 0.000   
## age\_class(30,59]:maritalsingle age\_class(59,95]:maritalsingle  
## 1 0.000 0.000   
## 2 0.000 0.001   
## 3 0.001 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.001 0.001   
## 12 0.000 0.002   
## 13 0.000 0.000   
## 14 0.000 0.002   
## 15 0.000 0.001   
## 16 0.001 0.002   
## 17 0.000 0.001   
## 18 0.000 0.002   
## 19 0.000 0.010   
## 20 0.000 0.199   
## 21 0.000 0.019   
## 22 0.000 0.005   
## 23 0.000 0.012   
## 24 0.000 0.001   
## 25 0.000 0.001   
## 26 0.000 0.001   
## 27 0.000 0.003   
## 28 0.000 0.003   
## 29 0.000 0.005   
## 30 0.000 0.002   
## 31 0.000 0.001   
## 32 0.000 0.000   
## 33 0.000 0.000   
## 34 0.000 0.006   
## 35 0.000 0.000   
## 36 0.000 0.000   
## 37 0.000 0.001   
## 38 0.000 0.007   
## 39 0.000 0.000   
## 40 0.000 0.001   
## 41 0.000 0.000   
## 42 0.000 0.001   
## 43 0.000 0.002   
## 44 0.000 0.005   
## 45 0.000 0.004   
## 46 0.000 0.006   
## 47 0.000 0.002   
## 48 0.000 0.005   
## 49 0.000 0.002   
## 50 0.000 0.004   
## 51 0.000 0.044   
## 52 0.000 0.268   
## 53 0.000 0.010   
## 54 0.000 0.003   
## 55 0.000 0.004   
## 56 0.000 0.001   
## 57 0.000 0.000   
## 58 0.002 0.002   
## 59 0.000 0.000   
## 60 0.005 0.001   
## 61 0.000 0.000   
## 62 0.007 0.002   
## 63 0.005 0.002   
## 64 0.002 0.000   
## 65 0.000 0.000   
## 66 0.002 0.001   
## 67 0.000 0.085   
## 68 0.000 0.030   
## 69 0.001 0.000   
## 70 0.009 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.004 0.000   
## 74 0.017 0.001   
## 75 0.004 0.001   
## 76 0.005 0.001   
## 77 0.000 0.000   
## 78 0.000 0.000   
## 79 0.000 0.018   
## 80 0.001 0.000   
## 81 0.033 0.007   
## 82 0.004 0.001   
## 83 0.892 0.200   
## 84 0.000 0.000   
## age\_class(30,59]:jobblue-collar age\_class(59,95]:jobblue-collar  
## 1 0.000 0.000   
## 2 0.000 0.001   
## 3 0.003 0.000   
## 4 0.001 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.001 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.001 0.000   
## 11 0.001 0.001   
## 12 0.000 0.005   
## 13 0.000 0.003   
## 14 0.001 0.000   
## 15 0.000 0.001   
## 16 0.000 0.000   
## 17 0.000 0.001   
## 18 0.001 0.001   
## 19 0.000 0.001   
## 20 0.000 0.014   
## 21 0.000 0.006   
## 22 0.000 0.003   
## 23 0.000 0.044   
## 24 0.000 0.032   
## 25 0.000 0.000   
## 26 0.000 0.024   
## 27 0.000 0.002   
## 28 0.000 0.007   
## 29 0.000 0.002   
## 30 0.000 0.001   
## 31 0.000 0.028   
## 32 0.000 0.004   
## 33 0.000 0.014   
## 34 0.000 0.003   
## 35 0.000 0.009   
## 36 0.000 0.008   
## 37 0.000 0.007   
## 38 0.000 0.005   
## 39 0.000 0.001   
## 40 0.000 0.010   
## 41 0.000 0.000   
## 42 0.000 0.017   
## 43 0.000 0.008   
## 44 0.000 0.025   
## 45 0.000 0.046   
## 46 0.000 0.016   
## 47 0.000 0.014   
## 48 0.000 0.001   
## 49 0.000 0.000   
## 50 0.000 0.001   
## 51 0.000 0.035   
## 52 0.000 0.011   
## 53 0.000 0.000   
## 54 0.000 0.000   
## 55 0.000 0.000   
## 56 0.000 0.000   
## 57 0.000 0.000   
## 58 0.000 0.001   
## 59 0.000 0.000   
## 60 0.004 0.000   
## 61 0.000 0.000   
## 62 0.001 0.000   
## 63 0.018 0.002   
## 64 0.001 0.000   
## 65 0.000 0.000   
## 66 0.003 0.001   
## 67 0.001 0.061   
## 68 0.006 0.022   
## 69 0.006 0.001   
## 70 0.134 0.007   
## 71 0.032 0.002   
## 72 0.005 0.000   
## 73 0.000 0.000   
## 74 0.081 0.007   
## 75 0.017 0.002   
## 76 0.000 0.000   
## 77 0.011 0.011   
## 78 0.000 0.000   
## 79 0.007 0.401   
## 80 0.029 0.008   
## 81 0.609 0.056   
## 82 0.004 0.004   
## 83 0.018 0.002   
## 84 0.000 0.000   
## age\_class(30,59]:jobentrepreneur age\_class(59,95]:jobentrepreneur  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.001 0.000   
## 5 0.004 0.000   
## 6 0.005 0.001   
## 7 0.007 0.001   
## 8 0.000 0.000   
## 9 0.001 0.000   
## 10 0.000 0.000   
## 11 0.002 0.001   
## 12 0.000 0.002   
## 13 0.000 0.001   
## 14 0.000 0.000   
## 15 0.000 0.001   
## 16 0.000 0.000   
## 17 0.000 0.001   
## 18 0.000 0.000   
## 19 0.000 0.001   
## 20 0.000 0.004   
## 21 0.000 0.000   
## 22 0.000 0.000   
## 23 0.000 0.019   
## 24 0.000 0.051   
## 25 0.000 0.003   
## 26 0.000 0.000   
## 27 0.000 0.021   
## 28 0.000 0.016   
## 29 0.000 0.150   
## 30 0.000 0.000   
## 31 0.000 0.008   
## 32 0.000 0.000   
## 33 0.000 0.036   
## 34 0.000 0.029   
## 35 0.000 0.052   
## 36 0.000 0.000   
## 37 0.000 0.012   
## 38 0.000 0.042   
## 39 0.000 0.000   
## 40 0.000 0.002   
## 41 0.000 0.002   
## 42 0.000 0.000   
## 43 0.000 0.001   
## 44 0.000 0.003   
## 45 0.000 0.027   
## 46 0.000 0.093   
## 47 0.000 0.003   
## 48 0.000 0.006   
## 49 0.000 0.004   
## 50 0.000 0.020   
## 51 0.000 0.002   
## 52 0.000 0.001   
## 53 0.000 0.000   
## 54 0.000 0.000   
## 55 0.000 0.012   
## 56 0.000 0.000   
## 57 0.000 0.001   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.001 0.000   
## 61 0.000 0.000   
## 62 0.000 0.000   
## 63 0.004 0.000   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.001 0.000   
## 67 0.001 0.015   
## 68 0.002 0.006   
## 69 0.000 0.000   
## 70 0.001 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.024 0.004   
## 75 0.098 0.015   
## 76 0.659 0.102   
## 77 0.005 0.006   
## 78 0.000 0.000   
## 79 0.002 0.185   
## 80 0.008 0.005   
## 81 0.167 0.028   
## 82 0.001 0.002   
## 83 0.005 0.001   
## 84 0.000 0.000   
## age\_class(30,59]:jobhousemaid age\_class(59,95]:jobhousemaid  
## 1 0.000 0.000   
## 2 0.000 0.001   
## 3 0.000 0.000   
## 4 0.005 0.001   
## 5 0.000 0.000   
## 6 0.004 0.002   
## 7 0.000 0.000   
## 8 0.001 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.001 0.002   
## 12 0.000 0.003   
## 13 0.000 0.000   
## 14 0.000 0.000   
## 15 0.000 0.000   
## 16 0.000 0.000   
## 17 0.000 0.000   
## 18 0.000 0.000   
## 19 0.000 0.000   
## 20 0.000 0.001   
## 21 0.000 0.002   
## 22 0.000 0.001   
## 23 0.000 0.002   
## 24 0.000 0.002   
## 25 0.000 0.005   
## 26 0.000 0.000   
## 27 0.001 0.045   
## 28 0.001 0.028   
## 29 0.000 0.001   
## 30 0.000 0.000   
## 31 0.000 0.027   
## 32 0.000 0.011   
## 33 0.000 0.001   
## 34 0.000 0.001   
## 35 0.000 0.000   
## 36 0.000 0.000   
## 37 0.001 0.040   
## 38 0.000 0.002   
## 39 0.000 0.002   
## 40 0.000 0.002   
## 41 0.000 0.015   
## 42 0.000 0.002   
## 43 0.000 0.018   
## 44 0.000 0.000   
## 45 0.000 0.001   
## 46 0.000 0.004   
## 47 0.000 0.007   
## 48 0.000 0.010   
## 49 0.000 0.001   
## 50 0.000 0.001   
## 51 0.000 0.000   
## 52 0.000 0.000   
## 53 0.000 0.003   
## 54 0.000 0.001   
## 55 0.000 0.000   
## 56 0.000 0.001   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.000 0.000   
## 61 0.000 0.000   
## 62 0.000 0.000   
## 63 0.002 0.000   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.001 0.000   
## 67 0.001 0.020   
## 68 0.001 0.009   
## 69 0.000 0.000   
## 70 0.000 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.000 0.000   
## 75 0.001 0.000   
## 76 0.000 0.000   
## 77 0.654 0.212   
## 78 0.175 0.084   
## 79 0.002 0.346   
## 80 0.013 0.013   
## 81 0.131 0.059   
## 82 0.001 0.003   
## 83 0.003 0.002   
## 84 0.000 0.000   
## age\_class(30,59]:jobmanagement age\_class(59,95]:jobmanagement  
## 1 0.000 0.000   
## 2 0.000 0.001   
## 3 0.001 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.001 0.000   
## 8 0.004 0.000   
## 9 0.001 0.000   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.000 0.007   
## 13 0.000 0.002   
## 14 0.001 0.001   
## 15 0.000 0.001   
## 16 0.000 0.001   
## 17 0.000 0.000   
## 18 0.000 0.001   
## 19 0.000 0.002   
## 20 0.000 0.000   
## 21 0.000 0.003   
## 22 0.000 0.002   
## 23 0.000 0.000   
## 24 0.000 0.002   
## 25 0.000 0.004   
## 26 0.000 0.006   
## 27 0.000 0.001   
## 28 0.000 0.008   
## 29 0.000 0.003   
## 30 0.000 0.000   
## 31 0.000 0.004   
## 32 0.000 0.002   
## 33 0.000 0.051   
## 34 0.000 0.003   
## 35 0.000 0.013   
## 36 0.000 0.008   
## 37 0.000 0.011   
## 38 0.000 0.000   
## 39 0.000 0.016   
## 40 0.000 0.000   
## 41 0.000 0.027   
## 42 0.000 0.032   
## 43 0.000 0.001   
## 44 0.000 0.000   
## 45 0.000 0.000   
## 46 0.000 0.001   
## 47 0.000 0.000   
## 48 0.000 0.000   
## 49 0.000 0.002   
## 50 0.000 0.001   
## 51 0.000 0.001   
## 52 0.000 0.001   
## 53 0.000 0.002   
## 54 0.000 0.000   
## 55 0.000 0.000   
## 56 0.000 0.000   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.004 0.000   
## 61 0.000 0.000   
## 62 0.002 0.000   
## 63 0.015 0.002   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.001 0.002   
## 67 0.004 0.059   
## 68 0.003 0.022   
## 69 0.004 0.001   
## 70 0.004 0.002   
## 71 0.000 0.000   
## 72 0.003 0.000   
## 73 0.170 0.021   
## 74 0.042 0.005   
## 75 0.043 0.004   
## 76 0.042 0.005   
## 77 0.014 0.016   
## 78 0.001 0.000   
## 79 0.007 0.541   
## 80 0.031 0.013   
## 81 0.574 0.073   
## 82 0.004 0.006   
## 83 0.019 0.003   
## 84 0.000 0.000   
## age\_class(30,59]:jobretired age\_class(59,95]:jobretired  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.001 0.000   
## 13 0.000 0.000   
## 14 0.000 0.000   
## 15 0.000 0.000   
## 16 0.000 0.000   
## 17 0.000 0.000   
## 18 0.000 0.000   
## 19 0.000 0.000   
## 20 0.000 0.000   
## 21 0.000 0.000   
## 22 0.000 0.000   
## 23 0.000 0.000   
## 24 0.000 0.000   
## 25 0.000 0.000   
## 26 0.000 0.000   
## 27 0.000 0.000   
## 28 0.000 0.000   
## 29 0.000 0.000   
## 30 0.000 0.000   
## 31 0.000 0.000   
## 32 0.000 0.000   
## 33 0.000 0.000   
## 34 0.000 0.000   
## 35 0.000 0.000   
## 36 0.000 0.000   
## 37 0.000 0.000   
## 38 0.000 0.000   
## 39 0.000 0.000   
## 40 0.000 0.000   
## 41 0.000 0.000   
## 42 0.000 0.000   
## 43 0.000 0.000   
## 44 0.000 0.000   
## 45 0.000 0.000   
## 46 0.000 0.000   
## 47 0.000 0.000   
## 48 0.000 0.000   
## 49 0.000 0.000   
## 50 0.000 0.000   
## 51 0.000 0.000   
## 52 0.000 0.000   
## 53 0.000 0.000   
## 54 0.000 0.000   
## 55 0.000 0.000   
## 56 0.000 0.000   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.000 0.000   
## 61 0.000 0.000   
## 62 0.000 0.000   
## 63 0.000 0.000   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.000 0.000   
## 67 0.001 0.003   
## 68 0.000 0.001   
## 69 0.000 0.000   
## 70 0.000 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.000 0.000   
## 75 0.000 0.000   
## 76 0.000 0.000   
## 77 0.000 0.001   
## 78 0.000 0.000   
## 79 0.005 0.038   
## 80 0.000 0.000   
## 81 0.000 0.000   
## 82 0.987 0.951   
## 83 0.003 0.004   
## 84 0.000 0.000   
## age\_class(30,59]:jobself-employed age\_class(59,95]:jobself-employed  
## 1 0.000 0.000   
## 2 0.000 0.001   
## 3 0.000 0.000   
## 4 0.001 0.000   
## 5 0.000 0.000   
## 6 0.001 0.000   
## 7 0.001 0.000   
## 8 0.000 0.000   
## 9 0.025 0.004   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.000 0.004   
## 13 0.000 0.001   
## 14 0.000 0.001   
## 15 0.000 0.000   
## 16 0.000 0.000   
## 17 0.000 0.002   
## 18 0.000 0.000   
## 19 0.000 0.006   
## 20 0.001 0.078   
## 21 0.000 0.000   
## 22 0.000 0.000   
## 23 0.000 0.000   
## 24 0.000 0.000   
## 25 0.000 0.059   
## 26 0.000 0.002   
## 27 0.000 0.003   
## 28 0.000 0.024   
## 29 0.000 0.005   
## 30 0.000 0.005   
## 31 0.000 0.000   
## 32 0.000 0.011   
## 33 0.000 0.035   
## 34 0.000 0.011   
## 35 0.000 0.020   
## 36 0.000 0.000   
## 37 0.000 0.014   
## 38 0.000 0.005   
## 39 0.000 0.028   
## 40 0.000 0.018   
## 41 0.000 0.004   
## 42 0.000 0.042   
## 43 0.000 0.001   
## 44 0.000 0.000   
## 45 0.000 0.018   
## 46 0.000 0.000   
## 47 0.000 0.006   
## 48 0.000 0.004   
## 49 0.000 0.006   
## 50 0.000 0.007   
## 51 0.000 0.006   
## 52 0.001 0.106   
## 53 0.000 0.004   
## 54 0.000 0.000   
## 55 0.000 0.003   
## 56 0.000 0.000   
## 57 0.000 0.002   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.001 0.000   
## 61 0.000 0.000   
## 62 0.001 0.000   
## 63 0.006 0.001   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.000 0.001   
## 67 0.002 0.026   
## 68 0.001 0.009   
## 69 0.003 0.001   
## 70 0.001 0.001   
## 71 0.003 0.000   
## 72 0.621 0.096   
## 73 0.103 0.015   
## 74 0.000 0.000   
## 75 0.002 0.000   
## 76 0.005 0.001   
## 77 0.004 0.007   
## 78 0.000 0.000   
## 79 0.002 0.249   
## 80 0.009 0.006   
## 81 0.196 0.033   
## 82 0.001 0.003   
## 83 0.006 0.001   
## 84 0.000 0.000   
## age\_class(30,59]:jobservices age\_class(59,95]:jobservices  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.001 0.000   
## 5 0.000 0.000   
## 6 0.000 0.000   
## 7 0.000 0.000   
## 8 0.002 0.000   
## 9 0.000 0.000   
## 10 0.017 0.001   
## 11 0.001 0.000   
## 12 0.000 0.003   
## 13 0.001 0.002   
## 14 0.002 0.000   
## 15 0.000 0.000   
## 16 0.000 0.000   
## 17 0.000 0.000   
## 18 0.000 0.002   
## 19 0.000 0.000   
## 20 0.000 0.067   
## 21 0.000 0.014   
## 22 0.000 0.020   
## 23 0.000 0.000   
## 24 0.000 0.049   
## 25 0.000 0.020   
## 26 0.000 0.017   
## 27 0.000 0.000   
## 28 0.000 0.000   
## 29 0.000 0.002   
## 30 0.000 0.001   
## 31 0.000 0.017   
## 32 0.000 0.065   
## 33 0.000 0.007   
## 34 0.000 0.039   
## 35 0.000 0.008   
## 36 0.000 0.001   
## 37 0.000 0.019   
## 38 0.000 0.056   
## 39 0.000 0.030   
## 40 0.000 0.025   
## 41 0.000 0.000   
## 42 0.000 0.047   
## 43 0.000 0.002   
## 44 0.000 0.010   
## 45 0.000 0.014   
## 46 0.000 0.039   
## 47 0.000 0.004   
## 48 0.000 0.009   
## 49 0.000 0.007   
## 50 0.000 0.009   
## 51 0.000 0.003   
## 52 0.000 0.089   
## 53 0.000 0.000   
## 54 0.000 0.005   
## 55 0.000 0.010   
## 56 0.000 0.001   
## 57 0.000 0.000   
## 58 0.001 0.000   
## 59 0.000 0.001   
## 60 0.005 0.000   
## 61 0.000 0.000   
## 62 0.004 0.000   
## 63 0.014 0.001   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.000 0.002   
## 67 0.000 0.027   
## 68 0.000 0.006   
## 69 0.461 0.023   
## 70 0.001 0.000   
## 71 0.003 0.000   
## 72 0.001 0.000   
## 73 0.004 0.000   
## 74 0.001 0.000   
## 75 0.000 0.000   
## 76 0.003 0.000   
## 77 0.006 0.005   
## 78 0.000 0.000   
## 79 0.005 0.186   
## 80 0.020 0.004   
## 81 0.410 0.023   
## 82 0.003 0.002   
## 83 0.030 0.002   
## 84 0.000 0.000   
## age\_class(30,59]:jobstudent age\_class(30,59]:jobtechnician  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.003 0.001   
## 4 0.001 0.000   
## 5 0.000 0.000   
## 6 0.000 0.002   
## 7 0.000 0.004   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.002   
## 11 0.018 0.002   
## 12 0.008 0.000   
## 13 0.011 0.000   
## 14 0.016 0.000   
## 15 0.079 0.000   
## 16 0.040 0.000   
## 17 0.002 0.000   
## 18 0.032 0.000   
## 19 0.001 0.000   
## 20 0.002 0.000   
## 21 0.013 0.000   
## 22 0.000 0.000   
## 23 0.000 0.000   
## 24 0.000 0.000   
## 25 0.002 0.000   
## 26 0.000 0.000   
## 27 0.002 0.000   
## 28 0.000 0.000   
## 29 0.000 0.000   
## 30 0.000 0.000   
## 31 0.002 0.000   
## 32 0.000 0.000   
## 33 0.000 0.000   
## 34 0.000 0.000   
## 35 0.000 0.000   
## 36 0.000 0.000   
## 37 0.000 0.000   
## 38 0.001 0.000   
## 39 0.000 0.000   
## 40 0.000 0.000   
## 41 0.007 0.000   
## 42 0.001 0.000   
## 43 0.000 0.000   
## 44 0.000 0.000   
## 45 0.000 0.000   
## 46 0.000 0.000   
## 47 0.005 0.000   
## 48 0.000 0.000   
## 49 0.001 0.000   
## 50 0.005 0.000   
## 51 0.008 0.000   
## 52 0.002 0.000   
## 53 0.001 0.000   
## 54 0.003 0.000   
## 55 0.001 0.000   
## 56 0.000 0.000   
## 57 0.043 0.000   
## 58 0.397 0.000   
## 59 0.004 0.000   
## 60 0.077 0.004   
## 61 0.001 0.000   
## 62 0.051 0.003   
## 63 0.002 0.015   
## 64 0.006 0.000   
## 65 0.000 0.000   
## 66 0.008 0.001   
## 67 0.000 0.002   
## 68 0.002 0.001   
## 69 0.001 0.020   
## 70 0.001 0.073   
## 71 0.000 0.003   
## 72 0.000 0.069   
## 73 0.002 0.160   
## 74 0.002 0.007   
## 75 0.003 0.011   
## 76 0.003 0.019   
## 77 0.001 0.010   
## 78 0.000 0.001   
## 79 0.001 0.007   
## 80 0.005 0.025   
## 81 0.112 0.528   
## 82 0.001 0.004   
## 83 0.004 0.020   
## 84 0.000 0.000   
## age\_class(59,95]:jobtechnician age\_class(30,59]:jobunemployed  
## 1 0.000 0.000   
## 2 0.001 0.000   
## 3 0.000 0.000   
## 4 0.000 0.000   
## 5 0.000 0.000   
## 6 0.000 0.003   
## 7 0.000 0.007   
## 8 0.000 0.017   
## 9 0.000 0.001   
## 10 0.000 0.002   
## 11 0.000 0.001   
## 12 0.006 0.000   
## 13 0.002 0.000   
## 14 0.000 0.000   
## 15 0.001 0.000   
## 16 0.000 0.000   
## 17 0.001 0.000   
## 18 0.000 0.001   
## 19 0.002 0.000   
## 20 0.007 0.000   
## 21 0.001 0.000   
## 22 0.000 0.000   
## 23 0.006 0.000   
## 24 0.000 0.000   
## 25 0.022 0.000   
## 26 0.020 0.000   
## 27 0.000 0.000   
## 28 0.071 0.000   
## 29 0.008 0.000   
## 30 0.001 0.000   
## 31 0.004 0.000   
## 32 0.000 0.000   
## 33 0.098 0.000   
## 34 0.004 0.000   
## 35 0.005 0.000   
## 36 0.005 0.000   
## 37 0.002 0.000   
## 38 0.014 0.000   
## 39 0.000 0.000   
## 40 0.010 0.000   
## 41 0.004 0.000   
## 42 0.006 0.000   
## 43 0.000 0.000   
## 44 0.002 0.000   
## 45 0.045 0.000   
## 46 0.001 0.000   
## 47 0.012 0.000   
## 48 0.007 0.000   
## 49 0.000 0.000   
## 50 0.022 0.000   
## 51 0.000 0.000   
## 52 0.016 0.000   
## 53 0.000 0.000   
## 54 0.000 0.000   
## 55 0.001 0.000   
## 56 0.000 0.000   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.000 0.001   
## 61 0.000 0.000   
## 62 0.001 0.001   
## 63 0.001 0.006   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.002 0.000   
## 67 0.044 0.001   
## 68 0.012 0.003   
## 69 0.002 0.011   
## 70 0.010 0.054   
## 71 0.000 0.672   
## 72 0.006 0.001   
## 73 0.014 0.009   
## 74 0.000 0.000   
## 75 0.001 0.001   
## 76 0.002 0.003   
## 77 0.012 0.003   
## 78 0.000 0.000   
## 79 0.414 0.002   
## 80 0.009 0.008   
## 81 0.053 0.179   
## 82 0.005 0.001   
## 83 0.003 0.007   
## 84 0.000 0.000   
## age\_class(59,95]:jobunemployed age\_class(30,59]:jobunknown  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.000   
## 4 0.000 0.001   
## 5 0.000 0.009   
## 6 0.000 0.002   
## 7 0.001 0.001   
## 8 0.002 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.000 0.000   
## 12 0.002 0.000   
## 13 0.001 0.000   
## 14 0.000 0.000   
## 15 0.001 0.000   
## 16 0.000 0.000   
## 17 0.002 0.000   
## 18 0.000 0.000   
## 19 0.000 0.000   
## 20 0.015 0.000   
## 21 0.016 0.000   
## 22 0.009 0.000   
## 23 0.003 0.000   
## 24 0.127 0.000   
## 25 0.003 0.000   
## 26 0.005 0.000   
## 27 0.000 0.001   
## 28 0.001 0.001   
## 29 0.036 0.000   
## 30 0.059 0.000   
## 31 0.031 0.001   
## 32 0.001 0.000   
## 33 0.022 0.000   
## 34 0.004 0.000   
## 35 0.050 0.000   
## 36 0.001 0.001   
## 37 0.000 0.000   
## 38 0.005 0.000   
## 39 0.000 0.000   
## 40 0.007 0.000   
## 41 0.015 0.000   
## 42 0.052 0.000   
## 43 0.065 0.001   
## 44 0.011 0.000   
## 45 0.004 0.000   
## 46 0.068 0.000   
## 47 0.001 0.000   
## 48 0.016 0.000   
## 49 0.010 0.000   
## 50 0.003 0.000   
## 51 0.039 0.000   
## 52 0.021 0.000   
## 53 0.005 0.000   
## 54 0.003 0.000   
## 55 0.001 0.000   
## 56 0.001 0.000   
## 57 0.000 0.000   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.000 0.000   
## 61 0.000 0.000   
## 62 0.000 0.000   
## 63 0.000 0.001   
## 64 0.000 0.000   
## 65 0.000 0.000   
## 66 0.000 0.000   
## 67 0.006 0.000   
## 68 0.004 0.000   
## 69 0.001 0.000   
## 70 0.005 0.000   
## 71 0.068 0.000   
## 72 0.000 0.000   
## 73 0.001 0.000   
## 74 0.000 0.000   
## 75 0.000 0.000   
## 76 0.000 0.000   
## 77 0.004 0.171   
## 78 0.000 0.773   
## 79 0.164 0.004   
## 80 0.002 0.002   
## 81 0.020 0.026   
## 82 0.002 0.000   
## 83 0.001 0.000   
## 84 0.000 0.000   
## age\_class(59,95]:jobunknown housingyes:loanyes  
## 1 0.000 0.000   
## 2 0.000 0.000   
## 3 0.000 0.002   
## 4 0.000 0.001   
## 5 0.006 0.001   
## 6 0.001 0.000   
## 7 0.001 0.001   
## 8 0.000 0.000   
## 9 0.000 0.000   
## 10 0.000 0.000   
## 11 0.000 0.002   
## 12 0.002 0.007   
## 13 0.000 0.031   
## 14 0.000 0.033   
## 15 0.000 0.000   
## 16 0.000 0.013   
## 17 0.000 0.000   
## 18 0.000 0.001   
## 19 0.001 0.000   
## 20 0.002 0.000   
## 21 0.001 0.000   
## 22 0.001 0.001   
## 23 0.001 0.002   
## 24 0.001 0.000   
## 25 0.006 0.000   
## 26 0.000 0.002   
## 27 0.050 0.000   
## 28 0.019 0.000   
## 29 0.000 0.000   
## 30 0.000 0.000   
## 31 0.022 0.000   
## 32 0.003 0.000   
## 33 0.001 0.000   
## 34 0.012 0.000   
## 35 0.013 0.000   
## 36 0.032 0.000   
## 37 0.001 0.000   
## 38 0.003 0.000   
## 39 0.001 0.001   
## 40 0.008 0.000   
## 41 0.002 0.000   
## 42 0.001 0.000   
## 43 0.036 0.000   
## 44 0.008 0.000   
## 45 0.000 0.000   
## 46 0.001 0.001   
## 47 0.015 0.000   
## 48 0.000 0.003   
## 49 0.001 0.002   
## 50 0.002 0.003   
## 51 0.004 0.000   
## 52 0.002 0.000   
## 53 0.001 0.000   
## 54 0.003 0.000   
## 55 0.002 0.004   
## 56 0.000 0.000   
## 57 0.000 0.001   
## 58 0.000 0.000   
## 59 0.000 0.000   
## 60 0.000 0.001   
## 61 0.000 0.079   
## 62 0.000 0.024   
## 63 0.000 0.000   
## 64 0.000 0.002   
## 65 0.000 0.281   
## 66 0.000 0.376   
## 67 0.008 0.048   
## 68 0.003 0.047   
## 69 0.000 0.000   
## 70 0.000 0.000   
## 71 0.000 0.000   
## 72 0.000 0.000   
## 73 0.000 0.000   
## 74 0.000 0.012   
## 75 0.000 0.007   
## 76 0.000 0.000   
## 77 0.086 0.000   
## 78 0.479 0.000   
## 79 0.134 0.000   
## 80 0.004 0.002   
## 81 0.018 0.000   
## 82 0.001 0.000   
## 83 0.000 0.000   
## 84 0.000 0.000

str(cd\_glm3)

## List of 2  
## $ condindx: num [1:84, 1] 1 1.79 2.18 2.3 2.35 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:84] "1" "2" "3" "4" ...  
## .. ..$ : chr "cond.index"  
## $ pi : num [1:84, 1:84] 7.31e-29 2.74e-31 4.18e-30 3.39e-33 1.55e-31 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : NULL  
## .. ..$ : chr [1:84] "intercept" "(Intercept)" "age\_class(30,59]" "age\_class(59,95]" ...  
## - attr(\*, "class")= chr "colldiag"

cd\_glm3$condindx

## cond.index  
## 1 1.000000e+00  
## 2 1.789515e+00  
## 3 2.181894e+00  
## 4 2.296165e+00  
## 5 2.354608e+00  
## 6 2.363883e+00  
## 7 2.377211e+00  
## 8 2.381974e+00  
## 9 2.391799e+00  
## 10 2.400481e+00  
## 11 2.468612e+00  
## 12 2.617628e+00  
## 13 2.682576e+00  
## 14 2.732472e+00  
## 15 2.870568e+00  
## 16 2.889884e+00  
## 17 2.957410e+00  
## 18 3.002480e+00  
## 19 3.056936e+00  
## 20 3.107204e+00  
## 21 3.134529e+00  
## 22 3.171475e+00  
## 23 3.204103e+00  
## 24 3.222173e+00  
## 25 3.246461e+00  
## 26 3.256548e+00  
## 27 3.267813e+00  
## 28 3.277267e+00  
## 29 3.282390e+00  
## 30 3.294698e+00  
## 31 3.299548e+00  
## 32 3.303846e+00  
## 33 3.306235e+00  
## 34 3.322885e+00  
## 35 3.329004e+00  
## 36 3.334080e+00  
## 37 3.341807e+00  
## 38 3.349562e+00  
## 39 3.352161e+00  
## 40 3.357231e+00  
## 41 3.369581e+00  
## 42 3.371978e+00  
## 43 3.377670e+00  
## 44 3.387251e+00  
## 45 3.390649e+00  
## 46 3.417144e+00  
## 47 3.418099e+00  
## 48 3.462188e+00  
## 49 3.492990e+00  
## 50 3.528046e+00  
## 51 3.560756e+00  
## 52 3.605813e+00  
## 53 3.629544e+00  
## 54 3.682083e+00  
## 55 3.774488e+00  
## 56 3.844529e+00  
## 57 3.936360e+00  
## 58 4.155323e+00  
## 59 4.421838e+00  
## 60 5.382526e+00  
## 61 5.940469e+00  
## 62 6.141414e+00  
## 63 7.088394e+00  
## 64 7.647210e+00  
## 65 8.060835e+00  
## 66 8.774655e+00  
## 67 9.306574e+00  
## 68 9.860487e+00  
## 69 1.064772e+01  
## 70 1.151717e+01  
## 71 1.155937e+01  
## 72 1.216698e+01  
## 73 1.251642e+01  
## 74 1.335764e+01  
## 75 1.408691e+01  
## 76 1.475581e+01  
## 77 1.962279e+01  
## 78 2.000784e+01  
## 79 2.156144e+01  
## 80 2.577979e+01  
## 81 3.701020e+01  
## 82 8.711488e+01  
## 83 9.881955e+01  
## 84 4.900691e+13

cd\_glm3$pi[84,]

## intercept (Intercept)   
## 1.000000e+00 1.000000e+00   
## age\_class(30,59] age\_class(59,95]   
## 1.039113e-06 1.724463e-08   
## jobblue-collar jobentrepreneur   
## 3.943160e-08 2.310550e-07   
## jobhousemaid jobmanagement   
## 8.507849e-07 2.356277e-07   
## jobretired jobself-employed   
## 7.055445e-08 1.939683e-08   
## jobservices jobstudent   
## 7.022890e-08 1.832415e-06   
## jobtechnician jobunemployed   
## 1.649856e-06 2.677416e-07   
## jobunknown maritalmarried   
## 3.604179e-08 1.114333e-06   
## maritalsingle balance\_class0   
## 1.961923e-06 8.230303e-08   
## balance\_class(0, 22] balance\_class(22,131]   
## 4.485925e-07 7.371929e-08   
## balance\_class(131,272] balance\_class(272,448]   
## 1.491661e-06 7.440619e-08   
## balance\_class(448,701] balance\_class(701,1126]   
## 2.037225e-04 2.647905e-06   
## balance\_class(1126,1859] balance\_class(1859,3574]   
## 3.547441e-06 1.457376e-06   
## balance\_class(3574,102127] housingyes   
## 6.407064e-07 3.061858e-05   
## loanyes contacttelephone   
## 9.270420e-07 1.292008e-06   
## contactunknown day   
## 2.733970e-05 2.169756e-06   
## monthfeb monthmar   
## 1.494454e-07 2.423893e-07   
## monthapr monthmay   
## 1.155032e-06 2.053865e-06   
## monthjun monthjul   
## 3.942451e-06 1.363788e-07   
## monthaug monthsep   
## 1.240670e-07 2.476967e-06   
## monthoct monthnov   
## 6.372919e-09 8.870239e-08   
## monthdec campaign\_class2   
## 1.906091e-07 4.446128e-05   
## campaign\_class3 campaign\_class4   
## 9.772642e-07 2.702220e-06   
## campaign\_class5 campaign\_classup 5   
## 1.058966e-06 1.862475e-07   
## pdays\_class(0,91] pdays\_class(91,108]   
## 6.258428e-07 5.458471e-07   
## pdays\_class(108,159] pdays\_class(159,181]   
## 1.558051e-06 4.127223e-06   
## pdays\_class (181,194] pdays\_class(194,258]   
## 3.202125e-06 8.741196e-07   
## pdays\_class(258,300] pdays\_class(300,343]   
## 1.393608e-07 2.772451e-07   
## pdays\_class(343,362] pdays\_class(362,871]   
## 3.046105e-06 2.284353e-07   
## age\_class(30,59]:maritalmarried age\_class(59,95]:maritalmarried   
## 3.296207e-07 4.740537e-07   
## age\_class(30,59]:maritalsingle age\_class(59,95]:maritalsingle   
## 2.075522e-06 4.976069e-07   
## age\_class(30,59]:jobblue-collar age\_class(59,95]:jobblue-collar   
## 1.238150e-05 1.625063e-08   
## age\_class(30,59]:jobentrepreneur age\_class(59,95]:jobentrepreneur   
## 1.198770e-07 2.755437e-07   
## age\_class(30,59]:jobhousemaid age\_class(59,95]:jobhousemaid   
## 7.161805e-10 2.926827e-09   
## age\_class(30,59]:jobmanagement age\_class(59,95]:jobmanagement   
## 4.704003e-07 3.372208e-08   
## age\_class(30,59]:jobretired age\_class(59,95]:jobretired   
## 1.039601e-07 1.146078e-07   
## age\_class(30,59]:jobself-employed age\_class(59,95]:jobself-employed   
## 2.890912e-08 6.329360e-08   
## age\_class(30,59]:jobservices age\_class(59,95]:jobservices   
## 2.740443e-07 1.253590e-08   
## age\_class(30,59]:jobstudent age\_class(30,59]:jobtechnician   
## 4.651868e-07 1.102115e-09   
## age\_class(59,95]:jobtechnician age\_class(30,59]:jobunemployed   
## 2.068477e-08 3.764847e-07   
## age\_class(59,95]:jobunemployed age\_class(30,59]:jobunknown   
## 1.082157e-08 1.278418e-08   
## age\_class(59,95]:jobunknown housingyes:loanyes   
## 2.320721e-08 9.550273e-05

Pdays e previous\_class danno un problema di multicollinearità. Tolgo previous class eil problema non c'è più.

## Bontà adattamento modello

hl\_glm3 <- hoslem.test(training\_set\_vsa$y =="yes", fitted(glm3\_vsa), g=10)  
hl\_glm3

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: training\_set\_vsa$y == "yes", fitted(glm3\_vsa)  
## X-squared = 77.4549, df = 8, p-value = 1.588e-13

Il test, se rigettato, rigetta l'ipotesi di buon adattamento. Qui non si può rigettare. è vero che c'è il problema che il numero di gruppi non dovrebbe essere inferiore a p+1 (<http://thestatsgeek.com/2014/02/16/the-hosmer-lemeshow-goodness-of-fit-test-for-logistic-regression/>), e io con le variabili dummy ho molte p. Ma anche se i gruppi sono 100, il p.value è significativo, quindi il buon adattamento va rigettato.

# Stima performance predittive

glm3\_vsa\_predictions <- predict(glm3\_vsa, validation\_set\_vsa, type="response")

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

AUC\_glm3\_vsa <- roc(validation\_set\_vsa$y, glm3\_vsa\_predictions, levels=c("no", "yes"))  
AUC\_glm3\_vsa$auc

## Area under the curve: 0.755

AUC del modello sul validation set è 0.755 (stabile anche dopo aver tolo previous per multicollinearità). Con quella competerà con gli altri modelli, se la diagnostica non me lo fa cambiare.

Vediamo la AUCPR

AUCPR\_glm3\_vsa <- pr.curve(glm3\_vsa\_predictions, weights.class0 = validation\_set\_vsa$y == "yes", curve=T)  
AUCPR\_glm3\_vsa

##   
## Precision-recall curve  
##   
## Area under curve (Integral):  
## 0.3444406   
##   
## Area under curve (Davis & Goadrich):  
## 0.3444216   
##   
## Curve for scores from 0.006400401 to 0.8996506   
## ( can be plotted with plot(x) )

plot(AUCPR\_glm3\_vsa)

Modello simile a glm3, ma uso i p.value per escludere variabili se possibile

formula\_glm4\_1 <- y ~ age\_class + job + marital + balance\_class + housing + loan + contact + day + month + campaign\_class + pdays\_class + age\_class\*marital + age\_class\*job + loan\*housing

glm4\_1\_vsa <- glm(formula\_glm4\_1, family = "binomial", data = training\_set\_vsa)  
  
summary(glm4\_1\_vsa) #AIC molto basso, è normale la sovrastima delle performance, che non vuol dire overfitting.

##   
## Call:  
## glm(formula = formula\_glm4\_1, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1687 -0.4810 -0.3656 -0.2391 3.0072   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.500080 0.451897 -5.532 3.16e-08  
## age\_class(30,59] 0.092068 0.428430 0.215 0.829848  
## age\_class(59,95] 0.526675 0.590778 0.891 0.372664  
## jobblue-collar 0.006063 0.178650 0.034 0.972926  
## jobentrepreneur 0.062842 0.387935 0.162 0.871312  
## jobhousemaid 0.106918 0.481816 0.222 0.824387  
## jobmanagement 0.122353 0.164247 0.745 0.456314  
## jobretired -10.231499 93.844429 -0.109 0.913182  
## jobself-employed 0.129211 0.268424 0.481 0.630253  
## jobservices 0.070667 0.198524 0.356 0.721871  
## jobstudent 0.164979 0.172464 0.957 0.338769  
## jobtechnician 0.057886 0.173687 0.333 0.738924  
## jobunemployed 0.056318 0.273420 0.206 0.836809  
## jobunknown 0.268962 0.886603 0.303 0.761614  
## maritalmarried -0.029380 0.414953 -0.071 0.943555  
## maritalsingle 0.369989 0.406510 0.910 0.362739  
## balance\_class0 0.009133 0.127519 0.072 0.942905  
## balance\_class(0, 22] -0.090903 0.157046 -0.579 0.562703  
## balance\_class(22,131] 0.075669 0.120813 0.626 0.531095  
## balance\_class(131,272] 0.274397 0.117516 2.335 0.019544  
## balance\_class(272,448] 0.354354 0.116856 3.032 0.002426  
## balance\_class(448,701] 0.280988 0.117285 2.396 0.016585  
## balance\_class(701,1126] 0.383203 0.116197 3.298 0.000974  
## balance\_class(1126,1859] 0.377881 0.115773 3.264 0.001099  
## balance\_class(1859,3574] 0.593122 0.113791 5.212 1.86e-07  
## balance\_class(3574,102127] 0.564231 0.114875 4.912 9.03e-07  
## housingyes -0.586780 0.052034 -11.277 < 2e-16  
## loanyes -0.550209 0.101243 -5.435 5.49e-08  
## contacttelephone -0.308210 0.083170 -3.706 0.000211  
## contactunknown -1.283894 0.083019 -15.465 < 2e-16  
## day 0.002347 0.002804 0.837 0.402590  
## monthfeb 0.718895 0.147235 4.883 1.05e-06  
## monthmar 2.143796 0.179165 11.965 < 2e-16  
## monthapr 1.150920 0.137452 8.373 < 2e-16  
## monthmay 0.584910 0.136655 4.280 1.87e-05  
## monthjun 1.231421 0.151019 8.154 3.52e-16  
## monthjul 0.514489 0.134080 3.837 0.000124  
## monthaug 0.348577 0.136463 2.554 0.010638  
## monthsep 1.746246 0.171984 10.154 < 2e-16  
## monthoct 1.622463 0.158571 10.232 < 2e-16  
## monthnov 0.208321 0.140448 1.483 0.138007  
## monthdec 1.939394 0.218544 8.874 < 2e-16  
## campaign\_class2 -0.218022 0.050740 -4.297 1.73e-05  
## campaign\_class3 -0.041808 0.066185 -0.632 0.527590  
## campaign\_class4 -0.355367 0.088196 -4.029 5.59e-05  
## campaign\_class5 -0.516713 0.123476 -4.185 2.85e-05  
## campaign\_classup 5 -0.729599 0.095520 -7.638 2.20e-14  
## pdays\_class(0,91] 1.045691 0.101408 10.312 < 2e-16  
## pdays\_class(91,108] 1.328945 0.105291 12.622 < 2e-16  
## pdays\_class(108,159] -0.003548 0.137247 -0.026 0.979379  
## pdays\_class(159,181] 0.803065 0.121545 6.607 3.92e-11  
## pdays\_class (181,194] 1.070988 0.111692 9.589 < 2e-16  
## pdays\_class(194,258] -0.084672 0.150489 -0.563 0.573674  
## pdays\_class(258,300] 0.219745 0.137347 1.600 0.109615  
## pdays\_class(300,343] -0.344559 0.158714 -2.171 0.029936  
## pdays\_class(343,362] -0.121708 0.174408 -0.698 0.485281  
## pdays\_class(362,871] 0.859966 0.117542 7.316 2.55e-13  
## age\_class(30,59]:maritalmarried -0.187301 0.421075 -0.445 0.656453  
## age\_class(59,95]:maritalmarried -0.423333 0.457321 -0.926 0.354611  
## age\_class(30,59]:maritalsingle -0.435756 0.414457 -1.051 0.293079  
## age\_class(59,95]:maritalsingle -1.008681 0.647257 -1.558 0.119140  
## age\_class(30,59]:jobblue-collar -0.193497 0.200684 -0.964 0.334952  
## age\_class(59,95]:jobblue-collar 0.227028 0.580469 0.391 0.695715  
## age\_class(30,59]:jobentrepreneur -0.229958 0.417375 -0.551 0.581659  
## age\_class(59,95]:jobentrepreneur 0.800050 0.813006 0.984 0.325084  
## age\_class(30,59]:jobhousemaid -0.491352 0.509042 -0.965 0.334421  
## age\_class(59,95]:jobhousemaid -0.055588 0.702190 -0.079 0.936902  
## age\_class(30,59]:jobmanagement -0.074559 0.184360 -0.404 0.685905  
## age\_class(59,95]:jobmanagement 0.869773 0.473654 1.836 0.066312  
## age\_class(30,59]:jobretired 9.874420 93.844571 0.105 0.916200  
## age\_class(59,95]:jobretired 10.767000 93.845290 0.115 0.908658  
## age\_class(30,59]:jobself-employed -0.397719 0.308222 -1.290 0.196924  
## age\_class(59,95]:jobself-employed 1.199187 0.654842 1.831 0.067061  
## age\_class(30,59]:jobservices -0.066928 0.225592 -0.297 0.766711  
## age\_class(59,95]:jobservices -0.022954 0.825470 -0.028 0.977816  
## age\_class(30,59]:jobstudent -0.057097 0.338063 -0.169 0.865879  
## age\_class(59,95]:jobstudent NA NA NA NA  
## age\_class(30,59]:jobtechnician -0.037721 0.195406 -0.193 0.846929  
## age\_class(59,95]:jobtechnician 0.439834 0.555580 0.792 0.428555  
## age\_class(30,59]:jobunemployed 0.165930 0.304908 0.544 0.586305  
## age\_class(59,95]:jobunemployed 0.292111 0.868208 0.336 0.736530  
## age\_class(30,59]:jobunknown -0.459836 0.934210 -0.492 0.622565  
## age\_class(59,95]:jobunknown -1.052628 1.249830 -0.842 0.399667  
## housingyes:loanyes 0.304603 0.136331 2.234 0.025464  
##   
## (Intercept) \*\*\*  
## age\_class(30,59]   
## age\_class(59,95]   
## jobblue-collar   
## jobentrepreneur   
## jobhousemaid   
## jobmanagement   
## jobretired   
## jobself-employed   
## jobservices   
## jobstudent   
## jobtechnician   
## jobunemployed   
## jobunknown   
## maritalmarried   
## maritalsingle   
## balance\_class0   
## balance\_class(0, 22]   
## balance\_class(22,131]   
## balance\_class(131,272] \*   
## balance\_class(272,448] \*\*   
## balance\_class(448,701] \*   
## balance\_class(701,1126] \*\*\*  
## balance\_class(1126,1859] \*\*   
## balance\_class(1859,3574] \*\*\*  
## balance\_class(3574,102127] \*\*\*  
## housingyes \*\*\*  
## loanyes \*\*\*  
## contacttelephone \*\*\*  
## contactunknown \*\*\*  
## day   
## monthfeb \*\*\*  
## monthmar \*\*\*  
## monthapr \*\*\*  
## monthmay \*\*\*  
## monthjun \*\*\*  
## monthjul \*\*\*  
## monthaug \*   
## monthsep \*\*\*  
## monthoct \*\*\*  
## monthnov   
## monthdec \*\*\*  
## campaign\_class2 \*\*\*  
## campaign\_class3   
## campaign\_class4 \*\*\*  
## campaign\_class5 \*\*\*  
## campaign\_classup 5 \*\*\*  
## pdays\_class(0,91] \*\*\*  
## pdays\_class(91,108] \*\*\*  
## pdays\_class(108,159]   
## pdays\_class(159,181] \*\*\*  
## pdays\_class (181,194] \*\*\*  
## pdays\_class(194,258]   
## pdays\_class(258,300]   
## pdays\_class(300,343] \*   
## pdays\_class(343,362]   
## pdays\_class(362,871] \*\*\*  
## age\_class(30,59]:maritalmarried   
## age\_class(59,95]:maritalmarried   
## age\_class(30,59]:maritalsingle   
## age\_class(59,95]:maritalsingle   
## age\_class(30,59]:jobblue-collar   
## age\_class(59,95]:jobblue-collar   
## age\_class(30,59]:jobentrepreneur   
## age\_class(59,95]:jobentrepreneur   
## age\_class(30,59]:jobhousemaid   
## age\_class(59,95]:jobhousemaid   
## age\_class(30,59]:jobmanagement   
## age\_class(59,95]:jobmanagement .   
## age\_class(30,59]:jobretired   
## age\_class(59,95]:jobretired   
## age\_class(30,59]:jobself-employed   
## age\_class(59,95]:jobself-employed .   
## age\_class(30,59]:jobservices   
## age\_class(59,95]:jobservices   
## age\_class(30,59]:jobstudent   
## age\_class(59,95]:jobstudent   
## age\_class(30,59]:jobtechnician   
## age\_class(59,95]:jobtechnician   
## age\_class(30,59]:jobunemployed   
## age\_class(59,95]:jobunemployed   
## age\_class(30,59]:jobunknown   
## age\_class(59,95]:jobunknown   
## housingyes:loanyes \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 16431 on 27044 degrees of freedom  
## AIC: 16597  
##   
## Number of Fisher Scoring iterations: 10

Tolgo interazion age*marital age*job e job: Raiggiungo previous\_class, magari ora non ci sarà collinearità.

formula\_glm4\_2 <- y ~ age\_class + marital + balance\_class + housing + loan + contact + day + month + campaign\_class + pdays\_class + loan\*housing

glm4\_2\_vsa <- glm(formula\_glm4\_2, family = "binomial", data = training\_set\_vsa)  
summary(glm4\_2\_vsa)

##   
## Call:  
## glm(formula = formula\_glm4\_2, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1069 -0.4821 -0.3667 -0.2357 2.9850   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.064553 0.187554 -11.008 < 2e-16 \*\*\*  
## age\_class(30,59] -0.364248 0.058598 -6.216 5.10e-10 \*\*\*  
## age\_class(59,95] 0.452160 0.096418 4.690 2.74e-06 \*\*\*  
## maritalmarried -0.261261 0.066047 -3.956 7.63e-05 \*\*\*  
## maritalsingle -0.019226 0.073981 -0.260 0.794960   
## balance\_class0 0.024507 0.127112 0.193 0.847116   
## balance\_class(0, 22] -0.082684 0.156168 -0.529 0.596488   
## balance\_class(22,131] 0.079639 0.120489 0.661 0.508636   
## balance\_class(131,272] 0.274901 0.117272 2.344 0.019071 \*   
## balance\_class(272,448] 0.353183 0.116679 3.027 0.002470 \*\*   
## balance\_class(448,701] 0.285922 0.116991 2.444 0.014527 \*   
## balance\_class(701,1126] 0.393068 0.115898 3.391 0.000695 \*\*\*  
## balance\_class(1126,1859] 0.386414 0.115418 3.348 0.000814 \*\*\*  
## balance\_class(1859,3574] 0.599467 0.113478 5.283 1.27e-07 \*\*\*  
## balance\_class(3574,102127] 0.577903 0.114434 5.050 4.42e-07 \*\*\*  
## housingyes -0.596090 0.051150 -11.654 < 2e-16 \*\*\*  
## loanyes -0.567897 0.100656 -5.642 1.68e-08 \*\*\*  
## contacttelephone -0.329670 0.082469 -3.998 6.40e-05 \*\*\*  
## contactunknown -1.308481 0.082936 -15.777 < 2e-16 \*\*\*  
## day 0.002414 0.002800 0.862 0.388604   
## monthfeb 0.731505 0.146933 4.979 6.41e-07 \*\*\*  
## monthmar 2.159120 0.178188 12.117 < 2e-16 \*\*\*  
## monthapr 1.139525 0.136973 8.319 < 2e-16 \*\*\*  
## monthmay 0.564220 0.136208 4.142 3.44e-05 \*\*\*  
## monthjun 1.213675 0.150631 8.057 7.80e-16 \*\*\*  
## monthjul 0.492884 0.133720 3.686 0.000228 \*\*\*  
## monthaug 0.346792 0.135470 2.560 0.010470 \*   
## monthsep 1.746937 0.171141 10.208 < 2e-16 \*\*\*  
## monthoct 1.616201 0.157818 10.241 < 2e-16 \*\*\*  
## monthnov 0.199532 0.140061 1.425 0.154269   
## monthdec 1.958748 0.217962 8.987 < 2e-16 \*\*\*  
## campaign\_class2 -0.213741 0.050548 -4.228 2.35e-05 \*\*\*  
## campaign\_class3 -0.037752 0.065945 -0.572 0.567002   
## campaign\_class4 -0.358964 0.087989 -4.080 4.51e-05 \*\*\*  
## campaign\_class5 -0.521675 0.123057 -4.239 2.24e-05 \*\*\*  
## campaign\_classup 5 -0.727092 0.095262 -7.633 2.30e-14 \*\*\*  
## pdays\_class(0,91] 1.056386 0.100973 10.462 < 2e-16 \*\*\*  
## pdays\_class(91,108] 1.348838 0.104731 12.879 < 2e-16 \*\*\*  
## pdays\_class(108,159] -0.007670 0.136960 -0.056 0.955342   
## pdays\_class(159,181] 0.809213 0.121025 6.686 2.29e-11 \*\*\*  
## pdays\_class (181,194] 1.080760 0.110995 9.737 < 2e-16 \*\*\*  
## pdays\_class(194,258] -0.084246 0.149973 -0.562 0.574294   
## pdays\_class(258,300] 0.219330 0.136753 1.604 0.108750   
## pdays\_class(300,343] -0.363735 0.158435 -2.296 0.021688 \*   
## pdays\_class(343,362] -0.139286 0.173983 -0.801 0.423378   
## pdays\_class(362,871] 0.858413 0.116874 7.345 2.06e-13 \*\*\*  
## housingyes:loanyes 0.326738 0.135878 2.405 0.016188 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 16488 on 27080 degrees of freedom  
## AIC: 16582  
##   
## Number of Fisher Scoring iterations: 6

anova(glm4\_1\_vsa, glm4\_2\_vsa)

## Analysis of Deviance Table  
##   
## Model 1: y ~ age\_class + job + marital + balance\_class + housing + loan +   
## contact + day + month + campaign\_class + pdays\_class + age\_class \*   
## marital + age\_class \* job + loan \* housing  
## Model 2: y ~ age\_class + marital + balance\_class + housing + loan + contact +   
## day + month + campaign\_class + pdays\_class + loan \* housing  
## Resid. Df Resid. Dev Df Deviance  
## 1 27044 16431   
## 2 27080 16488 -36 -56.994

# Multicollinearità

matrix\_glm4\_2 <- model.matrix(glm4\_2\_vsa)  
matrix\_glm4\_2 <- matrix\_glm4\_2[,colSums(matrix\_glm4\_2 != 0) != 0]  
  
cd\_glm4\_2 <- colldiag(matrix\_glm4\_2)  
print(cd\_glm4\_2)

## Condition  
## Index Variance Decomposition Proportions  
## intercept (Intercept) age\_class(30,59]  
## 1 1.000 0.000 0.000 0.001   
## 2 2.193 0.000 0.000 0.000   
## 3 2.338 0.000 0.000 0.000   
## 4 2.466 0.000 0.000 0.000   
## 5 2.545 0.000 0.000 0.000   
## 6 2.588 0.000 0.000 0.000   
## 7 2.664 0.000 0.000 0.000   
## 8 2.683 0.000 0.000 0.001   
## 9 2.756 0.000 0.000 0.000   
## 10 2.798 0.000 0.000 0.000   
## 11 2.820 0.000 0.000 0.000   
## 12 2.861 0.000 0.000 0.000   
## 13 2.888 0.000 0.000 0.000   
## 14 2.895 0.000 0.000 0.000   
## 15 2.898 0.000 0.000 0.000   
## 16 2.905 0.000 0.000 0.000   
## 17 2.917 0.000 0.000 0.000   
## 18 2.921 0.000 0.000 0.000   
## 19 2.929 0.000 0.000 0.000   
## 20 2.933 0.000 0.000 0.000   
## 21 2.944 0.000 0.000 0.000   
## 22 2.946 0.000 0.000 0.000   
## 23 2.956 0.000 0.000 0.000   
## 24 2.963 0.000 0.000 0.000   
## 25 2.970 0.000 0.000 0.000   
## 26 2.980 0.000 0.000 0.000   
## 27 2.983 0.000 0.000 0.000   
## 28 3.029 0.000 0.000 0.000   
## 29 3.048 0.000 0.000 0.000   
## 30 3.068 0.000 0.000 0.001   
## 31 3.103 0.000 0.000 0.000   
## 32 3.148 0.000 0.000 0.000   
## 33 3.213 0.000 0.000 0.000   
## 34 3.282 0.000 0.000 0.002   
## 35 3.379 0.000 0.000 0.000   
## 36 3.408 0.000 0.000 0.003   
## 37 3.502 0.000 0.000 0.001   
## 38 3.891 0.000 0.000 0.000   
## 39 5.151 0.000 0.000 0.009   
## 40 5.218 0.000 0.000 0.000   
## 41 6.835 0.000 0.000 0.007   
## 42 7.036 0.000 0.000 0.018   
## 43 7.811 0.000 0.000 0.097   
## 44 7.904 0.000 0.000 0.452   
## 45 9.955 0.000 0.000 0.233   
## 46 12.911 0.000 0.000 0.138   
## 47 24.422 0.000 0.000 0.032   
## 48 43127898220908.156 1.000 1.000 0.000   
## age\_class(59,95] maritalmarried maritalsingle balance\_class0  
## 1 0.000 0.001 0.001 0.001   
## 2 0.041 0.000 0.000 0.000   
## 3 0.001 0.000 0.001 0.000   
## 4 0.000 0.001 0.005 0.017   
## 5 0.009 0.001 0.006 0.001   
## 6 0.015 0.000 0.001 0.001   
## 7 0.001 0.001 0.003 0.001   
## 8 0.086 0.001 0.004 0.000   
## 9 0.009 0.001 0.007 0.002   
## 10 0.027 0.004 0.016 0.003   
## 11 0.021 0.004 0.018 0.000   
## 12 0.000 0.006 0.024 0.013   
## 13 0.002 0.000 0.000 0.004   
## 14 0.001 0.000 0.002 0.000   
## 15 0.000 0.002 0.007 0.001   
## 16 0.002 0.000 0.000 0.011   
## 17 0.000 0.000 0.000 0.011   
## 18 0.000 0.000 0.002 0.057   
## 19 0.000 0.000 0.000 0.001   
## 20 0.000 0.000 0.000 0.009   
## 21 0.000 0.000 0.002 0.051   
## 22 0.000 0.000 0.002 0.014   
## 23 0.000 0.000 0.000 0.001   
## 24 0.004 0.001 0.002 0.068   
## 25 0.004 0.001 0.004 0.003   
## 26 0.001 0.000 0.001 0.041   
## 27 0.008 0.002 0.011 0.047   
## 28 0.003 0.000 0.000 0.002   
## 29 0.000 0.003 0.008 0.072   
## 30 0.004 0.005 0.015 0.009   
## 31 0.023 0.000 0.002 0.006   
## 32 0.025 0.003 0.013 0.000   
## 33 0.008 0.000 0.000 0.007   
## 34 0.045 0.006 0.020 0.000   
## 35 0.047 0.000 0.001 0.001   
## 36 0.199 0.002 0.005 0.001   
## 37 0.112 0.000 0.002 0.002   
## 38 0.000 0.001 0.000 0.000   
## 39 0.000 0.007 0.004 0.002   
## 40 0.004 0.000 0.001 0.000   
## 41 0.001 0.028 0.013 0.008   
## 42 0.012 0.001 0.002 0.000   
## 43 0.009 0.000 0.007 0.004   
## 44 0.125 0.410 0.146 0.004   
## 45 0.082 0.348 0.419 0.237   
## 46 0.049 0.122 0.167 0.254   
## 47 0.017 0.036 0.055 0.033   
## 48 0.000 0.000 0.000 0.000   
## balance\_class(0, 22] balance\_class(22,131] balance\_class(131,272]  
## 1 0.000 0.001 0.001   
## 2 0.000 0.001 0.001   
## 3 0.001 0.000 0.000   
## 4 0.003 0.001 0.000   
## 5 0.001 0.007 0.004   
## 6 0.000 0.000 0.000   
## 7 0.004 0.008 0.002   
## 8 0.000 0.001 0.000   
## 9 0.001 0.000 0.003   
## 10 0.003 0.022 0.002   
## 11 0.013 0.001 0.000   
## 12 0.001 0.020 0.002   
## 13 0.000 0.005 0.014   
## 14 0.020 0.021 0.002   
## 15 0.099 0.050 0.007   
## 16 0.012 0.007 0.081   
## 17 0.000 0.002 0.018   
## 18 0.103 0.003 0.000   
## 19 0.092 0.011 0.009   
## 20 0.011 0.017 0.010   
## 21 0.010 0.004 0.008   
## 22 0.005 0.027 0.165   
## 23 0.054 0.052 0.013   
## 24 0.043 0.002 0.009   
## 25 0.131 0.005 0.003   
## 26 0.007 0.000 0.004   
## 27 0.004 0.043 0.007   
## 28 0.001 0.002 0.001   
## 29 0.002 0.003 0.001   
## 30 0.001 0.002 0.000   
## 31 0.009 0.054 0.011   
## 32 0.001 0.000 0.001   
## 33 0.010 0.004 0.014   
## 34 0.009 0.017 0.001   
## 35 0.001 0.005 0.002   
## 36 0.001 0.001 0.000   
## 37 0.000 0.000 0.001   
## 38 0.000 0.001 0.002   
## 39 0.001 0.002 0.003   
## 40 0.000 0.001 0.000   
## 41 0.006 0.009 0.007   
## 42 0.000 0.000 0.000   
## 43 0.004 0.010 0.010   
## 44 0.000 0.002 0.002   
## 45 0.129 0.243 0.252   
## 46 0.180 0.287 0.279   
## 47 0.029 0.044 0.048   
## 48 0.000 0.000 0.000   
## balance\_class(272,448] balance\_class(448,701] balance\_class(701,1126]  
## 1 0.001 0.001 0.001   
## 2 0.000 0.000 0.000   
## 3 0.000 0.000 0.001   
## 4 0.000 0.004 0.001   
## 5 0.003 0.003 0.000   
## 6 0.003 0.001 0.000   
## 7 0.000 0.004 0.001   
## 8 0.000 0.000 0.000   
## 9 0.000 0.002 0.000   
## 10 0.004 0.000 0.002   
## 11 0.002 0.003 0.003   
## 12 0.002 0.001 0.003   
## 13 0.017 0.001 0.001   
## 14 0.033 0.005 0.012   
## 15 0.001 0.000 0.000   
## 16 0.051 0.012 0.000   
## 17 0.031 0.004 0.136   
## 18 0.000 0.021 0.020   
## 19 0.083 0.128 0.016   
## 20 0.022 0.065 0.123   
## 21 0.000 0.000 0.021   
## 22 0.001 0.001 0.003   
## 23 0.058 0.013 0.019   
## 24 0.003 0.064 0.000   
## 25 0.017 0.019 0.000   
## 26 0.000 0.001 0.004   
## 27 0.000 0.009 0.008   
## 28 0.003 0.014 0.000   
## 29 0.051 0.015 0.005   
## 30 0.001 0.005 0.012   
## 31 0.005 0.000 0.000   
## 32 0.002 0.001 0.000   
## 33 0.002 0.000 0.000   
## 34 0.000 0.001 0.001   
## 35 0.001 0.001 0.000   
## 36 0.002 0.000 0.001   
## 37 0.000 0.002 0.000   
## 38 0.001 0.001 0.000   
## 39 0.004 0.003 0.003   
## 40 0.000 0.000 0.000   
## 41 0.008 0.007 0.008   
## 42 0.000 0.000 0.000   
## 43 0.012 0.014 0.013   
## 44 0.003 0.003 0.003   
## 45 0.257 0.264 0.266   
## 46 0.268 0.267 0.269   
## 47 0.047 0.043 0.043   
## 48 0.000 0.000 0.000   
## balance\_class(1126,1859] balance\_class(1859,3574]  
## 1 0.001 0.001   
## 2 0.002 0.004   
## 3 0.000 0.001   
## 4 0.001 0.001   
## 5 0.000 0.009   
## 6 0.000 0.002   
## 7 0.001 0.000   
## 8 0.003 0.000   
## 9 0.000 0.000   
## 10 0.001 0.004   
## 11 0.001 0.003   
## 12 0.003 0.002   
## 13 0.057 0.062   
## 14 0.012 0.000   
## 15 0.004 0.008   
## 16 0.047 0.009   
## 17 0.010 0.024   
## 18 0.013 0.003   
## 19 0.000 0.055   
## 20 0.030 0.030   
## 21 0.110 0.001   
## 22 0.010 0.000   
## 23 0.011 0.012   
## 24 0.033 0.010   
## 25 0.002 0.000   
## 26 0.004 0.047   
## 27 0.002 0.000   
## 28 0.000 0.014   
## 29 0.006 0.037   
## 30 0.000 0.002   
## 31 0.008 0.027   
## 32 0.002 0.000   
## 33 0.004 0.009   
## 34 0.007 0.004   
## 35 0.001 0.003   
## 36 0.000 0.003   
## 37 0.001 0.000   
## 38 0.000 0.000   
## 39 0.004 0.002   
## 40 0.000 0.000   
## 41 0.008 0.007   
## 42 0.000 0.000   
## 43 0.010 0.007   
## 44 0.003 0.005   
## 45 0.276 0.278   
## 46 0.272 0.280   
## 47 0.038 0.033   
## 48 0.000 0.000   
## balance\_class(3574,102127] housingyes loanyes contacttelephone  
## 1 0.001 0.002 0.001 0.001   
## 2 0.006 0.005 0.007 0.032   
## 3 0.002 0.000 0.056 0.005   
## 4 0.000 0.006 0.003 0.006   
## 5 0.015 0.000 0.008 0.002   
## 6 0.002 0.000 0.002 0.001   
## 7 0.000 0.001 0.000 0.006   
## 8 0.007 0.000 0.003 0.022   
## 9 0.001 0.002 0.006 0.086   
## 10 0.002 0.000 0.000 0.007   
## 11 0.003 0.000 0.001 0.012   
## 12 0.006 0.001 0.003 0.007   
## 13 0.018 0.000 0.000 0.004   
## 14 0.008 0.000 0.000 0.000   
## 15 0.008 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.001   
## 17 0.047 0.000 0.000 0.000   
## 18 0.008 0.000 0.000 0.000   
## 19 0.012 0.000 0.000 0.001   
## 20 0.025 0.000 0.000 0.000   
## 21 0.017 0.000 0.000 0.005   
## 22 0.005 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.001   
## 24 0.001 0.000 0.000 0.027   
## 25 0.000 0.000 0.000 0.004   
## 26 0.014 0.000 0.000 0.007   
## 27 0.048 0.000 0.000 0.003   
## 28 0.006 0.001 0.001 0.047   
## 29 0.011 0.000 0.000 0.001   
## 30 0.006 0.000 0.000 0.008   
## 31 0.045 0.000 0.001 0.000   
## 32 0.008 0.006 0.004 0.000   
## 33 0.028 0.001 0.000 0.133   
## 34 0.002 0.000 0.001 0.110   
## 35 0.014 0.001 0.000 0.215   
## 36 0.009 0.002 0.001 0.042   
## 37 0.001 0.001 0.000 0.160   
## 38 0.001 0.000 0.000 0.018   
## 39 0.004 0.051 0.030 0.002   
## 40 0.000 0.411 0.138 0.002   
## 41 0.008 0.007 0.014 0.018   
## 42 0.000 0.079 0.226 0.001   
## 43 0.006 0.355 0.446 0.002   
## 44 0.004 0.013 0.007 0.000   
## 45 0.282 0.000 0.001 0.000   
## 46 0.278 0.051 0.032 0.000   
## 47 0.031 0.004 0.008 0.001   
## 48 0.000 0.000 0.000 0.000   
## contactunknown day monthfeb monthmar monthapr monthmay monthjun  
## 1 0.001 0.002 0.000 0.000 0.000 0.000 0.000   
## 2 0.021 0.000 0.004 0.005 0.001 0.004 0.002   
## 3 0.020 0.000 0.001 0.000 0.003 0.001 0.009   
## 4 0.000 0.000 0.004 0.002 0.029 0.002 0.002   
## 5 0.012 0.000 0.006 0.000 0.008 0.000 0.011   
## 6 0.002 0.001 0.055 0.001 0.026 0.000 0.001   
## 7 0.003 0.000 0.011 0.001 0.021 0.006 0.020   
## 8 0.001 0.000 0.014 0.001 0.011 0.000 0.002   
## 9 0.000 0.000 0.000 0.001 0.000 0.001 0.000   
## 10 0.000 0.000 0.004 0.037 0.000 0.001 0.000   
## 11 0.000 0.000 0.004 0.000 0.000 0.001 0.002   
## 12 0.000 0.000 0.001 0.164 0.000 0.000 0.000   
## 13 0.000 0.000 0.002 0.045 0.000 0.000 0.000   
## 14 0.000 0.000 0.002 0.072 0.000 0.000 0.000   
## 15 0.000 0.000 0.002 0.001 0.000 0.000 0.000   
## 16 0.000 0.000 0.000 0.010 0.000 0.000 0.000   
## 17 0.000 0.000 0.000 0.001 0.000 0.000 0.000   
## 18 0.000 0.000 0.000 0.020 0.001 0.000 0.000   
## 19 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## 20 0.000 0.000 0.000 0.004 0.000 0.000 0.000   
## 21 0.000 0.000 0.000 0.004 0.000 0.000 0.000   
## 22 0.000 0.000 0.000 0.003 0.000 0.000 0.000   
## 23 0.000 0.000 0.000 0.001 0.000 0.000 0.000   
## 24 0.000 0.000 0.000 0.003 0.000 0.000 0.000   
## 25 0.000 0.000 0.000 0.000 0.000 0.000 0.001   
## 26 0.000 0.000 0.001 0.210 0.000 0.000 0.001   
## 27 0.000 0.000 0.000 0.000 0.001 0.000 0.000   
## 28 0.001 0.000 0.004 0.010 0.000 0.000 0.001   
## 29 0.000 0.000 0.003 0.000 0.001 0.000 0.000   
## 30 0.001 0.000 0.003 0.064 0.001 0.000 0.000   
## 31 0.003 0.000 0.007 0.000 0.000 0.000 0.000   
## 32 0.001 0.001 0.002 0.028 0.011 0.005 0.017   
## 33 0.000 0.002 0.021 0.000 0.001 0.000 0.010   
## 34 0.005 0.000 0.000 0.007 0.010 0.006 0.015   
## 35 0.000 0.000 0.067 0.001 0.000 0.001 0.006   
## 36 0.000 0.003 0.030 0.010 0.047 0.000 0.000   
## 37 0.002 0.000 0.000 0.003 0.084 0.001 0.002   
## 38 0.000 0.001 0.040 0.016 0.033 0.002 0.006   
## 39 0.018 0.004 0.001 0.000 0.000 0.001 0.001   
## 40 0.023 0.012 0.000 0.001 0.008 0.008 0.008   
## 41 0.766 0.019 0.018 0.004 0.006 0.047 0.058   
## 42 0.032 0.520 0.004 0.004 0.018 0.000 0.003   
## 43 0.022 0.207 0.004 0.000 0.008 0.005 0.007   
## 44 0.019 0.040 0.000 0.000 0.002 0.000 0.001   
## 45 0.000 0.004 0.001 0.000 0.003 0.006 0.002   
## 46 0.029 0.001 0.095 0.040 0.139 0.184 0.151   
## 47 0.014 0.183 0.585 0.225 0.526 0.717 0.659   
## 48 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## monthjul monthaug monthsep monthoct monthnov monthdec campaign\_class2  
## 1 0.000 0.000 0.000 0.000 0.000 0.000 0.002   
## 2 0.000 0.004 0.011 0.014 0.005 0.007 0.001   
## 3 0.006 0.001 0.001 0.000 0.003 0.000 0.000   
## 4 0.012 0.011 0.002 0.002 0.001 0.004 0.004   
## 5 0.003 0.009 0.005 0.011 0.021 0.002 0.000   
## 6 0.000 0.002 0.011 0.002 0.011 0.016 0.000   
## 7 0.000 0.000 0.020 0.011 0.002 0.004 0.000   
## 8 0.002 0.002 0.032 0.040 0.035 0.000 0.004   
## 9 0.007 0.016 0.043 0.001 0.000 0.063 0.014   
## 10 0.002 0.000 0.014 0.000 0.000 0.107 0.005   
## 11 0.010 0.014 0.004 0.002 0.002 0.114 0.001   
## 12 0.000 0.000 0.061 0.025 0.002 0.004 0.001   
## 13 0.001 0.000 0.039 0.059 0.001 0.021 0.018   
## 14 0.000 0.000 0.024 0.043 0.000 0.002 0.000   
## 15 0.000 0.001 0.003 0.004 0.002 0.029 0.018   
## 16 0.001 0.001 0.001 0.001 0.000 0.001 0.003   
## 17 0.000 0.001 0.002 0.011 0.000 0.012 0.016   
## 18 0.000 0.000 0.004 0.031 0.000 0.002 0.028   
## 19 0.000 0.000 0.026 0.011 0.000 0.000 0.000   
## 20 0.000 0.000 0.001 0.006 0.000 0.000 0.001   
## 21 0.001 0.000 0.012 0.015 0.000 0.000 0.000   
## 22 0.001 0.000 0.013 0.017 0.000 0.001 0.029   
## 23 0.000 0.000 0.000 0.000 0.000 0.010 0.003   
## 24 0.001 0.000 0.023 0.019 0.001 0.000 0.008   
## 25 0.000 0.000 0.021 0.017 0.001 0.005 0.024   
## 26 0.000 0.002 0.001 0.006 0.000 0.000 0.000   
## 27 0.004 0.002 0.011 0.052 0.000 0.000 0.001   
## 28 0.007 0.018 0.009 0.001 0.001 0.055 0.016   
## 29 0.013 0.001 0.001 0.020 0.000 0.005 0.099   
## 30 0.000 0.003 0.026 0.003 0.002 0.166 0.004   
## 31 0.005 0.000 0.049 0.007 0.000 0.083 0.012   
## 32 0.000 0.006 0.001 0.006 0.010 0.020 0.033   
## 33 0.009 0.000 0.082 0.057 0.018 0.013 0.000   
## 34 0.005 0.009 0.000 0.029 0.000 0.016 0.008   
## 35 0.014 0.004 0.025 0.024 0.021 0.001 0.000   
## 36 0.000 0.002 0.049 0.029 0.018 0.026 0.010   
## 37 0.004 0.000 0.003 0.002 0.039 0.000 0.000   
## 38 0.006 0.003 0.047 0.045 0.029 0.059 0.000   
## 39 0.003 0.010 0.001 0.003 0.000 0.000 0.537   
## 40 0.004 0.021 0.001 0.001 0.001 0.002 0.081   
## 41 0.024 0.034 0.002 0.002 0.016 0.002 0.002   
## 42 0.010 0.017 0.001 0.013 0.018 0.001 0.003   
## 43 0.027 0.001 0.001 0.000 0.009 0.001 0.000   
## 44 0.004 0.002 0.000 0.000 0.008 0.000 0.000   
## 45 0.005 0.006 0.001 0.001 0.003 0.000 0.000   
## 46 0.138 0.139 0.052 0.065 0.147 0.023 0.008   
## 47 0.669 0.660 0.263 0.290 0.572 0.121 0.001   
## 48 0.000 0.000 0.000 0.000 0.000 0.000 0.000   
## campaign\_class3 campaign\_class4 campaign\_class5 campaign\_classup 5  
## 1 0.001 0.001 0.001 0.001   
## 2 0.000 0.000 0.000 0.000   
## 3 0.001 0.000 0.000 0.000   
## 4 0.000 0.012 0.001 0.053   
## 5 0.001 0.012 0.002 0.014   
## 6 0.001 0.003 0.001 0.000   
## 7 0.001 0.000 0.000 0.006   
## 8 0.002 0.002 0.000 0.000   
## 9 0.028 0.019 0.017 0.004   
## 10 0.034 0.001 0.000 0.000   
## 11 0.019 0.097 0.010 0.003   
## 12 0.001 0.001 0.061 0.004   
## 13 0.004 0.019 0.007 0.000   
## 14 0.001 0.007 0.009 0.029   
## 15 0.091 0.024 0.065 0.011   
## 16 0.011 0.036 0.129 0.026   
## 17 0.017 0.069 0.005 0.006   
## 18 0.007 0.022 0.056 0.001   
## 19 0.008 0.000 0.003 0.009   
## 20 0.002 0.000 0.005 0.015   
## 21 0.035 0.008 0.011 0.029   
## 22 0.029 0.046 0.023 0.023   
## 23 0.073 0.015 0.291 0.001   
## 24 0.002 0.001 0.004 0.022   
## 25 0.026 0.001 0.045 0.089   
## 26 0.000 0.021 0.002 0.015   
## 27 0.000 0.073 0.021 0.065   
## 28 0.170 0.103 0.036 0.002   
## 29 0.005 0.008 0.004 0.065   
## 30 0.023 0.062 0.018 0.002   
## 31 0.000 0.002 0.003 0.019   
## 32 0.013 0.002 0.000 0.046   
## 33 0.001 0.000 0.003 0.001   
## 34 0.015 0.005 0.000 0.007   
## 35 0.003 0.001 0.000 0.005   
## 36 0.005 0.001 0.000 0.007   
## 37 0.000 0.001 0.000 0.001   
## 38 0.000 0.000 0.000 0.002   
## 39 0.315 0.259 0.131 0.310   
## 40 0.042 0.052 0.026 0.063   
## 41 0.001 0.000 0.000 0.000   
## 42 0.003 0.004 0.004 0.019   
## 43 0.001 0.002 0.001 0.007   
## 44 0.000 0.001 0.000 0.002   
## 45 0.000 0.000 0.000 0.000   
## 46 0.005 0.005 0.003 0.005   
## 47 0.000 0.001 0.001 0.008   
## 48 0.000 0.000 0.000 0.000   
## pdays\_class(0,91] pdays\_class(91,108] pdays\_class(108,159]  
## 1 0.000 0.000 0.000   
## 2 0.010 0.019 0.010   
## 3 0.001 0.002 0.007   
## 4 0.002 0.000 0.014   
## 5 0.000 0.015 0.029   
## 6 0.005 0.002 0.052   
## 7 0.001 0.025 0.004   
## 8 0.021 0.004 0.013   
## 9 0.022 0.150 0.000   
## 10 0.027 0.004 0.026   
## 11 0.034 0.005 0.025   
## 12 0.088 0.055 0.000   
## 13 0.017 0.015 0.007   
## 14 0.026 0.026 0.049   
## 15 0.001 0.002 0.018   
## 16 0.002 0.015 0.012   
## 17 0.005 0.000 0.022   
## 18 0.017 0.014 0.043   
## 19 0.002 0.002 0.006   
## 20 0.032 0.000 0.016   
## 21 0.021 0.023 0.077   
## 22 0.058 0.009 0.008   
## 23 0.001 0.004 0.002   
## 24 0.170 0.105 0.000   
## 25 0.027 0.031 0.014   
## 26 0.085 0.022 0.067   
## 27 0.020 0.000 0.004   
## 28 0.007 0.008 0.021   
## 29 0.006 0.000 0.002   
## 30 0.036 0.000 0.017   
## 31 0.025 0.015 0.111   
## 32 0.094 0.072 0.000   
## 33 0.019 0.070 0.000   
## 34 0.013 0.055 0.007   
## 35 0.000 0.048 0.093   
## 36 0.000 0.009 0.033   
## 37 0.002 0.070 0.000   
## 38 0.065 0.086 0.182   
## 39 0.005 0.002 0.000   
## 40 0.001 0.001 0.000   
## 41 0.026 0.011 0.005   
## 42 0.002 0.001 0.001   
## 43 0.000 0.000 0.000   
## 44 0.000 0.000 0.000   
## 45 0.000 0.001 0.000   
## 46 0.001 0.001 0.000   
## 47 0.001 0.001 0.000   
## 48 0.000 0.000 0.000   
## pdays\_class(159,181] pdays\_class (181,194] pdays\_class(194,258]  
## 1 0.000 0.000 0.000   
## 2 0.002 0.021 0.004   
## 3 0.001 0.000 0.007   
## 4 0.004 0.005 0.012   
## 5 0.006 0.024 0.004   
## 6 0.005 0.006 0.114   
## 7 0.013 0.002 0.038   
## 8 0.043 0.001 0.034   
## 9 0.029 0.011 0.001   
## 10 0.101 0.040 0.020   
## 11 0.001 0.088 0.004   
## 12 0.001 0.021 0.004   
## 13 0.009 0.015 0.009   
## 14 0.178 0.003 0.012   
## 15 0.004 0.004 0.017   
## 16 0.011 0.025 0.000   
## 17 0.001 0.012 0.000   
## 18 0.000 0.060 0.035   
## 19 0.010 0.007 0.002   
## 20 0.016 0.003 0.007   
## 21 0.000 0.010 0.001   
## 22 0.000 0.017 0.027   
## 23 0.006 0.001 0.001   
## 24 0.005 0.020 0.000   
## 25 0.029 0.079 0.049   
## 26 0.073 0.051 0.009   
## 27 0.007 0.085 0.034   
## 28 0.028 0.010 0.031   
## 29 0.000 0.000 0.023   
## 30 0.113 0.089 0.011   
## 31 0.063 0.032 0.003   
## 32 0.073 0.001 0.008   
## 33 0.007 0.000 0.095   
## 34 0.057 0.007 0.002   
## 35 0.000 0.002 0.089   
## 36 0.015 0.001 0.078   
## 37 0.007 0.053 0.000   
## 38 0.048 0.182 0.162   
## 39 0.000 0.000 0.000   
## 40 0.003 0.001 0.006   
## 41 0.029 0.005 0.006   
## 42 0.000 0.001 0.003   
## 43 0.000 0.000 0.007   
## 44 0.001 0.000 0.001   
## 45 0.000 0.000 0.000   
## 46 0.000 0.000 0.005   
## 47 0.000 0.001 0.022   
## 48 0.000 0.000 0.000   
## pdays\_class(258,300] pdays\_class(300,343] pdays\_class(343,362]  
## 1 0.000 0.000 0.000   
## 2 0.000 0.000 0.005   
## 3 0.002 0.003 0.000   
## 4 0.019 0.065 0.016   
## 5 0.017 0.031 0.009   
## 6 0.022 0.063 0.000   
## 7 0.006 0.039 0.084   
## 8 0.004 0.056 0.000   
## 9 0.002 0.000 0.013   
## 10 0.095 0.004 0.010   
## 11 0.005 0.008 0.016   
## 12 0.022 0.000 0.002   
## 13 0.068 0.002 0.034   
## 14 0.065 0.025 0.015   
## 15 0.042 0.000 0.078   
## 16 0.007 0.000 0.038   
## 17 0.000 0.014 0.063   
## 18 0.041 0.011 0.001   
## 19 0.004 0.001 0.008   
## 20 0.057 0.000 0.008   
## 21 0.013 0.016 0.023   
## 22 0.069 0.004 0.000   
## 23 0.002 0.007 0.028   
## 24 0.001 0.001 0.000   
## 25 0.009 0.017 0.064   
## 26 0.014 0.002 0.025   
## 27 0.013 0.001 0.049   
## 28 0.044 0.018 0.002   
## 29 0.000 0.019 0.002   
## 30 0.054 0.003 0.010   
## 31 0.029 0.002 0.001   
## 32 0.003 0.026 0.090   
## 33 0.014 0.027 0.005   
## 34 0.021 0.004 0.138   
## 35 0.075 0.001 0.003   
## 36 0.006 0.143 0.006   
## 37 0.033 0.258 0.024   
## 38 0.067 0.074 0.001   
## 39 0.001 0.001 0.003   
## 40 0.010 0.005 0.004   
## 41 0.035 0.041 0.113   
## 42 0.000 0.002 0.000   
## 43 0.001 0.000 0.001   
## 44 0.003 0.000 0.001   
## 45 0.000 0.000 0.000   
## 46 0.000 0.002 0.008   
## 47 0.003 0.000 0.000   
## 48 0.000 0.000 0.000   
## pdays\_class(362,871] housingyes:loanyes  
## 1 0.000 0.001   
## 2 0.000 0.013   
## 3 0.002 0.056   
## 4 0.003 0.000   
## 5 0.006 0.012   
## 6 0.002 0.002   
## 7 0.078 0.000   
## 8 0.011 0.004   
## 9 0.000 0.008   
## 10 0.028 0.001   
## 11 0.001 0.004   
## 12 0.006 0.001   
## 13 0.024 0.000   
## 14 0.014 0.000   
## 15 0.078 0.000   
## 16 0.059 0.000   
## 17 0.014 0.000   
## 18 0.066 0.000   
## 19 0.012 0.000   
## 20 0.025 0.000   
## 21 0.118 0.000   
## 22 0.008 0.000   
## 23 0.002 0.000   
## 24 0.024 0.000   
## 25 0.000 0.001   
## 26 0.001 0.000   
## 27 0.000 0.001   
## 28 0.000 0.002   
## 29 0.071 0.005   
## 30 0.024 0.000   
## 31 0.081 0.000   
## 32 0.043 0.001   
## 33 0.054 0.001   
## 34 0.044 0.002   
## 35 0.007 0.001   
## 36 0.004 0.000   
## 37 0.002 0.001   
## 38 0.012 0.000   
## 39 0.000 0.023   
## 40 0.010 0.075   
## 41 0.062 0.008   
## 42 0.000 0.247   
## 43 0.000 0.505   
## 44 0.002 0.006   
## 45 0.000 0.000   
## 46 0.002 0.014   
## 47 0.000 0.003   
## 48 0.000 0.000

str(cd\_glm4\_2)

## List of 2  
## $ condindx: num [1:48, 1] 1 2.19 2.34 2.47 2.54 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:48] "1" "2" "3" "4" ...  
## .. ..$ : chr "cond.index"  
## $ pi : num [1:48, 1:48] 1.22e-28 1.37e-29 1.26e-29 3.29e-31 8.22e-30 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : NULL  
## .. ..$ : chr [1:48] "intercept" "(Intercept)" "age\_class(30,59]" "age\_class(59,95]" ...  
## - attr(\*, "class")= chr "colldiag"

cd\_glm4\_2$condindx

## cond.index  
## 1 1.000000e+00  
## 2 2.192826e+00  
## 3 2.337719e+00  
## 4 2.466052e+00  
## 5 2.544518e+00  
## 6 2.588319e+00  
## 7 2.663832e+00  
## 8 2.682912e+00  
## 9 2.756137e+00  
## 10 2.797921e+00  
## 11 2.820251e+00  
## 12 2.860842e+00  
## 13 2.887513e+00  
## 14 2.895462e+00  
## 15 2.897889e+00  
## 16 2.905191e+00  
## 17 2.917461e+00  
## 18 2.920550e+00  
## 19 2.929264e+00  
## 20 2.932584e+00  
## 21 2.943936e+00  
## 22 2.946241e+00  
## 23 2.956250e+00  
## 24 2.963420e+00  
## 25 2.969551e+00  
## 26 2.980325e+00  
## 27 2.982784e+00  
## 28 3.028804e+00  
## 29 3.047703e+00  
## 30 3.067600e+00  
## 31 3.103458e+00  
## 32 3.148130e+00  
## 33 3.213176e+00  
## 34 3.281981e+00  
## 35 3.378940e+00  
## 36 3.407770e+00  
## 37 3.501635e+00  
## 38 3.891211e+00  
## 39 5.150839e+00  
## 40 5.218256e+00  
## 41 6.835298e+00  
## 42 7.036074e+00  
## 43 7.811143e+00  
## 44 7.903754e+00  
## 45 9.955215e+00  
## 46 1.291073e+01  
## 47 2.442238e+01  
## 48 4.312790e+13

cd\_glm4\_2$pi[48,]

## intercept (Intercept)   
## 1.000000e+00 1.000000e+00   
## age\_class(30,59] age\_class(59,95]   
## 5.597972e-07 1.023729e-06   
## maritalmarried maritalsingle   
## 9.485725e-07 1.762157e-07   
## balance\_class0 balance\_class(0, 22]   
## 4.038719e-06 2.805880e-06   
## balance\_class(22,131] balance\_class(131,272]   
## 1.246110e-05 7.364212e-06   
## balance\_class(272,448] balance\_class(448,701]   
## 6.024070e-06 2.154730e-04   
## balance\_class(701,1126] balance\_class(1126,1859]   
## 1.839297e-06 4.076299e-06   
## balance\_class(1859,3574] balance\_class(3574,102127]   
## 3.043862e-06 5.471478e-06   
## housingyes loanyes   
## 1.075901e-05 1.291520e-08   
## contacttelephone contactunknown   
## 4.426141e-07 7.606005e-05   
## day monthfeb   
## 8.782769e-06 1.664152e-07   
## monthmar monthapr   
## 3.015666e-07 1.683590e-07   
## monthmay monthjun   
## 3.057065e-06 3.574052e-06   
## monthjul monthaug   
## 2.553713e-06 6.332679e-08   
## monthsep monthoct   
## 8.410457e-08 1.061445e-07   
## monthnov monthdec   
## 9.426110e-07 2.358628e-07   
## campaign\_class2 campaign\_class3   
## 9.851269e-05 3.946733e-07   
## campaign\_class4 campaign\_class5   
## 1.559199e-06 2.363766e-07   
## campaign\_classup 5 pdays\_class(0,91]   
## 4.624924e-08 2.266453e-09   
## pdays\_class(91,108] pdays\_class(108,159]   
## 3.900007e-07 1.326767e-06   
## pdays\_class(159,181] pdays\_class (181,194]   
## 5.004285e-08 4.159230e-07   
## pdays\_class(194,258] pdays\_class(258,300]   
## 2.321058e-06 2.511080e-07   
## pdays\_class(300,343] pdays\_class(343,362]   
## 1.107903e-10 8.647637e-07   
## pdays\_class(362,871] housingyes:loanyes   
## 3.656145e-09 1.324656e-04

Previous continua a dare problemi di multicollinearità. Lo ritolgo all'origine.

## Bontà adattamento modello

hl\_glm4\_2 <- hoslem.test(training\_set\_vsa$y =="yes", fitted(glm4\_2\_vsa), g=10)  
hl\_glm4\_2

##   
## Hosmer and Lemeshow goodness of fit (GOF) test  
##   
## data: training\_set\_vsa$y == "yes", fitted(glm4\_2\_vsa)  
## X-squared = 73.9396, df = 8, p-value = 8.041e-13

Nada.

## Stima delle performance predittive

glm4\_2\_vsa\_predictions <- predict(glm4\_2\_vsa, validation\_set\_vsa, type="response")  
  
AUC\_glm4\_2\_vsa <- roc(validation\_set\_vsa$y, glm4\_2\_vsa\_predictions, levels=c("no", "yes"))  
AUC\_glm4\_2\_vsa$auc

## Area under the curve: 0.7556

plot(AUC\_glm4\_2\_vsa, legacy.axes=TRUE)

##   
## Call:  
## roc.default(response = validation\_set\_vsa$y, predictor = glm4\_2\_vsa\_predictions, levels = c("no", "yes"))  
##   
## Data: glm4\_2\_vsa\_predictions in 8016 controls (validation\_set\_vsa$y no) < 1026 cases (validation\_set\_vsa$y yes).  
## Area under the curve: 0.7556

AUC di 0.7556. Senza togliere le variabili con basso p.value eravamo a 0.755, praticamente performance identiche con stessi problemi di multicollinearità Giusto per scrupolo voglio vedere se gli errori standard sono inferiori nel secondo caso (anche se solo il principio della parsimonia basta a giustificare il secondo modello).

a <- tidy(glm3\_vsa)  
b <- tidy(glm4\_2\_vsa)  
glm\_comp\_3\_and\_4\_2 <- a %>%  
 inner\_join(b, by = "term") %>%  
 select(term, SE\_glm3 = std.error.x, SE\_glm4\_2 = std.error.y) %>%  
 mutate(comp\_SE = SE\_glm4\_2 - SE\_glm3) %>%  
 arrange(comp\_SE)

Quasi tutti inferiori, anche se di poco. Mi domando se un tale misero vantaggio di SE e di AUC mi debba indurre a escludere delle variabili. Preferire glm3 o glm4? Mah. Considera però che glm3 ha il problema del rank e che in glm4 ho valutato i p-value senza la correzione di bonferroni.

AUCPR\_glm4\_2\_vsa <- pr.curve(glm4\_2\_vsa\_predictions, weights.class0 = validation\_set\_vsa$y == "yes", curve=T)  
AUCPR\_glm4\_2\_vsa

##   
## Precision-recall curve  
##   
## Area under curve (Integral):  
## 0.3396937   
##   
## Area under curve (Davis & Goadrich):  
## 0.3396539   
##   
## Curve for scores from 0.007738038 to 0.8641317   
## ( can be plotted with plot(x) )

plot(AUCPR\_glm4\_2\_vsa)

formula\_glm5\_1 <- y ~ poutcome  
glm5\_1\_vsa <- glm(formula\_glm5\_1, family = "binomial", data = training\_set\_vsa)  
summary(glm5\_1\_vsa)

##   
## Call:  
## glm(formula = formula\_glm5\_1, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4414 -0.4384 -0.4384 -0.4384 2.1864   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.91499 0.05495 -34.851 < 2e-16 \*\*\*  
## poutcomeother 0.26313 0.09818 2.680 0.00736 \*\*   
## poutcomesuccess 2.51716 0.08846 28.455 < 2e-16 \*\*\*  
## poutcomeunknown -0.37923 0.05968 -6.354 2.1e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 17998 on 27123 degrees of freedom  
## AIC: 18006  
##   
## Number of Fisher Scoring iterations: 5

Alpha = 0.05 / ? Non si sa. Non ha senso fare la forward. Dovresti assumere un numero di variabili che incliderai, ma magari escludi qualcosa che andrebbe messo.

formula\_glm5\_2 <- y~age + job + marital + education + default + balance + housing + loan + contact + pdays + day + month + campaign + previous  
glm5\_2\_vsa <- glm(formula\_glm5\_2, family = "binomial", data = training\_set\_vsa)  
summary(glm5\_2\_vsa)

##   
## Call:  
## glm(formula = formula\_glm5\_2, family = "binomial", data = training\_set\_vsa)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.4020 -0.5136 -0.3912 -0.2297 3.7919   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.352e+00 2.086e-01 -11.274 < 2e-16 \*\*\*  
## age 5.577e-03 2.441e-03 2.285 0.022317 \*   
## jobblue-collar -1.136e-01 8.327e-02 -1.364 0.172682   
## jobentrepreneur -2.330e-01 1.410e-01 -1.653 0.098429 .   
## jobhousemaid -2.748e-01 1.449e-01 -1.896 0.057927 .   
## jobmanagement -6.623e-02 8.297e-02 -0.798 0.424701   
## jobretired 3.267e-01 1.086e-01 3.007 0.002642 \*\*   
## jobself-employed -1.802e-01 1.272e-01 -1.416 0.156660   
## jobservices 1.062e-02 9.306e-02 0.114 0.909150   
## jobstudent 4.190e-01 1.249e-01 3.355 0.000795 \*\*\*  
## jobtechnician -1.097e-02 7.857e-02 -0.140 0.889002   
## jobunemployed 1.743e-01 1.210e-01 1.441 0.149684   
## jobunknown -2.914e-01 2.691e-01 -1.083 0.278875   
## maritalmarried -1.656e-01 6.500e-02 -2.547 0.010858 \*   
## maritalsingle 9.567e-02 7.466e-02 1.281 0.200045   
## educationsecondary 1.541e-01 7.157e-02 2.153 0.031332 \*   
## educationtertiary 3.999e-01 8.281e-02 4.830 1.37e-06 \*\*\*  
## educationunknown 2.265e-01 1.158e-01 1.956 0.050523 .   
## defaultyes -7.774e-02 1.809e-01 -0.430 0.667423   
## balance 2.150e-05 5.693e-06 3.777 0.000159 \*\*\*  
## housingyes -6.507e-01 4.857e-02 -13.398 < 2e-16 \*\*\*  
## loanyes -4.885e-01 6.819e-02 -7.163 7.90e-13 \*\*\*  
## contacttelephone -2.593e-01 8.098e-02 -3.202 0.001364 \*\*   
## contactunknown -1.422e+00 8.201e-02 -17.337 < 2e-16 \*\*\*  
## pdays 9.488e-04 2.090e-04 4.539 5.66e-06 \*\*\*  
## day 3.789e-03 2.780e-03 1.363 0.172929   
## monthfeb 7.335e-01 1.449e-01 5.063 4.12e-07 \*\*\*  
## monthmar 2.171e+00 1.742e-01 12.460 < 2e-16 \*\*\*  
## monthapr 1.030e+00 1.333e-01 7.724 1.12e-14 \*\*\*  
## monthmay 5.549e-01 1.326e-01 4.184 2.86e-05 \*\*\*  
## monthjun 1.289e+00 1.476e-01 8.734 < 2e-16 \*\*\*  
## monthjul 4.001e-01 1.304e-01 3.067 0.002159 \*\*   
## monthaug 2.464e-01 1.327e-01 1.856 0.063399 .   
## monthsep 2.002e+00 1.660e-01 12.060 < 2e-16 \*\*\*  
## monthoct 1.807e+00 1.532e-01 11.790 < 2e-16 \*\*\*  
## monthnov 2.671e-01 1.360e-01 1.964 0.049475 \*   
## monthdec 1.944e+00 2.121e-01 9.162 < 2e-16 \*\*\*  
## campaign -1.020e-01 1.119e-02 -9.117 < 2e-16 \*\*\*  
## previous 4.183e-02 9.080e-03 4.607 4.09e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 19589 on 27126 degrees of freedom  
## Residual deviance: 16973 on 27088 degrees of freedom  
## AIC: 17051  
##   
## Number of Fisher Scoring iterations: 6

glm5\_2\_vsa\_table <- tidy(glm5\_2\_vsa)  
da\_escludere <- glm5\_2\_vsa\_table %>%  
 filter(p.value > 0.05/nrow(a))

Praticamente tutto. è evidente che il test perde di potenza, oppure che i test statistici per questa analisi tendono a considerare il modello non corretto (vedi anche HL). Bisogna rifarsi perciò alla AUC. La step tramite p.value è abortita.

# The train and test set are loaded into your workspace.  
  
# Set random seed. Don't remove this line  
set.seed(1)  
  
# Load the rpart, rattle, rpart.plot and RColorBrewer package  
library("rpart")  
library("rpart.plot")  
library("RColorBrewer")  
library("rattle")

## Rattle: A free graphical interface for data mining with R.  
## Version 3.4.1 Copyright (c) 2006-2014 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

# Build a tree model: tree  
tree <- rpart(y ~ age + job + marital + balance + housing + loan + contact + day + month + campaign + pdays + previous, data = training\_set\_vsa, method = "class", control=rpart.control(minsplit=5, cp=0.001))  
  
# Draw the decision tree  
pred <- predict(tree, validation\_set\_vsa, type = "class")  
conf <- table(validation\_set\_vsa$y, pred)  
conf

## pred  
## no yes  
## no 7801 215  
## yes 801 225

# Auroc test

test\_set\_vsa$glm3\_vsa\_predictions\_test <- predict(glm3\_vsa, test\_set\_vsa, type="response")

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

AUC\_glm3\_vsa\_test <- roc(test\_set\_vsa$y, test\_set\_vsa$glm3\_vsa\_predictions\_test, levels=c("no", "yes"))  
AUC\_glm3\_vsa\_test$auc

## Area under the curve: 0.766

# Aucpr test

AUCPR\_glm3\_vsa <- pr.curve(test\_set\_vsa$glm3\_vsa\_predictions\_test, weights.class0 = test\_set\_vsa$y == "yes", curve=T)  
AUCPR\_glm3\_vsa

##   
## Precision-recall curve  
##   
## Area under curve (Integral):  
## 0.3650118   
##   
## Area under curve (Davis & Goadrich):  
## 0.3649406   
##   
## Curve for scores from 0.007979179 to 0.8990745   
## ( can be plotted with plot(x) )

# Grafici

plot(AUC\_glm3\_vsa\_test, legacy.axes = TRUE)

##   
## Call:  
## roc.default(response = test\_set\_vsa$y, predictor = test\_set\_vsa$glm3\_vsa\_predictions\_test, levels = c("no", "yes"))  
##   
## Data: test\_set\_vsa$glm3\_vsa\_predictions\_test in 7955 controls (test\_set\_vsa$y no) < 1087 cases (test\_set\_vsa$y yes).  
## Area under the curve: 0.766

plot(AUCPR\_glm3\_vsa)

g\_estim\_prob\_class <- ggplot(data = test\_set\_vsa, aes(x = glm3\_vsa\_predictions\_test, col = y)) +  
 geom\_density()  
g\_estim\_prob\_class

#construisco un dataframe per il plot sensitivuty probability vs cutoff. vedi https://cran.r-project.org/web/packages/ROCR/ROCR.pdf  
  
prediction\_test\_obj <- prediction(predictions = test\_set\_vsa$glm3\_vsa\_predictions\_test, labels = test\_set\_vsa$y)  
tpr\_obj <- performance(prediction\_test\_obj, measure = "tpr")  
fpr\_obj <- performance(prediction\_test\_obj, measure = "spec")  
ppv\_obj <- performance(prediction\_test\_obj, measure = "prec")   
  
sens\_spec\_cutoff\_df\_wide <- data.frame (cutoff = as.numeric( unlist ( tpr\_obj@x.values) ), sensitivity = as.numeric( unlist ( tpr\_obj@y.values) ), specificity = as.numeric( unlist ( fpr\_obj@y.values) ), ppv = as.numeric( unlist (ppv\_obj@y.values) ))  
  
sens\_spec\_cutoff\_df\_tidy <- sens\_spec\_cutoff\_df\_wide %>%  
 gather(key = type\_indicator, value\_indicator, sensitivity:ppv)  
  
g\_sens\_spec\_cutoff <- ggplot(sens\_spec\_cutoff\_df\_tidy, aes(x = cutoff, y = value\_indicator, col = type\_indicator)) +  
 geom\_line() +   
 scale\_x\_continuous(breaks = seq(0.01, 1, 0.02)) +  
 scale\_y\_continuous(breaks = seq(0, 1, 0.05)) +  
 theme(axis.text.x = element\_text(angle = 45, hjust = 1))  
g\_sens\_spec\_cutoff#eccellente!

## Warning in loop\_apply(n, do.ply): Removed 1 rows containing missing values  
## (geom\_path).

Con questo grafico, `g\_sens\_spec\_cutoff``, posso scegliere la soglia di cutoff che desidero e sapere tpr, fpr, ppv all'istante.

# Decisione sulla soglia

sens\_spec\_cutoff\_df\_wide %>%  
 filter(sensitivity > 0.47, sensitivity < 0.53)

## cutoff sensitivity specificity ppv  
## 1 0.1957971 0.4701012 0.9024513 0.3970474  
## 2 0.1955598 0.4710212 0.9024513 0.3975155  
## 3 0.1954559 0.4710212 0.9023256 0.3972071  
## 4 0.1953147 0.4719411 0.9023256 0.3976744  
## 5 0.1952826 0.4719411 0.9021999 0.3973664  
## 6 0.1952626 0.4719411 0.9020742 0.3970588  
## 7 0.1950500 0.4719411 0.9019485 0.3967517  
## 8 0.1949055 0.4719411 0.9018228 0.3964451  
## 9 0.1947558 0.4719411 0.9016970 0.3961390  
## 10 0.1947215 0.4719411 0.9015713 0.3958333  
## 11 0.1946618 0.4719411 0.9014456 0.3955281  
## 12 0.1945417 0.4719411 0.9013199 0.3952234  
## 13 0.1945238 0.4719411 0.9011942 0.3949192  
## 14 0.1944633 0.4719411 0.9010685 0.3946154  
## 15 0.1944410 0.4719411 0.9009428 0.3943121  
## 16 0.1942369 0.4719411 0.9008171 0.3940092  
## 17 0.1942008 0.4719411 0.9006914 0.3937068  
## 18 0.1940493 0.4719411 0.9005657 0.3934049  
## 19 0.1939006 0.4719411 0.9004400 0.3931034  
## 20 0.1938833 0.4719411 0.9003143 0.3928025  
## 21 0.1938698 0.4719411 0.9001886 0.3925019  
## 22 0.1937547 0.4719411 0.9000629 0.3922018  
## 23 0.1937476 0.4719411 0.8999371 0.3919022  
## 24 0.1934196 0.4728611 0.8999371 0.3923664  
## 25 0.1933346 0.4728611 0.8998114 0.3920671  
## 26 0.1931454 0.4728611 0.8996857 0.3917683  
## 27 0.1929933 0.4737810 0.8996857 0.3922315  
## 28 0.1929858 0.4737810 0.8995600 0.3919330  
## 29 0.1929833 0.4737810 0.8994343 0.3916350  
## 30 0.1929498 0.4737810 0.8993086 0.3913374  
## 31 0.1928349 0.4737810 0.8991829 0.3910402  
## 32 0.1926441 0.4737810 0.8990572 0.3907436  
## 33 0.1926038 0.4737810 0.8989315 0.3904473  
## 34 0.1925321 0.4737810 0.8988058 0.3901515  
## 35 0.1925120 0.4737810 0.8986801 0.3898562  
## 36 0.1923975 0.4737810 0.8985544 0.3895613  
## 37 0.1923506 0.4737810 0.8984287 0.3892668  
## 38 0.1921264 0.4737810 0.8983030 0.3889728  
## 39 0.1920746 0.4737810 0.8981772 0.3886792  
## 40 0.1917260 0.4737810 0.8980515 0.3883861  
## 41 0.1917227 0.4737810 0.8979258 0.3880934  
## 42 0.1916278 0.4737810 0.8978001 0.3878012  
## 43 0.1915795 0.4737810 0.8976744 0.3875094  
## 44 0.1913071 0.4737810 0.8975487 0.3872180  
## 45 0.1912966 0.4737810 0.8974230 0.3869271  
## 46 0.1912232 0.4747010 0.8974230 0.3873874  
## 47 0.1910853 0.4747010 0.8972973 0.3870968  
## 48 0.1910223 0.4747010 0.8971716 0.3868066  
## 49 0.1906739 0.4747010 0.8970459 0.3865169  
## 50 0.1906482 0.4747010 0.8969202 0.3862275  
## 51 0.1905339 0.4756210 0.8969202 0.3866866  
## 52 0.1905332 0.4756210 0.8967945 0.3863976  
## 53 0.1904836 0.4765409 0.8967945 0.3868559  
## 54 0.1904556 0.4765409 0.8966688 0.3865672  
## 55 0.1903851 0.4774609 0.8966688 0.3870246  
## 56 0.1903834 0.4774609 0.8965431 0.3867362  
## 57 0.1900495 0.4783809 0.8965431 0.3871929  
## 58 0.1898074 0.4793008 0.8965431 0.3876488  
## 59 0.1897627 0.4793008 0.8964173 0.3873606  
## 60 0.1897009 0.4802208 0.8964173 0.3878158  
## 61 0.1896233 0.4802208 0.8962916 0.3875278  
## 62 0.1895834 0.4802208 0.8961659 0.3872404  
## 63 0.1894827 0.4802208 0.8960402 0.3869533  
## 64 0.1894572 0.4802208 0.8959145 0.3866667  
## 65 0.1894220 0.4802208 0.8957888 0.3863805  
## 66 0.1893915 0.4802208 0.8956631 0.3860947  
## 67 0.1893890 0.4802208 0.8955374 0.3858093  
## 68 0.1893539 0.4811408 0.8955374 0.3862629  
## 69 0.1892683 0.4811408 0.8954117 0.3859779  
## 70 0.1891371 0.4811408 0.8952860 0.3856932  
## 71 0.1889929 0.4811408 0.8951603 0.3854090  
## 72 0.1888395 0.4811408 0.8950346 0.3851252  
## 73 0.1886635 0.4811408 0.8949089 0.3848418  
## 74 0.1885997 0.4811408 0.8947832 0.3845588  
## 75 0.1885880 0.4811408 0.8946574 0.3842763  
## 76 0.1885732 0.4811408 0.8945317 0.3839941  
## 77 0.1885500 0.4811408 0.8944060 0.3837124  
## 78 0.1884728 0.4811408 0.8942803 0.3834311  
## 79 0.1884018 0.4811408 0.8941546 0.3831502  
## 80 0.1882734 0.4820607 0.8941546 0.3836018  
## 81 0.1880996 0.4820607 0.8940289 0.3833211  
## 82 0.1874842 0.4820607 0.8939032 0.3830409  
## 83 0.1873088 0.4820607 0.8937775 0.3827611  
## 84 0.1872882 0.4820607 0.8936518 0.3824818  
## 85 0.1872003 0.4820607 0.8935261 0.3822028  
## 86 0.1870637 0.4820607 0.8934004 0.3819242  
## 87 0.1870618 0.4820607 0.8932747 0.3816460  
## 88 0.1868705 0.4820607 0.8931490 0.3813683  
## 89 0.1867157 0.4829807 0.8931490 0.3818182  
## 90 0.1863848 0.4829807 0.8930233 0.3815407  
## 91 0.1862615 0.4829807 0.8928975 0.3812636  
## 92 0.1862368 0.4829807 0.8927718 0.3809869  
## 93 0.1861681 0.4839006 0.8927718 0.3814358  
## 94 0.1861387 0.4839006 0.8926461 0.3811594  
## 95 0.1861190 0.4839006 0.8925204 0.3808834  
## 96 0.1860730 0.4839006 0.8923947 0.3806078  
## 97 0.1860720 0.4839006 0.8922690 0.3803326  
## 98 0.1860699 0.4839006 0.8921433 0.3800578  
## 99 0.1860094 0.4839006 0.8920176 0.3797834  
## 100 0.1859426 0.4839006 0.8918919 0.3795094  
## 101 0.1858069 0.4839006 0.8917662 0.3792358  
## 102 0.1857711 0.4839006 0.8916405 0.3789625  
## 103 0.1857483 0.4839006 0.8915148 0.3786897  
## 104 0.1857340 0.4839006 0.8913891 0.3784173  
## 105 0.1856945 0.4839006 0.8912634 0.3781452  
## 106 0.1856734 0.4839006 0.8911376 0.3778736  
## 107 0.1854986 0.4839006 0.8910119 0.3776023  
## 108 0.1853490 0.4848206 0.8910119 0.3780488  
## 109 0.1853091 0.4848206 0.8908862 0.3777778  
## 110 0.1852284 0.4848206 0.8907605 0.3775072  
## 111 0.1852251 0.4848206 0.8906348 0.3772369  
## 112 0.1850258 0.4857406 0.8906348 0.3776824  
## 113 0.1850118 0.4857406 0.8905091 0.3774124  
## 114 0.1849070 0.4866605 0.8905091 0.3778571  
## 115 0.1847903 0.4866605 0.8903834 0.3775874  
## 116 0.1845894 0.4866605 0.8902577 0.3773181  
## 117 0.1845804 0.4866605 0.8901320 0.3770492  
## 118 0.1844847 0.4866605 0.8900063 0.3767806  
## 119 0.1843794 0.4866605 0.8898806 0.3765125  
## 120 0.1843302 0.4875805 0.8898806 0.3769559  
## 121 0.1842873 0.4875805 0.8897549 0.3766880  
## 122 0.1842494 0.4875805 0.8896292 0.3764205  
## 123 0.1841244 0.4875805 0.8895035 0.3761533  
## 124 0.1840704 0.4875805 0.8893777 0.3758865  
## 125 0.1840500 0.4875805 0.8892520 0.3756201  
## 126 0.1838929 0.4875805 0.8891263 0.3753541  
## 127 0.1837885 0.4875805 0.8890006 0.3750885  
## 128 0.1837711 0.4885005 0.8890006 0.3755304  
## 129 0.1837540 0.4885005 0.8888749 0.3752650  
## 130 0.1835178 0.4885005 0.8887492 0.3750000  
## 131 0.1834813 0.4885005 0.8886235 0.3747354  
## 132 0.1834659 0.4885005 0.8884978 0.3744711  
## 133 0.1833811 0.4885005 0.8883721 0.3742072  
## 134 0.1833352 0.4885005 0.8882464 0.3739437  
## 135 0.1831999 0.4885005 0.8881207 0.3736805  
## 136 0.1830379 0.4885005 0.8879950 0.3734177  
## 137 0.1830150 0.4885005 0.8878693 0.3731553  
## 138 0.1828916 0.4885005 0.8877436 0.3728933  
## 139 0.1828902 0.4894204 0.8877436 0.3733333  
## 140 0.1828821 0.4894204 0.8876179 0.3730715  
## 141 0.1828435 0.4894204 0.8874921 0.3728101  
## 142 0.1827636 0.4894204 0.8873664 0.3725490  
## 143 0.1827182 0.4894204 0.8872407 0.3722883  
## 144 0.1825815 0.4894204 0.8871150 0.3720280  
## 145 0.1824414 0.4894204 0.8869893 0.3717680  
## 146 0.1823990 0.4894204 0.8868636 0.3715084  
## 147 0.1823473 0.4903404 0.8868636 0.3719470  
## 148 0.1822290 0.4903404 0.8867379 0.3716876  
## 149 0.1820748 0.4903404 0.8866122 0.3714286  
## 150 0.1820255 0.4903404 0.8864865 0.3711699  
## 151 0.1820055 0.4903404 0.8863608 0.3709116  
## 152 0.1819762 0.4912603 0.8863608 0.3713491  
## 153 0.1818428 0.4921803 0.8863608 0.3717860  
## 154 0.1816851 0.4921803 0.8862351 0.3715278  
## 155 0.1816802 0.4921803 0.8861094 0.3712700  
## 156 0.1816729 0.4921803 0.8859837 0.3710125  
## 157 0.1816559 0.4931003 0.8859837 0.3714484  
## 158 0.1816094 0.4931003 0.8858580 0.3711911  
## 159 0.1814125 0.4931003 0.8857322 0.3709343  
## 160 0.1812348 0.4931003 0.8856065 0.3706777  
## 161 0.1812283 0.4931003 0.8854808 0.3704216  
## 162 0.1811851 0.4931003 0.8853551 0.3701657  
## 163 0.1811731 0.4931003 0.8852294 0.3699103  
## 164 0.1809476 0.4931003 0.8851037 0.3696552  
## 165 0.1808707 0.4931003 0.8849780 0.3694004  
## 166 0.1808272 0.4931003 0.8848523 0.3691460  
## 167 0.1808121 0.4931003 0.8847266 0.3688919  
## 168 0.1806823 0.4931003 0.8846009 0.3686382  
## 169 0.1804976 0.4931003 0.8844752 0.3683849  
## 170 0.1803807 0.4931003 0.8843495 0.3681319  
## 171 0.1798724 0.4931003 0.8842238 0.3678792  
## 172 0.1798114 0.4931003 0.8840981 0.3676269  
## 173 0.1796700 0.4940202 0.8840981 0.3680603  
## 174 0.1795897 0.4940202 0.8839723 0.3678082  
## 175 0.1795726 0.4940202 0.8838466 0.3675565  
## 176 0.1795476 0.4940202 0.8837209 0.3673051  
## 177 0.1795227 0.4940202 0.8835952 0.3670540  
## 178 0.1794895 0.4940202 0.8834695 0.3668033  
## 179 0.1792405 0.4940202 0.8833438 0.3665529  
## 180 0.1790263 0.4940202 0.8832181 0.3663029  
## 181 0.1790255 0.4949402 0.8832181 0.3667348  
## 182 0.1789477 0.4949402 0.8830924 0.3664850  
## 183 0.1789122 0.4949402 0.8829667 0.3662355  
## 184 0.1788906 0.4958602 0.8829667 0.3666667  
## 185 0.1787022 0.4958602 0.8828410 0.3664174  
## 186 0.1783559 0.4958602 0.8827153 0.3661685  
## 187 0.1779963 0.4958602 0.8825896 0.3659199  
## 188 0.1779888 0.4958602 0.8824639 0.3656716  
## 189 0.1779712 0.4958602 0.8823382 0.3654237  
## 190 0.1778612 0.4967801 0.8823382 0.3658537  
## 191 0.1777275 0.4967801 0.8822124 0.3656060  
## 192 0.1776705 0.4967801 0.8820867 0.3653586  
## 193 0.1775395 0.4967801 0.8819610 0.3651116  
## 194 0.1774334 0.4967801 0.8818353 0.3648649  
## 195 0.1773975 0.4967801 0.8817096 0.3646185  
## 196 0.1773564 0.4967801 0.8815839 0.3643725  
## 197 0.1772714 0.4967801 0.8814582 0.3641268  
## 198 0.1771082 0.4967801 0.8813325 0.3638814  
## 199 0.1770839 0.4967801 0.8812068 0.3636364  
## 200 0.1769767 0.4967801 0.8810811 0.3633917  
## 201 0.1769764 0.4977001 0.8810811 0.3638198  
## 202 0.1768164 0.4986201 0.8810811 0.3642473  
## 203 0.1766207 0.4995400 0.8810811 0.3646743  
## 204 0.1765403 0.4995400 0.8809554 0.3644295  
## 205 0.1764683 0.4995400 0.8808297 0.3641851  
## 206 0.1764245 0.4995400 0.8807040 0.3639410  
## 207 0.1763968 0.4995400 0.8805783 0.3636973  
## 208 0.1763497 0.4995400 0.8804525 0.3634538  
## 209 0.1763027 0.4995400 0.8803268 0.3632107  
## 210 0.1761450 0.4995400 0.8802011 0.3629679  
## 211 0.1760560 0.4995400 0.8800754 0.3627255  
## 212 0.1758297 0.5004600 0.8800754 0.3631509  
## 213 0.1754478 0.5013799 0.8800754 0.3635757  
## 214 0.1754413 0.5013799 0.8799497 0.3633333  
## 215 0.1753525 0.5013799 0.8798240 0.3630913  
## 216 0.1753433 0.5013799 0.8796983 0.3628495  
## 217 0.1752551 0.5013799 0.8795726 0.3626081  
## 218 0.1749495 0.5013799 0.8794469 0.3623670  
## 219 0.1749489 0.5022999 0.8794469 0.3627907  
## 220 0.1748964 0.5032199 0.8794469 0.3632138  
## 221 0.1748179 0.5032199 0.8793212 0.3629728  
## 222 0.1746895 0.5032199 0.8791955 0.3627321  
## 223 0.1745712 0.5032199 0.8790698 0.3624917  
## 224 0.1744451 0.5041398 0.8790698 0.3629139  
## 225 0.1743366 0.5050598 0.8790698 0.3633355  
## 226 0.1743240 0.5050598 0.8789441 0.3630952  
## 227 0.1742418 0.5050598 0.8788184 0.3628553  
## 228 0.1741868 0.5050598 0.8786926 0.3626156  
## 229 0.1741635 0.5050598 0.8785669 0.3623762  
## 230 0.1740890 0.5050598 0.8784412 0.3621372  
## 231 0.1740206 0.5050598 0.8783155 0.3618985  
## 232 0.1738609 0.5050598 0.8781898 0.3616601  
## 233 0.1737706 0.5050598 0.8780641 0.3614220  
## 234 0.1736390 0.5059798 0.8780641 0.3618421  
## 235 0.1735827 0.5059798 0.8778127 0.3613666  
## 236 0.1735091 0.5068997 0.8776870 0.3615486  
## 237 0.1735067 0.5068997 0.8775613 0.3613115  
## 238 0.1733971 0.5068997 0.8774356 0.3610747  
## 239 0.1733708 0.5068997 0.8773099 0.3608382  
## 240 0.1732165 0.5068997 0.8771842 0.3606021  
## 241 0.1730730 0.5068997 0.8770585 0.3603663  
## 242 0.1730543 0.5068997 0.8769327 0.3601307  
## 243 0.1728905 0.5068997 0.8768070 0.3598955  
## 244 0.1728752 0.5068997 0.8766813 0.3596606  
## 245 0.1728086 0.5068997 0.8765556 0.3594260  
## 246 0.1727111 0.5068997 0.8764299 0.3591917  
## 247 0.1726990 0.5068997 0.8763042 0.3589577  
## 248 0.1726959 0.5068997 0.8761785 0.3587240  
## 249 0.1726769 0.5068997 0.8760528 0.3584906  
## 250 0.1725843 0.5078197 0.8760528 0.3589077  
## 251 0.1725353 0.5078197 0.8759271 0.3586745  
## 252 0.1722684 0.5078197 0.8758014 0.3584416  
## 253 0.1722272 0.5078197 0.8756757 0.3582090  
## 254 0.1721814 0.5078197 0.8755500 0.3579767  
## 255 0.1721168 0.5078197 0.8754243 0.3577447  
## 256 0.1720608 0.5078197 0.8752986 0.3575130  
## 257 0.1719836 0.5078197 0.8751728 0.3572816  
## 258 0.1719501 0.5078197 0.8750471 0.3570505  
## 259 0.1717475 0.5087397 0.8750471 0.3574661  
## 260 0.1716161 0.5087397 0.8749214 0.3572351  
## 261 0.1715939 0.5087397 0.8747957 0.3570045  
## 262 0.1715805 0.5096596 0.8747957 0.3574194  
## 263 0.1715549 0.5096596 0.8746700 0.3571889  
## 264 0.1714694 0.5105796 0.8745443 0.3573728  
## 265 0.1714561 0.5105796 0.8744186 0.3571429  
## 266 0.1714297 0.5105796 0.8742929 0.3569132  
## 267 0.1713537 0.5105796 0.8741672 0.3566838  
## 268 0.1713497 0.5114995 0.8741672 0.3570970  
## 269 0.1712735 0.5124195 0.8740415 0.3572803  
## 270 0.1712314 0.5124195 0.8739158 0.3570513  
## 271 0.1711766 0.5124195 0.8737901 0.3568225  
## 272 0.1711452 0.5124195 0.8736644 0.3565941  
## 273 0.1710837 0.5124195 0.8735387 0.3563660  
## 274 0.1709513 0.5133395 0.8735387 0.3567775  
## 275 0.1709441 0.5142594 0.8735387 0.3571885  
## 276 0.1708085 0.5142594 0.8734129 0.3569604  
## 277 0.1706882 0.5142594 0.8732872 0.3567326  
## 278 0.1705924 0.5142594 0.8731615 0.3565051  
## 279 0.1704851 0.5151794 0.8731615 0.3569152  
## 280 0.1703998 0.5151794 0.8730358 0.3566879  
## 281 0.1703808 0.5151794 0.8729101 0.3564609  
## 282 0.1703787 0.5151794 0.8727844 0.3562341  
## 283 0.1703365 0.5151794 0.8726587 0.3560076  
## 284 0.1703038 0.5151794 0.8725330 0.3557814  
## 285 0.1702397 0.5160994 0.8725330 0.3561905  
## 286 0.1702350 0.5170193 0.8725330 0.3565990  
## 287 0.1701867 0.5170193 0.8724073 0.3563729  
## 288 0.1700537 0.5170193 0.8722816 0.3561470  
## 289 0.1699724 0.5170193 0.8721559 0.3559215  
## 290 0.1696751 0.5170193 0.8720302 0.3556962  
## 291 0.1696601 0.5170193 0.8719045 0.3554712  
## 292 0.1695513 0.5170193 0.8717788 0.3552465  
## 293 0.1695037 0.5170193 0.8716530 0.3550221  
## 294 0.1693031 0.5170193 0.8715273 0.3547980  
## 295 0.1692256 0.5179393 0.8715273 0.3552050  
## 296 0.1690918 0.5188592 0.8715273 0.3556116  
## 297 0.1689219 0.5188592 0.8714016 0.3553875  
## 298 0.1688318 0.5188592 0.8712759 0.3551637  
## 299 0.1687930 0.5188592 0.8711502 0.3549402  
## 300 0.1687845 0.5188592 0.8710245 0.3547170  
## 301 0.1686660 0.5188592 0.8708988 0.3544940  
## 302 0.1686325 0.5188592 0.8707731 0.3542714  
## 303 0.1685214 0.5188592 0.8706474 0.3540490  
## 304 0.1684659 0.5188592 0.8705217 0.3538269  
## 305 0.1684443 0.5197792 0.8705217 0.3542320  
## 306 0.1683749 0.5197792 0.8703960 0.3540100  
## 307 0.1683564 0.5197792 0.8702703 0.3537884  
## 308 0.1683474 0.5197792 0.8701446 0.3535670  
## 309 0.1683175 0.5206992 0.8701446 0.3539712  
## 310 0.1682988 0.5206992 0.8700189 0.3537500  
## 311 0.1682944 0.5206992 0.8698931 0.3535290  
## 312 0.1680783 0.5206992 0.8697674 0.3533084  
## 313 0.1680542 0.5216191 0.8697674 0.3537118  
## 314 0.1679771 0.5216191 0.8696417 0.3534913  
## 315 0.1679515 0.5216191 0.8695160 0.3532710  
## 316 0.1678603 0.5216191 0.8693903 0.3530511  
## 317 0.1677304 0.5216191 0.8692646 0.3528314  
## 318 0.1675829 0.5216191 0.8691389 0.3526119  
## 319 0.1674942 0.5216191 0.8690132 0.3523928  
## 320 0.1674818 0.5216191 0.8688875 0.3521739  
## 321 0.1673908 0.5216191 0.8687618 0.3519553  
## 322 0.1672526 0.5216191 0.8686361 0.3517370  
## 323 0.1671785 0.5225391 0.8686361 0.3521389  
## 324 0.1668779 0.5225391 0.8685104 0.3519207  
## 325 0.1665776 0.5234591 0.8685104 0.3523220  
## 326 0.1663791 0.5234591 0.8683847 0.3521040  
## 327 0.1662760 0.5234591 0.8682590 0.3518862  
## 328 0.1662133 0.5234591 0.8681332 0.3516687  
## 329 0.1662101 0.5234591 0.8680075 0.3514515  
## 330 0.1659588 0.5234591 0.8678818 0.3512346  
## 331 0.1658278 0.5234591 0.8677561 0.3510179  
## 332 0.1657972 0.5243790 0.8677561 0.3514180  
## 333 0.1657825 0.5243790 0.8676304 0.3512015  
## 334 0.1656505 0.5243790 0.8675047 0.3509852  
## 335 0.1656404 0.5243790 0.8673790 0.3507692  
## 336 0.1656342 0.5243790 0.8672533 0.3505535  
## 337 0.1654256 0.5243790 0.8671276 0.3503380  
## 338 0.1654109 0.5243790 0.8670019 0.3501229  
## 339 0.1653623 0.5252990 0.8670019 0.3505218  
## 340 0.1653362 0.5252990 0.8668762 0.3503067  
## 341 0.1652870 0.5252990 0.8667505 0.3500920  
## 342 0.1652616 0.5252990 0.8666248 0.3498775  
## 343 0.1651979 0.5262190 0.8666248 0.3502756  
## 344 0.1651610 0.5262190 0.8664991 0.3500612  
## 345 0.1651422 0.5271389 0.8664991 0.3504587  
## 346 0.1651215 0.5271389 0.8663734 0.3502445  
## 347 0.1650041 0.5271389 0.8662476 0.3500305  
## 348 0.1648698 0.5271389 0.8661219 0.3498168  
## 349 0.1648462 0.5271389 0.8658705 0.3493902  
## 350 0.1647660 0.5271389 0.8657448 0.3491773  
## 351 0.1647503 0.5271389 0.8656191 0.3489647  
## 352 0.1647416 0.5280589 0.8656191 0.3493609  
## 353 0.1647407 0.5280589 0.8654934 0.3491484  
## 354 0.1647154 0.5280589 0.8653677 0.3489362  
## 355 0.1646955 0.5280589 0.8652420 0.3487242  
## 356 0.1646540 0.5289788 0.8652420 0.3491196  
## 357 0.1644638 0.5289788 0.8651163 0.3489078  
## 358 0.1644053 0.5289788 0.8649906 0.3486962  
## 359 0.1641859 0.5289788 0.8648649 0.3484848  
## 360 0.1640599 0.5289788 0.8647392 0.3482738  
## 361 0.1639702 0.5289788 0.8646135 0.3480630  
## 362 0.1639628 0.5289788 0.8644877 0.3478524  
## 363 0.1638220 0.5289788 0.8643620 0.3476421  
## 364 0.1638197 0.5298988 0.8643620 0.3480363  
## 365 0.1637209 0.5298988 0.8642363 0.3478261  
## 366 0.1637158 0.5298988 0.8641106 0.3476162  
## 367 0.1637002 0.5298988 0.8639849 0.3474065

#Quando cutoff = 0.1758297 allora:  
  
sens\_spec\_cutoff\_df\_wide %>%  
 filter(cutoff > 0.1758, cutoff < 0.1759)

## cutoff sensitivity specificity ppv  
## 1 0.1758297 0.50046 0.8800754 0.3631509

#ho un tpr del 50%, un tnr del 88% e un ppv del 36%

La mia soglia è 0.1758

# Confusion matrix

#usa prediction\_test\_obj di rocr, vedi sopra e https://cran.r-project.org/web/packages/ROCR/ROCR.pdf  
test\_set\_vsa$yhat <- factor(as.numeric(test\_set\_vsa$glm3\_vsa\_predictions\_test > 0.1758))  
test\_set\_vsa$yact <- factor(as.numeric(test\_set\_vsa$y == "yes"))  
conf\_matrix <- confusionMatrix(test\_set\_vsa$yhat, test\_set\_vsa$yact, positive ="1")  
conf\_matrix

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 7001 543  
## 1 954 544  
##   
## Accuracy : 0.8344   
## 95% CI : (0.8266, 0.842)  
## No Information Rate : 0.8798   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.3271   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.50046   
## Specificity : 0.88008   
## Pos Pred Value : 0.36315   
## Neg Pred Value : 0.92802   
## Prevalence : 0.12022   
## Detection Rate : 0.06016   
## Detection Prevalence : 0.16567   
## Balanced Accuracy : 0.69027   
##   
## 'Positive' Class : 1   
##

# Diagnostic likelihood ratio

Non so la prevalence se va calcolata su tutto il dataset o solo sul training. Io direi solo sul training, perché nella CV è il campione che ho a disposizione per qualunque stima. peraltro è identica a quella di tutto il dataset.

ppv\_def <-sens\_spec\_cutoff\_df\_wide$ppv[sens\_spec\_cutoff\_df\_wide$cutoff > 0.1758 & sens\_spec\_cutoff\_df\_wide$cutoff < 0.1759]  
prevalence <- mean(training\_set\_vsa$y == "yes")  
DLRp <- ( (ppv\_def/(1-ppv\_def)) / (prevalence/(1-prevalence)) )  
DLRp #sopra 4, buono.

## [1] 4.30025

# Lift chart cumulato

lift\_df <- data.frame(yact = test\_set\_vsa$yact, y\_pred\_prob = test\_set\_vsa$glm3\_vsa\_predictions\_test)  
lift\_df\_sum <- lift\_df %>%  
 arrange(y\_pred\_prob) %>%  
 mutate(groups = ntile(y\_pred\_prob, 10)) %>%  
 group\_by(groups) %>%  
 summarise(n\_clienti = n(), n\_y = sum(yact == "1")) %>%  
 arrange(desc(groups)) %>%  
 mutate(perc\_clienti = n\_clienti / sum(n\_clienti), perc\_clienti\_positivi = n\_y / sum(n\_y), perc\_cum\_clienti\_pos\_mod = cumsum(perc\_clienti\_positivi), perc\_cum\_clienti = cumsum(perc\_clienti), perc\_cum\_clienti\_pos\_exp = perc\_cum\_clienti) %>%  
 select(groups, perc\_cum\_clienti, perc\_cum\_clienti\_pos\_mod, perc\_cum\_clienti\_pos\_exp) %>%  
 gather(key = type\_indicator, value\_indicator, perc\_cum\_clienti\_pos\_mod:perc\_cum\_clienti\_pos\_exp) %>%  
 ggplot(aes(x = perc\_cum\_clienti, y = value\_indicator, col = type\_indicator)) +   
 geom\_point (size = 3) +   
 geom\_line() +  
 scale\_x\_continuous(breaks = seq(0, 1, 0.1)) +  
 scale\_y\_continuous(breaks = seq(0, 1, 0.1)) +  
 geom\_vline(xintercept = 0.17)  
lift\_df\_sum