

# Pilgrim Bank

Luca Bajardi e Francesca Collini

31/07/2020

Carichiamo i dati leggendo il file csv e settiamo il seme del generatore pseudo-casuale così da avere i risultati sempre uguali.

```
rm(list =ls())
set.seed(1)
Pilgrim = read.csv(file = "PilgrimABC.csv", header=T)
attach(Pilgrim)
```

Osserviamo che il nostro dataset contiene 31634 osservazioni e 11 variabili (le ultime due non sono interessanti ai fini della nostra analisi). I dati sono relativi a due anni, 1999 e 2000, per ciascuno abbiamo a disposizione un'informazione relativa al profitto (o alla perdita) di un determinato cliente in quell'anno e un'informazione che ci dice se quel cliente in quell'anno ha utilizzato l'opzione di online banking oppure no: la variabile Online quindi è una variabile binaria. Abbiamo poi a disposizione altre informazioni relative al cliente:

- AGE: variabile categorica che indica a che fascia di età appartiene il cliente (1 = less than 15 years; 2 = 15-24 years; 3 = 25-34 years; 4 = 35-44 years; 5 = 45-54 years; 6 = 55-64 years; 7 = 65 years and older.)
- INCOME: variabile categorica che indica il range del reddito di ciascun cliente (1 = less than \$15,000; 2 = \$15,000 – \$19,999; 3 = \$20,000 – \$29,999; 4 = \$30,000 – \$39,999; 5 = \$40,000 – \$49,999; 6 = \$50,000 – \$74,999; 7 = \$75,000 – \$99,999; 8 = \$100,000 – \$124,999; 9 = \$125,000 and more.)
- TENURE: rappresenta l'età di servizio
- DISTRICT: variabile categorica che indica uno delle tre regioni geografiche in cui si trova il cliente.

```
dim(Pilgrim)
```

```
## [1] 31634      11
names(Pilgrim[, 1:9])
```

```
## [1] "ID"          "X9Profit"     "X9Online"     "X9Age"        "X9Inc"
## [6] "X9Tenure"    "X9District"   "X0Profit"     "X0Online"
```

Rinominiamo le colonne per evitare problemi con i nomi:

```
Profit=Pilgrim$X9Profit
Online=Pilgrim$X9Online
Age=Pilgrim$X9Age
Income=Pilgrim$X9Inc
Tenure=Pilgrim$X9Tenure
District=Pilgrim$X9District
```

```
head(Pilgrim)
```

```
##   ID X9Profit X9Online X9Age X9Inc X9Tenure X9District X0Profit X0Online
## 1  1       21       0    NA    NA     6.33      1200      NA      NA
## 2  2      -6       0     6     3    29.50      1200     -32      0
```

```

## 3 3 -49 1 5 5 26.41 1100 -22 1
## 4 4 -4 0 NA NA 2.25 1200 NA NA
## 5 5 -61 0 2 9 9.91 1200 -4 0
## 6 6 -38 0 NA 3 2.33 1300 14 0
## X9Billpay X0Billpay
## 1 0 NA
## 2 0 0
## 3 0 0
## 4 0 NA
## 5 0 0
## 6 0 0

```

Dal summary del profitto possiamo facilmente osservare che la distribuzione è molto asimmetrica perché media e mediana sono molto diverse tra loro. Inoltre notiamo che tutto il primo quartile è negativo.

```
summary(Profit)
```

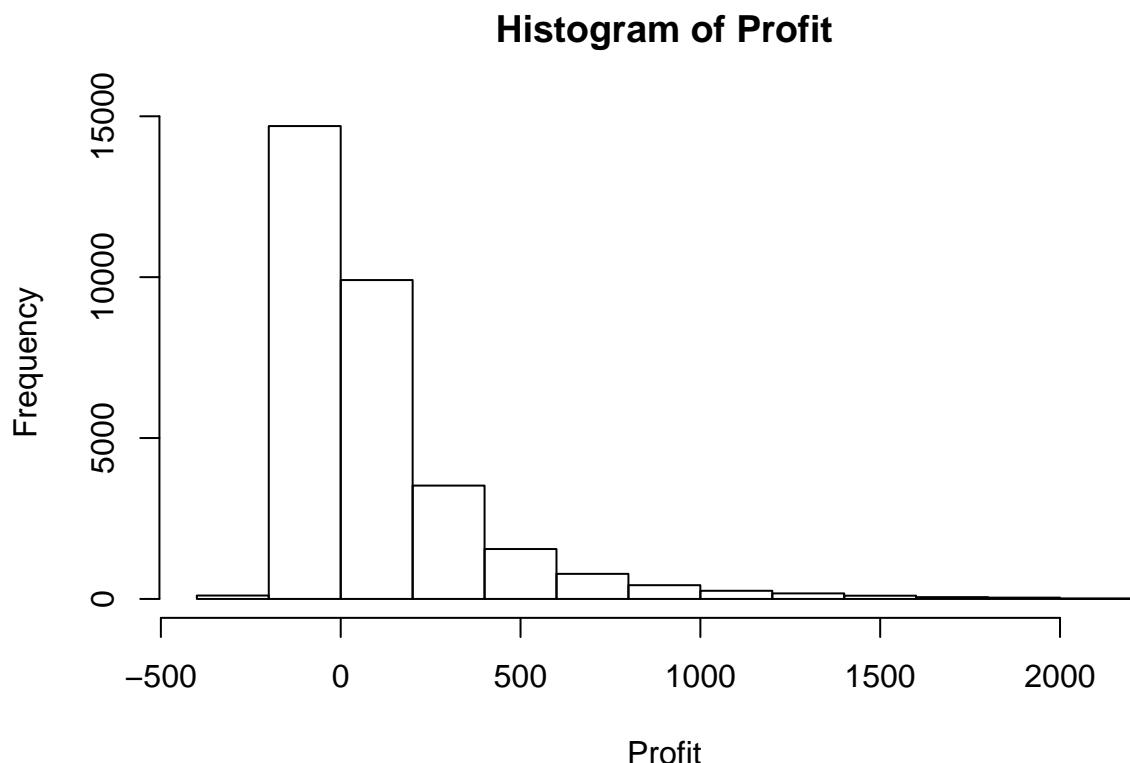
```

##   Min. 1st Qu. Median Mean 3rd Qu. Max.
## -221.0 -34.0 9.0 111.5 164.0 2071.0

```

Dall'istogramma sul profitto nel 1999 possiamo notare che c'è un discreto numero di clienti che mi fa perdere e che l'istogramma è molto scodato a destra quindi ci sono pochissimi clienti che mi fanno guadagnare molto.

```
hist(Profit)
```



Possiamo notare che ci sono 16832 clienti che generano profitto sulle 31634 presenti nel dataset.

```

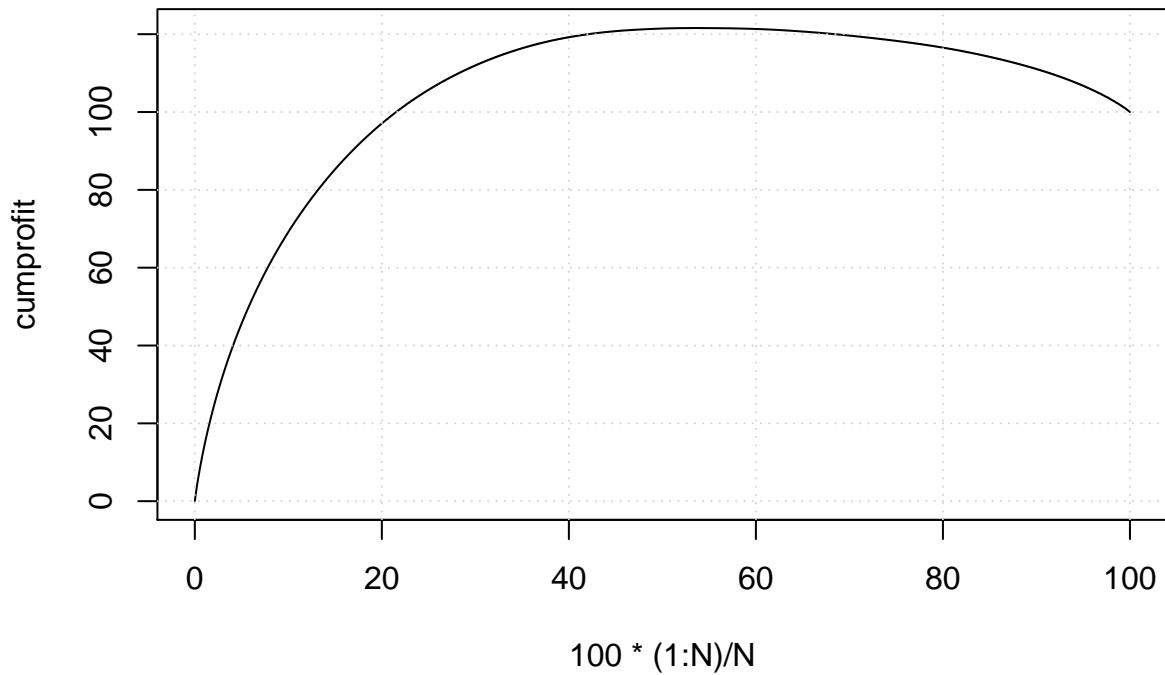
N=length(Profit)
Nprofitable = sum(Profit>0)
cat('profitable = ', Nprofitable, ' out of ', N, '\n')

```

```
## profitable = 16832 out of 31634
```

Tramite la curva di Pareto possiamo confermare che poche persone fanno guadagnare tanto, mentre la maggior parte delle persone fa guadagnare poco o addirittura fa perdere guadagno.

```
cumprofit=cumsum(sort(Profit,decreasing=TRUE))*100/sum(Profit)
plot(100*(1:N)/N,cumprofit,type='l')
grid()
```



Dall'analisi dei profitti medi possiamo notare che il profitto generato da chi utilizza i servizi online è maggiore rispetto a chi li utilizza offline.

```
cat('average profit ', mean(Profit), '\n')

## average profit 111.5027

ProfitOnline = Profit[Online==1]
cat('average profit ON', mean(ProfitOnline), '\n')

## average profit ON 116.6668

ProfitOffline = Profit[Online==0]
cat('average profit OFF', mean(ProfitOffline), '\n')

## average profit OFF 110.7862
```

Quello che vorremmo capire è se questa differenza è stocasticamente significativa, quindi se davvero coloro che utilizzano i servizi online mi permettono di guadagnare di più. Per questo motivo possiamo eseguire un `t.test` sulle due popolazioni, i clienti che utilizzano il servizio online e coloro che non lo usano, e vado a vedere quanto vale il p-value. L'ipotesi nulla in questo caso è che le medie siano uguali e che quindi la

differenza tra le medie delle due popolazioni non sia significativa. Analizziamo inoltre l'intervallo di confidenza per vedere se i dati sono sufficienti o no. Infatti se intervallo fosse troppo grande vorrebbe dire che avrei bisogno di più clienti per formulare una teoria generale.

```
cat('Conf int profit ', t.test(Profit)$conf.int, '\n')

## Conf int profit 108.496 114.5094

cat('p-value difference ', t.test(ProfitOnline, ProfitOffline)$p.value, '\n')

## p-value difference 0.2254368
```

Risulta esserci una differenza di circa 6 dollari tra le due popolazioni, quindi non risulta più di tanto significativa dal lato business, ma anche il p-value è maggiore di 0.05 e quindi possiamo concludere che questa differenza non è significativa nemmeno dal punto di vista statistico, e quindi non rifiutiamo l'ipotesi nulla sulla differenza tra le medie.

Possiamo ottenere lo stesso risultato impostando il modello di regressione classico:

```
mod = lm(Profit ~ Online)
summary(mod)

##
## Call:
## lm(formula = Profit ~ Online)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -337.67 -144.79 -101.79   52.21 1960.21 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 110.786     1.637   67.678 <2e-16 ***
## Online       5.881      4.690    1.254    0.21    
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 272.8 on 31632 degrees of freedom
## Multiple R-squared:  4.97e-05, Adjusted R-squared:  1.809e-05 
## F-statistic: 1.572 on 1 and 31632 DF,  p-value: 0.2099
```

Dal summary del modello, possiamo osservare che il valore dell'intercetta è la media del profitto tra coloro che operano offline e il valore aggiunto di quelli che operano online è di quasi 6 dollari come la differenza tra la media di quelli che operano online e la media di quelli che operano offline. Anche in questo caso però, il p-value è superiore a 0.05 come in precedenza. Questo significa che la variabile Online non è significativa. Tuttavia, questo potrebbe essere dovuto al fatto che stiamo mettendo insieme clienti molto diversi tra loro e quindi considerando un unico modello non riusciamo a distinguere bene i singoli effetti. Per questo introduciamo l'età nel modello di regressione.

La variabile **Age** è riconosciuta numerica anche se in realtà è categorica. Nell'analisi dovremo trasformarla in **factor** perché altrimenti vi è troppa influenza dell'ordinamento delle variabili categoriali.

```
Age1 = as.factor(Age)
summary(Age1)

## 1 2 3 4 5 6 7 NA's
## 710 3650 5390 5376 3236 2290 2693 8289
```

Possiamo notare che ci sono 8289 osservazioni in cui non è presente l'età, infatti nel **summary** sottostante

possiamo notare che nonostante ci siano 32 mila osservazioni ci sono solo 23 mila gradi di libertà in quanto quelle con i missing values sono eliminate automaticamente.

```
mod = lm(Profit ~ Online+Age1)
summary(mod)

##
## Call:
## lm(formula = Profit ~ Online + Age1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -404.52 -162.90  -84.62   68.80 1952.10 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.802     10.485  -0.172   0.864    
## Online       27.246     5.519   4.937 8.01e-07 *** 
## Age12        54.425    11.401   4.774 1.82e-06 *** 
## Age13        112.699   11.098   10.155 < 2e-16 *** 
## Age14        133.820   11.103   12.053 < 2e-16 *** 
## Age15        144.986   11.531   12.574 < 2e-16 *** 
## Age16        160.844   11.965   13.443 < 2e-16 *** 
## Age17        193.072   11.757   16.422 < 2e-16 *** 
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
##
## Residual standard error: 277.9 on 23337 degrees of freedom
##   (8289 observations deleted due to missingness)
## Multiple R-squared:  0.02494,   Adjusted R-squared:  0.02464 
## F-statistic: 85.26 on 7 and 23337 DF,  p-value: < 2.2e-16
```

Ora sulle fasce d'età vi è una significatività sia statistica che di business, infatti le diverse età potrebbero implicare diversi redditi, in linea generale un giovane tende ad usare di più l'online banking ma ha un reddito minore e quindi fa guadagnare di meno.

Posso notare che a prescindere dall'utilizzo dell'online o dell'offline i giovani sono meno profittevoli degli anziani:

```
lapply(split(Profit,as.factor(Age)),mean)
```

```
## $`1` 
## [1] 3.45493
## 
## $`2` 
## [1] 58.48959
## 
## $`3` 
## [1] 115.1122
## 
## $`4` 
## [1] 135.6618
## 
## $`5` 
## [1] 145.7596
## 
```

```

## $`6`
## [1] 160.41
##
## $`7`
## [1] 192.2614

```

Inoltre il 20% dei giovani usa l'online, questa percentuale scende per le fasce di età successive fino ad arrivare a poco più del 3%.

```
lapply(split(Online,as.factor(Age)),mean)
```

```

## $`1`
## [1] 0.1929577
##
## $`2`
## [1] 0.2153425
##
## $`3`
## [1] 0.154731
##
## $`4`
## [1] 0.1337426
##
## $`5`
## [1] 0.09456119
##
## $`6`
## [1] 0.05021834
##
## $`7`
## [1] 0.03639064

```

Ci potrebbe essere un bias dovuto ai dati mancanti. Infatti ci sono molte osservazioni in cui non abbiamo l'informazione sull'età. Non sappiamo perché questi dati sono mancanti, anche se è difficile pensare che sia semplicemente dovuto a delle dimenticanze o sviste in fase di raccolta dei dati, forse avrebbe più senso pensare ai conti cointestati oppure al fatto che il conto venga intestato ad una società.

```
sum(is.na(Age))
```

```

## [1] 8289
AgeGiven = ifelse(is.na(Age),0,1) # 0 dove c'è NA, 1 se c'è l'età
mod = lm(Profit ~ AgeGiven)
summary(mod)

```

```

##
## Call:
## lm(formula = Profit ~ AgeGiven)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -346.19  -150.19   -90.96   50.81 1961.04
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 72.962     2.986   24.43 <2e-16 ***
## AgeGiven    52.224     3.476   15.02 <2e-16 ***

```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 271.9 on 31632 degrees of freedom
## Multiple R-squared: 0.007085, Adjusted R-squared: 0.007054
## F-statistic: 225.7 on 1 and 31632 DF, p-value: < 2.2e-16

```

C'è una differenza statisticamente significativa sul profitto tra le osservazioni in cui è presente l'informazione sull'età e dove non c'è. La variabile AgeGiven infatti risulta significativa, la distribuzione tra le due popolazioni di conseguenza non risulta omogenea. Quindi, eliminando i dati dove manca l'età, stiamo distorcendo l'analisi, perché queste medie sono "sporcate". Infatti c'è una profitabilità media più alta tra chi mi ha dato l'età rispetto a chi non me l'ha data.

**LUCA SI È FERMATO QUA** Possiamo provare diversi metodi per cercare di recuperare quelle osservazioni dove l'età non è presente, in modo da includerle comunque nell'analisi. Se avessimo delle righe complete ed alcune con un solo campo mancante, potremmo costruire un modello di regressione in modo da prevedere quel valore. Una possibile alternativa, più immediata, è quella di sostituire i valori mancanti con i valori nulli. Questa strada sembra abbastanza convincente, quando l'età è una variabile categorica e possiamo quindi scegliere un valore arbitrario. Quando invece la variabile è numerica non ha molto senso, perché equivale a dire che coloro che non hanno fornito l'età, risultano neonati.

```

AgeZero = ifelse(is.na(Age), 0, Age)
table(AgeZero)

## AgeZero
##   0   1   2   3   4   5   6   7
## 8289  710 3650 5390 5376 3236 2290 2693

mod = lm(Profit ~ Online+AgeZero)
summary(mod)

## 
## Call:
## lm(formula = Profit ~ Online + AgeZero)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max
## -393.91 -147.07  -82.03   49.97 1976.97
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 57.0311    2.6014  21.923 < 2e-16 ***
## Online      13.7925    4.6487   2.967  0.00301 ** 
## AgeZero     17.6803    0.6697  26.402 < 2e-16 ***
## ---      
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 269.9 on 31631 degrees of freedom
## Multiple R-squared: 0.02161, Adjusted R-squared: 0.02155 
## F-statistic: 349.3 on 2 and 31631 DF, p-value: < 2.2e-16

```

I valori dei coefficienti sono dovuti al fatto che in modo arbitrario abbiamo scelto di sostituire i valori mancanti con il valore nullo. # Replace missing with mean Una possibile alternativa è quella di sostituire i valori mancanti con la media dell'età calcolata sui valori presenti.

```

mm = mean(Age, na.rm=TRUE)#non consideriamo i valori mancanti
AgeAverage = ifelse(is.na(Age), mm, Age)

```

```



```

Ovviamente osserviamo che i valori dei coefficienti cambiano, ed in particolare cambia il contributo della variabile Online, quindi a seconda di come tappo il buco, di come scelgo di inserire i valori mancanti nel modello, otteniamo un risultato diverso. Questo ci suggerisce che probabilmente, non è questa la strada giusta per procedere.

```

mod = lm(Profit ~ Online+AgeZero+AgeGiven)
summary(mod)

## 
## Call:
## lm(formula = Profit ~ Online + AgeZero + AgeGiven)
## 
## Residuals:
##   Min     1Q   Median     3Q    Max 
## -409.34 -144.14  -82.69   52.07 1967.12 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  70.926     3.000  23.643 < 2e-16 *** 
## Online       19.649     4.685   4.194 2.75e-05 *** 
## AgeZero      25.603     1.086  23.582 < 2e-16 *** 
## AgeGiven    -51.849     5.598  -9.263 < 2e-16 *** 
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

## 

```

```

## Residual standard error: 269.5 on 31630 degrees of freedom
## Multiple R-squared:  0.02426,   Adjusted R-squared:  0.02417
## F-statistic: 262.1 on 3 and 31630 DF,  p-value: < 2.2e-16

```

Per tentare di evitare questo problema potremmo considerare i modelli ottenuti tenendo conto sia della variabile Age opportunamente modificata, sia del fatto che la variabile Age venga fornita oppure no. Questo ci permette di considerare una variabile categorica che indica la presenza o meno dell'informazione sull'età, quindi non importa più di tanto come vado a rimepire il buco dell'informazione mancante.

```

mod = lm(Profit ~ Online+AgeAverage+AgeGiven)
summary(mod)

```

```

##
## Call:
## lm(formula = Profit ~ Online + AgeAverage + AgeGiven)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -409.34 -144.14  -82.69   52.07 1967.12
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -32.663     5.377  -6.074 1.26e-09 ***
## Online       19.649     4.685   4.194 2.75e-05 ***
## AgeAverage   25.603     1.086  23.582 < 2e-16 ***
## AgeGiven     51.740     3.448  15.006 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 269.5 on 31630 degrees of freedom
## Multiple R-squared:  0.02426,   Adjusted R-squared:  0.02417
## F-statistic: 262.1 on 3 and 31630 DF,  p-value: < 2.2e-16

```

Osserviamo che in entrambi i casi, tutte le variabili risultano significative perché il p-value corrispondente è sufficientemente piccolo. Inoltre il valore della variabile Online è lo stesso del caso precedente anche se abbiamo scelto di tappare i buchi scegliendo dei valori arbitrari. Tuttavia il valore dell' $R^2$  risulta essere molto molto piccolo, quindi questo modello povero arriva a spiegare circa il 2.5% della variabilità.

Per cercare un modello più valido, possiamo considerare un'altra variabile, ad esempio il reddito di ciascun cliente. Anche in questo caso possiamo scegliere come includere i valori NA, sceglieremo ad esempio i valori nulli.

```

IncomeZero = ifelse(is.na(Income), 0, Income)
IncomeGiven = ifelse(is.na(Income), 0, 1)
mod = lm(Profit ~ Online+AgeAverage+AgeGiven+IncomeZero+IncomeGiven)
summary(mod)

```

```

##
## Call:
## lm(formula = Profit ~ Online + AgeAverage + AgeGiven + IncomeZero +
##     IncomeGiven)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -459.03 -144.61  -74.61   50.17 1963.39
##
## Coefficients:

```

```

##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -38.191     5.336  -7.158 8.38e-13 ***
## Online       11.867     4.650   2.552   0.0107 *
## AgeAverage   26.891     1.078  24.941 < 2e-16 ***
## AgeGiven     14.490     8.272   1.752   0.0799 .
## IncomeZero   18.771     0.748  25.094 < 2e-16 ***
## IncomeGiven  -63.553     9.047  -7.024 2.20e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 266.8 on 31628 degrees of freedom
## Multiple R-squared:  0.04365,    Adjusted R-squared:  0.0435
## F-statistic: 288.7 on 5 and 31628 DF,  p-value: < 2.2e-16

```

Possiamo notare, che il valore della variabile Online scende un po' che l' $R^2$  è quasi raddoppiato, ma rimane comunque piccolo.

Notiamo che non ci sono valori mancanti nella variabile District, tuttavia viene considerata numerica, quindi vorremmo trasformarla in categorica, in modo che ogni valore venga associato ad uno dei tre distretti. Quello che possiamo fare è introdurre due variabili binarie che rappresentano i distretti. Notiamo inoltre che non ci sono valori mancanti neanche nella variabile Tenure.

```

# Control for Tenure and district
any(is.na(District))

## [1] FALSE
table(District)

## District
## 1100 1200 1300
## 3142 24342 4150

District1100 = ifelse(District==1100,1,0)
District1200 = ifelse(District==1200,1,0)
any(is.na(Tenure))

```

```
## [1] FALSE
```

Andando ad effettuare la regressione con tutte queste variabili, notiamo che l' $R^2$  (e anche l' $R^2_{adj}$ ) sta aumentando anche se sempre di poco, mentre il contributo della variabile Online adesso vale circa 13.

```
mod = lm(Profit ~ Online+AgeAverage+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200)
summary(mod)
```

```

##
## Call:
## lm(formula = Profit ~ Online + AgeAverage + AgeGiven + IncomeZero +
##     IncomeGiven + Tenure + District1100 + District1200)
##
## Residuals:
##      Min      1Q      Median      3Q      Max 
## -487.17 -141.21   -65.88    48.87 1993.27 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -44.2382    6.5180  -6.787 1.16e-11 ***
## Online       13.8233    4.6091   2.999  0.00271 ** 
## AgeAverage   16.6701    1.1482  14.519 < 2e-16 ***

```

```

## AgeGiven      4.3913    8.2017   0.535  0.59237
## IncomeZero    16.8530   0.7554  22.310 < 2e-16 ***
## IncomeGiven   -57.1191   8.9956  -6.350 2.19e-10 ***
## Tenure        4.7464    0.1918  24.742 < 2e-16 ***
## District1100 -7.9955   6.2582  -1.278  0.20140
## District1200  13.1986   4.4734   2.950  0.00318 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 264.2 on 31625 degrees of freedom
## Multiple R-squared:  0.06234,   Adjusted R-squared:  0.0621
## F-statistic: 262.8 on 8 and 31625 DF,  p-value: < 2.2e-16

#potremmo considerare Age come categorico
AgeCat = ifelse(is.na(Age)==TRUE,0,age)
Age1=as.factor(AgeCat)
levels(Age1)

## [1] "0" "1" "2" "3" "4" "5" "6" "7"

mod = lm(Profit ~ Online+Age1+IncomeZero+IncomeGiven+Tenure+District1100+District1200)
summary(mod)

##
## Call:
## lm(formula = Profit ~ Online + Age1 + IncomeZero + IncomeGiven +
##     Tenure + District1100 + District1200)
##
## Residuals:
##       Min     1Q     Median     3Q     Max 
## -498.67 -141.61  -65.93   48.75 1993.55 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 22.8351    5.0941   4.483 7.40e-06 ***
## Online      13.8916    4.6093   3.014 0.002582 **  
## Age11      -63.1264   12.3213  -5.123 3.02e-07 ***
## Age12      -34.6976   9.0630  -3.829 0.000129 *** 
## Age13       1.5890    8.8141   0.180 0.856937    
## Age14       4.4084    8.9231   0.494 0.621280    
## Age15       7.3930    9.4162   0.785 0.432382    
## Age16      26.5698    9.8813   2.689 0.007173 **  
## Age17      64.8400    9.6484   6.720 1.84e-11 ***
## IncomeZero  16.7425    0.7762  21.570 < 2e-16 ***
## IncomeGiven -57.4804   9.0048  -6.383 1.76e-10 ***
## Tenure      4.7925    0.1920  24.965 < 2e-16 ***
## District1100 -7.9082   6.2555  -1.264 0.206171    
## District1200 13.2550   4.4722   2.964 0.003040 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 264.1 on 31620 degrees of freedom
## Multiple R-squared:  0.06342,   Adjusted R-squared:  0.06304
## F-statistic: 164.7 on 13 and 31620 DF,  p-value: < 2.2e-16

```

#potremmo considerare il rapporto tra Online e Age

## PARTE B

In conclusione, i modelli trovati sono poco utili nella previsione del profitto. Possiamo provare a migliorare la situazione, utilizzando la storia della banca, e quindi anche le informazioni passate dei clienti.

```
#Andiamo a rinominare le variabili  
Profit9=X9Profit  
Online9=X9Online  
Profit0=X0Profit  
Online0=X0Online
```

### Modello base

Proviamo a prevedere il profitto di ciascun cliente nel 2000, andando ad utilizzare l'informazione binaria riguardo all'uso dell'online banking. L' $R^2$  è troppo basso, quindi vorremmo migliorare questo modello.

```
mod1 = lm(Profit0 ~ Online9)  
summary(mod1)
```

```
##  
## Call:  
## lm(formula = Profit0 ~ Online9)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -5784.9  -172.9  -120.9    62.1 26944.1  
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 141.900     2.565 55.319 < 2e-16 ***  
## Online9      23.490     7.267  3.232 0.00123 **  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 389.9 on 26394 degrees of freedom  
##   (5238 observations deleted due to missingness)  
## Multiple R-squared:  0.0003957, Adjusted R-squared:  0.0003578  
## F-statistic: 10.45 on 1 and 26394 DF, p-value: 0.001229
```

Per provare a far aumentare il valore dell' $R^2$ , possiamo utilizzare le informazioni anagrafiche a nostra disposizione relative al singolo cliente, quelle che abbiamo utilizzato nei modelli precedenti.

```
mod2 = lm(Profit0 ~ Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200)  
summary(mod2)
```

```
##  
## Call:  
## lm(formula = Profit0 ~ Online9 + AgeZero + AgeGiven + IncomeZero +  
##       IncomeGiven + Tenure + District1100 + District1200)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -5933.2  -168.7  -85.0    56.8 26797.3  
##
```

```

## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 58.7253   8.8328   6.649 3.02e-11 ***
## Online9     28.5864   7.2548   3.940 8.16e-05 ***
## AgeZero    13.4625   1.7582   7.657 1.97e-14 ***
## AgeGiven   -57.8432  14.3949  -4.018 5.88e-05 ***
## IncomeZero  21.5758   1.1483  18.789 < 2e-16 ***
## IncomeGiven -85.7359  14.0981  -6.081 1.21e-09 ***
## Tenure      4.7547   0.3007  15.809 < 2e-16 ***
## District1100 -13.5127 10.0338  -1.347  0.178
## District1200 10.9915   7.1485   1.538  0.124
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 383.3 on 26387 degrees of freedom
##   (5238 observations deleted due to missingness)
## Multiple R-squared:  0.03419, Adjusted R-squared:  0.0339
## F-statistic: 116.8 on 8 and 26387 DF, p-value: < 2.2e-16

```

Questo migliora abbastanza le prestazioni, perché così riusciamo a spiegare molta più della variabilità presente. Inoltre, abbiamo un'ulteriore informazione sui singoli clienti: il profitto nel 1999. Inserendo questa informazione nel modello è come se dicesse che il profitto nell'anno successivo dipende dal profitto dell'anno in corso e dalle caratteristiche del singolo cliente. Quindi in linea di principio, coloro che l'anno precedente mi hanno fatto guadagnare, continueranno a farmi guadagnare anche nell'anno successivo.

```
mod3 = lm(Profit0 ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200)
summary(mod3)
```

```

##
## Call:
## lm(formula = Profit0 ~ Profit9 + Online9 + AgeZero + AgeGiven +
##     IncomeZero + IncomeGiven + Tenure + District1100 + District1200)
##
## Residuals:
##    Min      1Q  Median      3Q     Max
## -6948.0  -72.6  -33.2   28.6 26901.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.070248   7.185594   4.881 1.06e-06 ***
## Profit9      0.825353   0.007099  116.264 < 2e-16 ***
## Online9     15.063749   5.900660   2.553 0.010689 *
## AgeZero     -0.783422   1.434981  -0.546 0.585108
## AgeGiven    -2.298837  11.715494  -0.196 0.844438
## IncomeZero   7.123508   0.942052   7.562 4.11e-14 ***
## IncomeGiven -32.355443  11.473583  -2.820 0.004806 **
## Tenure       0.922225   0.246777   3.737 0.000187 ***
## District1100 -8.012553  8.159471  -0.982 0.326112
## District1200 -1.517990  5.814085  -0.261 0.794026
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 311.7 on 26386 degrees of freedom
##   (5238 observations deleted due to missingness)
## Multiple R-squared:  0.3614, Adjusted R-squared:  0.3611

```

```
## F-statistic: 1659 on 9 and 26386 DF, p-value: < 2.2e-16
```

Adesso l' $R^2$  risulta notevolmente migliorato: riusciamo a spiegare oltre il 36% della variabilità.

Possiamo anche supporre che i dati anagrafici non siano rilevanti, ma magari non li abbiamo utilizzati nel modo giusto nel modello. Infatti provando a tenere la variabile del profitto precedente come regressore, ma eliminando le variabili `Age` e `Income`, otteniamo un modello che spiega all'incirca la stessa variabilità.

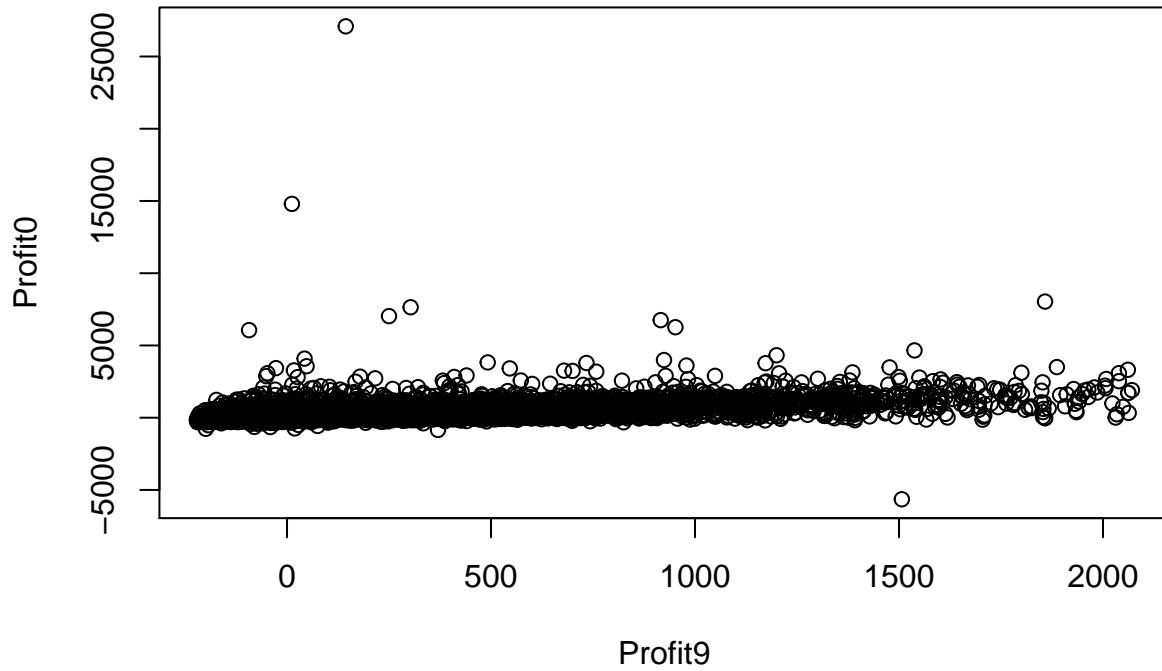
```
mod4 = lm(Profit0 ~ Profit9+Online9+Tenure+District1100+District1200)
summary(mod4)
```

```
##
## Call:
## lm(formula = Profit0 ~ Profit9 + Online9 + Tenure + District1100 +
##     District1200)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -6965.3    -70.3   -35.7    27.6  26908.6 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 31.023428  5.932798  5.229 1.72e-07 ***
## Profit9      0.831875  0.007017 118.547 < 2e-16 ***
## Online9      18.774205  5.841623  3.214  0.00131 ** 
## Tenure       0.919726  0.228984  4.017 5.92e-05 *** 
## District1100 -11.205954  8.153961 -1.374  0.16936  
## District1200  3.889211  5.776169  0.673  0.50075  
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 312 on 26390 degrees of freedom
## (5238 observations deleted due to missingness)
## Multiple R-squared:  0.3599, Adjusted R-squared:  0.3598 
## F-statistic: 2968 on 5 and 26390 DF, p-value: < 2.2e-16
```

## Profitto nel tempo

Quello che possiamo osservare è come le due variabili legate al profitto del singolo cliente nei diversi anni, siano correlate tra loro. Osserviamo infatti che c'è una correlazione di circa il 60%. Sono presenti inoltre solamente 8 osservazioni in cui il profitto nel 2000 risulta essere maggiore di 5000, quindi possiamo pensare che quelle osservazioni siano degli outliers.

```
plot(Profit9,Profit0)
```



```
cor(Profit9,Profit0,use="complete.obs")
```

```
## [1] 0.5993369
```

```
sum(Profit0>5000,na.rm=TRUE)
```

```
## [1] 8
```

Quello che possiamo fare è andare ad osservare il valore dell' $R^2$  nel caso in cui questi valori vengano eliminati, per capire se il modello risulta migliore.

```
c=which(Profit0>5000)
detach(Pilgrim)
data=Pilgrim[-c,]
attach(data)
Out_Profit9=X9Profit
Out_Online9=X9Online
Out_Age=X9Age
Out_Income=X9Inc
Out_Tenure=X9Tenure
Out_District=X9District
Out_Profit0=X0Profit
Out_Online0=X0Online
Out_District1100 = ifelse(Out_District==1100,1,0)
Out_District1200 = ifelse(Out_District==1200,1,0)
Out_AgeGiven = ifelse(is.na(Out_Age),0,1)
Out_AgeZero = ifelse(is.na(Out_Age),0,Out_Age)
Out_IncomeZero = ifelse(is.na(Out_Income),0,Out_Income)
```

```

Out_IncomeGiven = ifelse(is.na(Out_Income), 0, 1)
mod3 = lm(Out_Profit0 ~ Out_Profit9+Out_Online9+Out_AgeZero+Out_AgeGiven+Out_IncomeZero+Out_IncomeGiven
summary(mod3)

##
## Call:
## lm(formula = Out_Profit0 ~ Out_Profit9 + Out_Online9 + Out_AgeZero +
##     Out_AgeGiven + Out_IncomeZero + Out_IncomeGiven + Out_Tenure +
##     Out_District1100 + Out_District1200)
##
## Residuals:
##    Min      1Q  Median      3Q      Max
## -6931.7   -70.0   -31.5    31.0   4011.0
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.572456  5.271173  6.748 1.52e-11 ***
## Out_Profit9  0.817345  0.005213 156.793 < 2e-16 ***
## Out_Online9 18.170277  4.328440  4.198 2.70e-05 ***
## Out_AgeZero -1.807824  1.052745 -1.717 0.085946 .
## Out_AgeGiven 2.195794  8.593913  0.256 0.798335
## Out_IncomeZero 6.398557  0.691090  9.259 < 2e-16 ***
## Out_IncomeGiven -28.027989  8.416348 -3.330 0.000869 ***
## Out_Tenure    0.838592  0.181070  4.631 3.65e-06 ***
## Out_District1100 -10.954157  5.985906 -1.830 0.067262 .
## Out_District1200 -4.504767  4.264877 -1.056 0.290865
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 228.7 on 26378 degrees of freedom
## (5238 observations deleted due to missingness)
## Multiple R-squared:  0.506, Adjusted R-squared:  0.5058
## F-statistic:  3002 on 9 and 26378 DF, p-value: < 2.2e-16

mod4 = lm(Out_Profit0 ~ Out_Profit9+Out_Online9+Out_Tenure+Out_District1100+Out_District1200)
summary(mod4)

##
## Call:
## lm(formula = Out_Profit0 ~ Out_Profit9 + Out_Online9 + Out_Tenure +
##     Out_District1100 + Out_District1200)
##
## Residuals:
##    Min      1Q  Median      3Q      Max
## -6947.8   -68.1   -33.9    30.3   4009.2
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.233833  4.354890  7.631 2.40e-14 ***
## Out_Profit9  0.822841  0.005156 159.577 < 2e-16 ***
## Out_Online9 22.135589  4.287856  5.162 2.46e-07 ***
## Out_Tenure   0.770985  0.168128  4.586 4.55e-06 ***
## Out_District1100 -14.051521  5.985672 -2.348  0.0189 *
## Out_District1200  0.403304  4.239795  0.095  0.9242

```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 229 on 26382 degrees of freedom
##   (5238 observations deleted due to missingness)
## Multiple R-squared: 0.5043, Adjusted R-squared: 0.5042
## F-statistic: 5367 on 5 and 26382 DF, p-value: < 2.2e-16
detach(data)
attach(Pilgrim)

```

Quindi andando a ripetere il codice precedente, quando però eliminiamo gli outliers, miglioriamo l' $R^2$  che negli ultimi due modelli raggiunge circa il 50%.

Osserviamo inoltre che abbiamo un numero diverso di valori mancanti: ci sono 5219 osservazioni in cui non abbiamo l'informazione sulla online banking nel 2000 e 5238 osservazioni in cui non abbiamo l'informazione sul profitto nel 2000, questo vuol dire che il cliente non è rimasto nella banca nell'anno successivo. Quindi logicamente, non ci sono osservazioni in cui abbiamo l'informazione di Profit0 ma non di Online0. Ci sono invece 19 osservazioni in cui al contrario abbiamo l'informazione sul profitto, ma non sappiamo se il cliente ha utilizzato la banca online.

```

sum(is.na(Online0))

## [1] 5219
sum(is.na(Prom0))

## [1] 5238
which(is.na(Prom0) != is.na(Online0))

## [1] 2309 4128 4191 5131 5683 8612 9310 11030 11780 12737 18735
## [12] 21065 23438 25446 26315 27981 30219 30961 31544
which(is.na(Prom0) < is.na(Online0))

## integer(0)
which(is.na(Prom0) > is.na(Online0))

## [1] 2309 4128 4191 5131 5683 8612 9310 11030 11780 12737 18735
## [12] 21065 23438 25446 26315 27981 30219 30961 31544

```

### ADD retain variable

Quello che possiamo fare è aggiungere una variabile aggiuntiva **Retain** che ci dice se il cliente rimane nella banca anche nell'anno successivo. A questo punto vorremmo capire come prevedere al meglio se un cliente rimane con la banca nell'anno successivo oppure no, questo potrebbe dipendere da tutte le caratteristiche del cliente.

```

Retain = ifelse(is.na(Prom0), 0, 1)
mod1 = lm(Retain ~ Online9)
summary(mod1)

##
## Call:
## lm(formula = Retain ~ Online9)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5000 -0.2500  0.0000  0.2500  0.5000
## 
```

```

## -0.8534  0.1466  0.1682  0.1682  0.1682
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.831785   0.002230 373.031 < 2e-16 ***
## Online9      0.021614   0.006388   3.383 0.000717 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3716 on 31632 degrees of freedom
## Multiple R-squared:  0.0003617, Adjusted R-squared:  0.0003301
## F-statistic: 11.45 on 1 and 31632 DF, p-value: 0.0007171
mod2 = lm(Retain ~ Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200)
summary(mod2)

##
## Call:
## lm(formula = Retain ~ Online9 + AgeZero + AgeGiven + IncomeZero +
##     IncomeGiven + Tenure + District1100 + District1200)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.04743  0.02556  0.07967  0.10976  0.46678
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.5356666  0.0065081 82.308 < 2e-16 ***
## Online9      0.0107639  0.0058863   1.829 0.067464 .
## AgeZero      0.0014025  0.0014664   0.956 0.338848
## AgeGiven     0.1664412  0.0117295  14.190 < 2e-16 ***
## IncomeZero   0.0035164  0.0009647   3.645 0.000268 ***
## IncomeGiven  0.1501529  0.0114885  13.070 < 2e-16 ***
## Tenure        0.0039224  0.0002450  16.010 < 2e-16 ***
## District1100 -0.0030727  0.0079925  -0.384 0.700645
## District1200  0.0074633  0.0057131   1.306 0.191448
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3375 on 31625 degrees of freedom
## Multiple R-squared:  0.176, Adjusted R-squared:  0.1758
## F-statistic: 844.5 on 8 and 31625 DF, p-value: < 2.2e-16
mod3 = lm(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200)
summary(mod3)

##
## Call:
## lm(formula = Retain ~ Profit9 + Online9 + AgeZero + AgeGiven +
##     IncomeZero + IncomeGiven + Tenure + District1100 + District1200)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0580  0.0253  0.0793  0.1102  0.4675
##

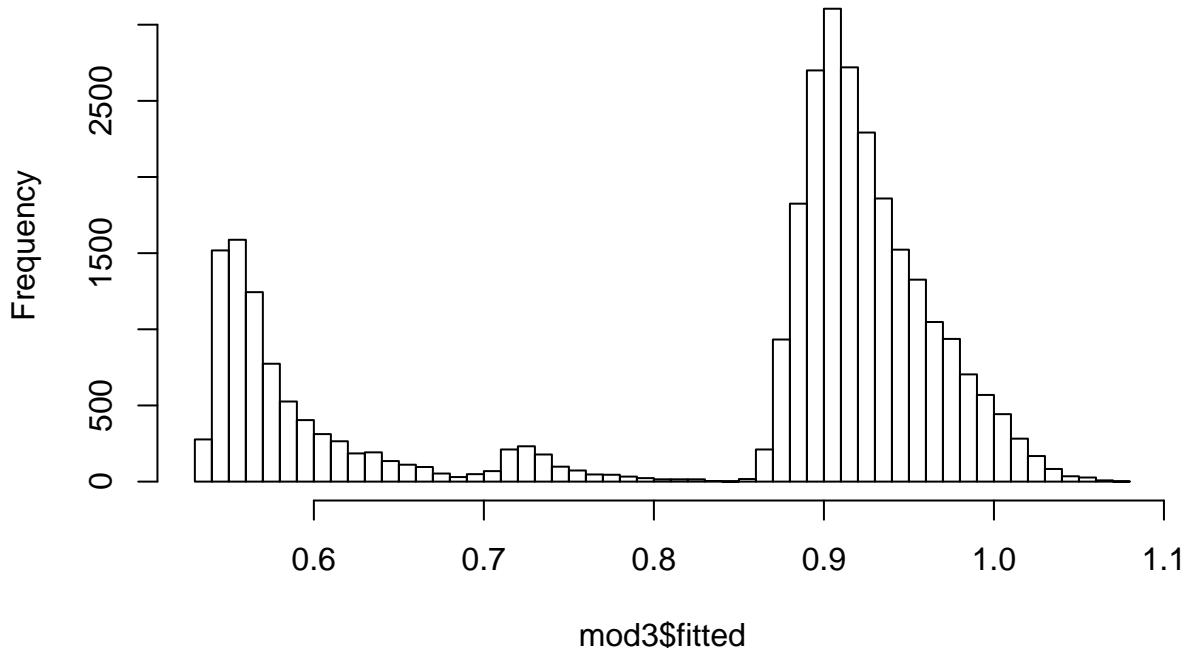
```

```

## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.352e-01  6.510e-03 82.221 < 2e-16 ***
## Profit9        1.896e-05  7.181e-06  2.640  0.00829 **
## Online9         1.050e-02  5.887e-03  1.784  0.07443 .
## AgeZero        1.086e-03  1.471e-03  0.739  0.46020
## AgeGiven        1.676e-01  1.174e-02 14.283 < 2e-16 ***
## IncomeZero     3.197e-03  9.722e-04  3.288  0.00101 **
## IncomeGiven    1.512e-01  1.149e-02 13.157 < 2e-16 ***
## Tenure          3.832e-03  2.473e-04 15.495 < 2e-16 ***
## District1100 -2.921e-03  7.992e-03 -0.366  0.71473
## District1200  7.213e-03  5.713e-03  1.262  0.20678
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3374 on 31624 degrees of freedom
## Multiple R-squared:  0.1762, Adjusted R-squared:  0.176
## F-statistic: 751.6 on 9 and 31624 DF,  p-value: < 2.2e-16
hist(mod3$fitted, nclass=50, main ="Histogram of mod3")

```

**Histogram of mod3**



Andare a considerare il valore dell' $R^2$  per valutare la bontà del modello non è molto opportuno perché stiamo prevedendo se il cliente rimane oppure no, quindi è una variabile binaria. Dall'istogramma possiamo facilmente dedurre ci sono due gruppi abbastanza distinti, quindi quello che possiamo fare è fissare una certa soglia al di sopra della quale possiamo concludere che il cliente è sicuro, mentre al di sotto si trovano quei clienti che probabilmente il prossimo anno cambieranno la banca. Infatti ci sono due mode molto alte e sufficientemente lontane, mentre tra i valori di 0.7 e 0.8, troviamo una "zona grigia", dove la moda è molto

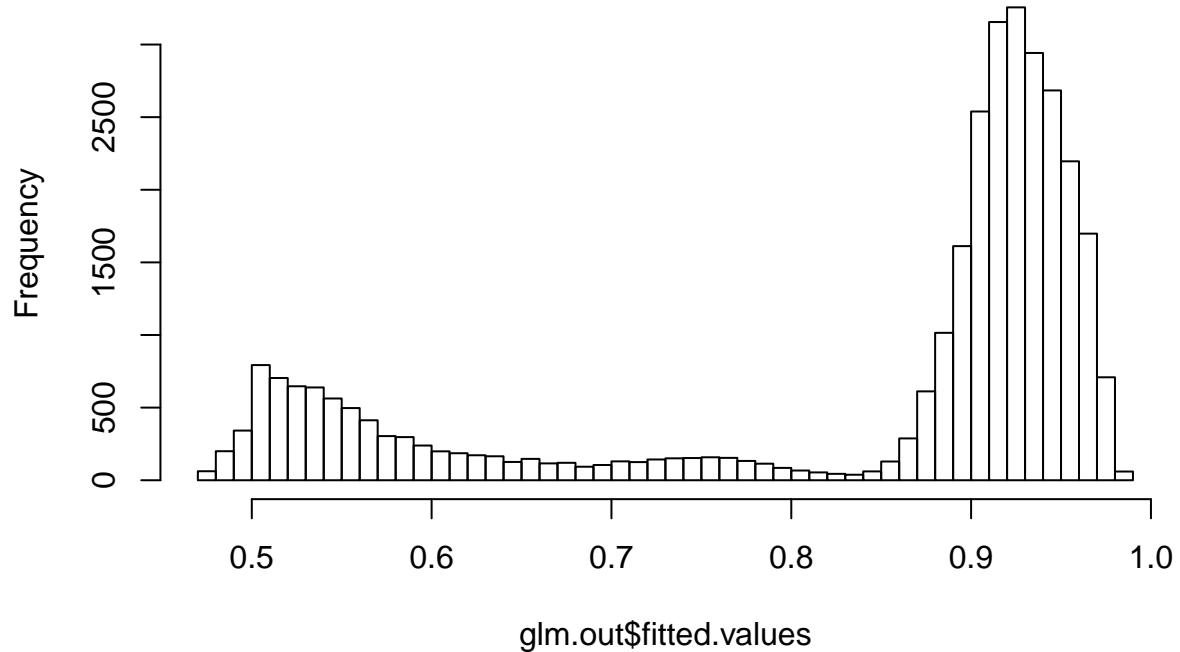
bassa e i clienti sono abbastanza incerti. Inoltre saremmo interessati ad interpretare i valori dell'istogramma come delle probabilità, anche se ci sono dei valori più grandi di 1. In questo modo sarei in grado di capire sotto quale valore dovrei preoccuparmi del cliente.

#— PART 3 - ANALYZE retention with Logistic regression Proviamo a prevedere il comportamento futuro di un cliente, utilizzando la regressione logistica.

```
glm.out = glm(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+
  IncomeGiven+Tenure+District1100+District1200,family=binomial(logit))
summary(glm.out)

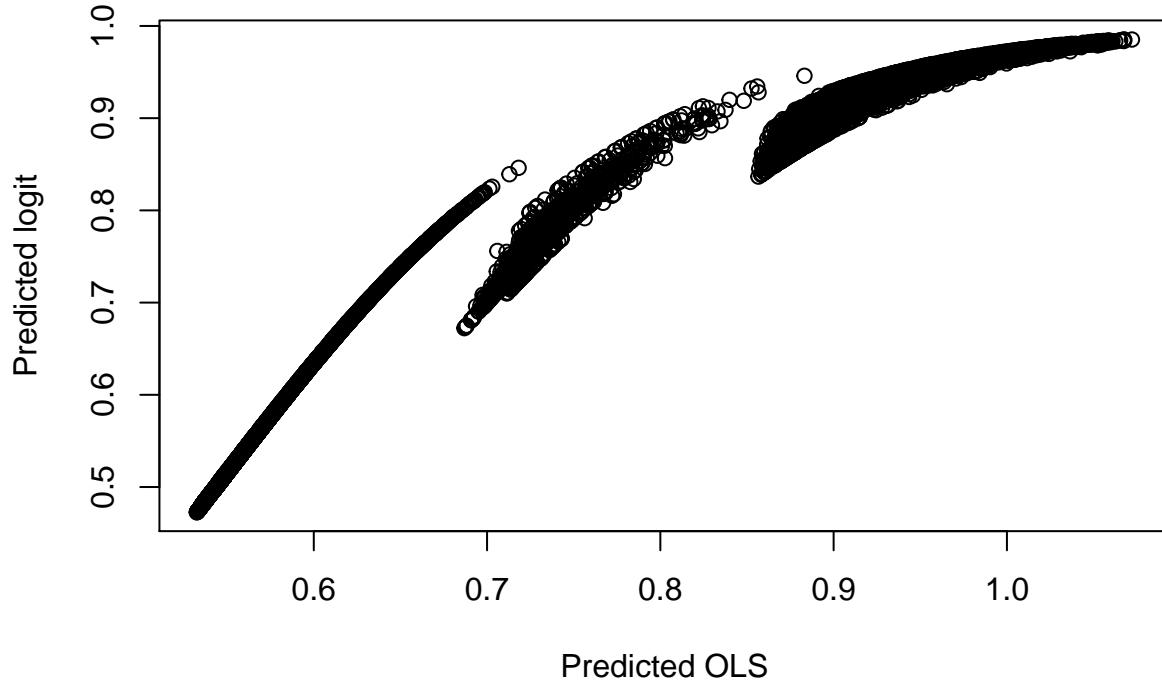
##
## Call:
## glm(formula = Retain ~ Profit9 + Online9 + AgeZero + AgeGiven +
##       IncomeZero + IncomeGiven + Tenure + District1100 + District1200,
##       family = binomial(logit))
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max 
## -2.8383   0.2916   0.3883   0.4667   1.2242 
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -7.985e-02 5.145e-02 -1.552   0.1207    
## Profit9      1.841e-04 7.179e-05  2.564   0.0104 *  
## Online9      1.056e-01 5.333e-02  1.979   0.0478 *  
## AgeZero      6.845e-02 1.596e-02  4.290  1.79e-05 *** 
## AgeGiven     8.371e-01 9.307e-02  8.994 < 2e-16 *** 
## IncomeZero   5.325e-02 1.058e-02  5.035  4.78e-07 *** 
## IncomeGiven  7.750e-01 8.999e-02  8.612 < 2e-16 *** 
## Tenure       3.748e-02 2.413e-03 15.532 < 2e-16 *** 
## District1100 -3.174e-02 6.798e-02 -0.467   0.6406    
## District1200  6.700e-02 4.915e-02  1.363   0.1729    
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 28395  on 31633  degrees of freedom
## Residual deviance: 23299  on 31624  degrees of freedom
## AIC: 23319
##
## Number of Fisher Scoring iterations: 5
hist(glm.out$fitted.values,nclass=50, main="Histogram of Logistic Regression")
```

## Histogram of Logistic Regression



Questo metodo risulta molto veloce,in quanto si ferma dopo solo 5 iterazioni. Inoltre le diaognostiche del modello sono un po' diverse, notiamo che non abbiamo più a disposizione l' $R^2$ ,ma non cambia il senso generale, come osserviamo dall'istogramma.

```
plot(mod3$fitted, glm.out$fitted, xlab="Predicted OLS", ylab="Predicted logit")
```



```
# Regressione logistica andando ad eliminare gli outliers
glm.out = glm(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+
  IncomeGiven+Tenure+District1100+District1200,family=binomial(logit))
summary(glm.out)
```

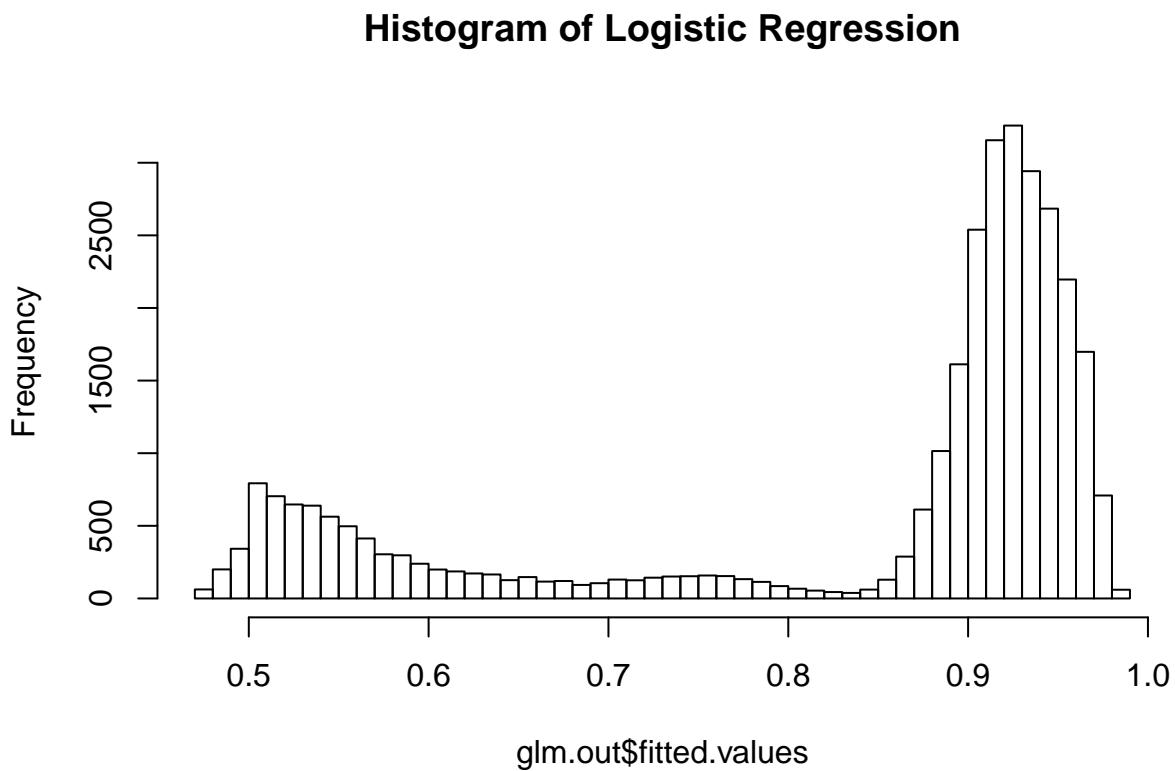
```
##
## Call:
## glm(formula = Retain ~ Profit9 + Online9 + AgeZero + AgeGiven +
##       IncomeZero + IncomeGiven + Tenure + District1100 + District1200,
##       family = binomial(logit))
##
## Deviance Residuals:
##    Min      1Q   Median      3Q     Max
## -2.8383  0.2916  0.3883  0.4667  1.2242
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.985e-02 5.145e-02 -1.552  0.1207
## Profit9      1.841e-04 7.179e-05  2.564  0.0104 *
## Online9      1.056e-01 5.333e-02  1.979  0.0478 *
## AgeZero      6.845e-02 1.596e-02  4.290  1.79e-05 ***
## AgeGiven     8.371e-01 9.307e-02  8.994 < 2e-16 ***
## IncomeZero   5.325e-02 1.058e-02  5.035  4.78e-07 ***
## IncomeGiven  7.750e-01 8.999e-02  8.612 < 2e-16 ***
## Tenure       3.748e-02 2.413e-03 15.532 < 2e-16 ***
## District1100 -3.174e-02 6.798e-02 -0.467  0.6406
```

```

## District1200 6.700e-02 4.915e-02 1.363 0.1729
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 28395 on 31633 degrees of freedom
## Residual deviance: 23299 on 31624 degrees of freedom
## AIC: 23319
##
## Number of Fisher Scoring iterations: 5

hist(glm.out$fitted.values,nclass=50, main="Histogram of Logistic Regression")

```



#— PART 4 - Demographics vs. Past profit to analyze profitability Possiamo considerare le variabili Age e Income effettivamente come categoriche.

```
mod1 = lm(Profit0 ~ Profit9+Online9+Tenure+District1100+District1200+factor(Age)+factor(Income))  
summary(mod1)
```

```
##  
## Call:  
## lm(formula = Profit0 ~ Profit9 + Online9 + Tenure + District1100 +  
##       District1200 + factor(Age) + factor(Income))  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -100000 -100000 -100000 -100000 -100000
```

```

## -6958.2   -75.5   -31.1    33.4 26898.0
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)           14.823899  16.171007  0.917 0.359312
## Profit9              0.831492   0.007838 106.086 < 2e-16 ***
## Online9               7.319690   6.578991  1.113 0.265899
## Tenure                0.714656   0.280908  2.544 0.010963 *
## District1100        -7.042084   9.293054 -0.758 0.448592
## District1200        -0.923487   6.598269 -0.140 0.888694
## factor(Age)2          4.361207  15.035530  0.290 0.771773
## factor(Age)3          17.812641  14.754618  1.207 0.227346
## factor(Age)4          6.523992  14.846707  0.439 0.660359
## factor(Age)5          -6.767698  15.364702 -0.440 0.659601
## factor(Age)6          3.608215  15.885365  0.227 0.820316
## factor(Age)7          11.647038  15.692484  0.742 0.457971
## factor(Income)2       -17.471886  14.026549 -1.246 0.212914
## factor(Income)3       -0.028443  10.128287 -0.003 0.997759
## factor(Income)4        6.478633  10.291317  0.630 0.529013
## factor(Income)5       -2.561715  10.264249 -0.250 0.802917
## factor(Income)6        16.029132  8.991267  1.783 0.074642 .
## factor(Income)7        25.047181  9.819904  2.551 0.010759 *
## factor(Income)8        42.635779  11.216004  3.801 0.000144 ***
## factor(Income)9        51.513185  10.088420  5.106 3.32e-07 ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 313.8 on 21063 degrees of freedom
##   (10551 observations deleted due to missingness)
## Multiple R-squared:  0.3737, Adjusted R-squared:  0.3731
## F-statistic: 661.4 on 19 and 21063 DF,  p-value: < 2.2e-16
mod2 = lm(Profit0 ~ Profit9+Online9+Tenure)
summary(mod2)

```

```

##
## Call:
## lm(formula = Profit0 ~ Profit9 + Online9 + Tenure)
##
## Residuals:
##      Min      1Q      Median      3Q      Max
## -6967.7   -70.0   -35.9    27.5 26910.9
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)            32.956289   3.230813 10.201 < 2e-16 ***
## Profit9                 0.832647   0.007008 118.810 < 2e-16 ***
## Online9                19.438251   5.833628  3.332 0.000863 ***
## Tenure                  0.902109   0.228864  3.942 8.11e-05 ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 312.1 on 26392 degrees of freedom
##   (5238 observations deleted due to missingness)
## Multiple R-squared:  0.3598, Adjusted R-squared:  0.3597

```

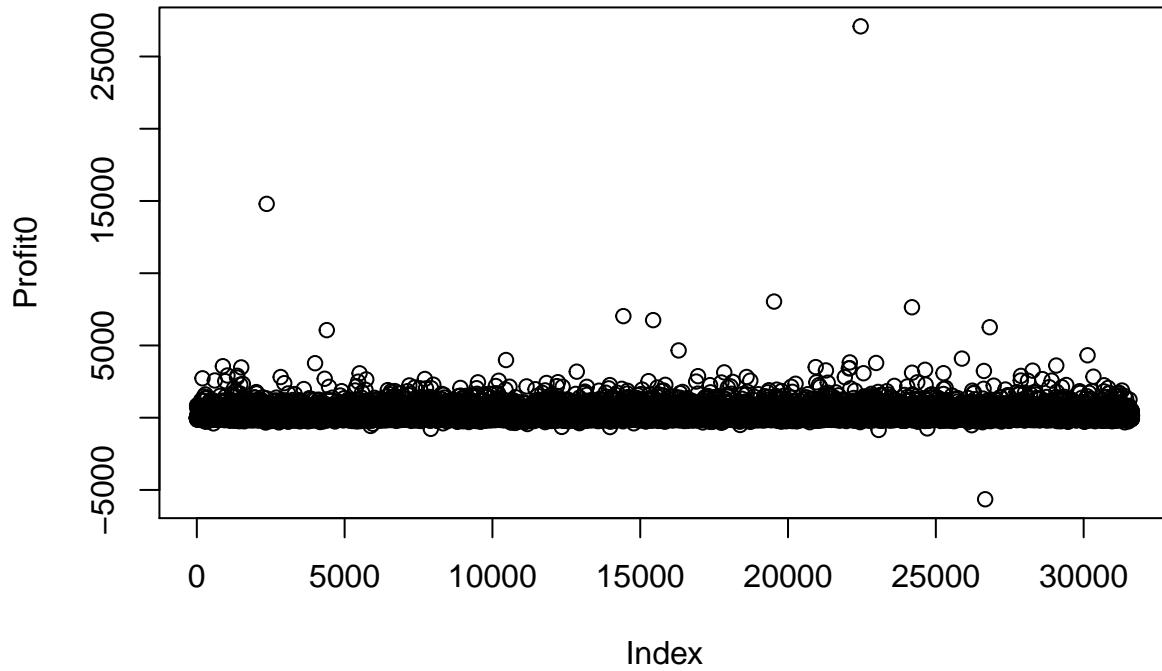
```

## F-statistic: 4944 on 3 and 26392 DF, p-value: < 2.2e-16
mod3 = lm(Profit0 ~ District1100+District1200+factor(Age)+factor(Income))
summary(mod3)

##
## Call:
## lm(formula = Profit0 ~ District1100 + District1200 + factor(Age) +
##     factor(Income))
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -5828.8  -177.2   -89.5    70.4  26828.9 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -6.195     20.010  -0.310  0.7569    
## District1100 -13.146    11.550  -1.138  0.2551    
## District1200   8.852     8.198   1.080  0.2803    
## factor(Age)2  30.488    18.683   1.632  0.1027    
## factor(Age)3  84.514    18.287   4.622 3.83e-06 *** 
## factor(Age)4  89.375    18.309   4.881 1.06e-06 *** 
## factor(Age)5  88.983    18.826   4.727 2.30e-06 *** 
## factor(Age)6 125.998    19.317   6.523 7.06e-11 *** 
## factor(Age)7 167.784    18.989   8.836 < 2e-16 *** 
## factor(Income)2 -9.506    17.433  -0.545  0.5856    
## factor(Income)3 12.698    12.584   1.009  0.3130    
## factor(Income)4 24.172    12.776   1.892  0.0585 .  
## factor(Income)5 17.473    12.742   1.371  0.1703    
## factor(Income)6 59.009    11.150   5.292 1.22e-07 *** 
## factor(Income)7 86.681    12.165   7.125 1.07e-12 *** 
## factor(Income)8 120.626   13.894   8.682 < 2e-16 *** 
## factor(Income)9 189.270   12.403   15.261 < 2e-16 *** 
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
##
## Residual standard error: 390 on 21066 degrees of freedom
##   (10551 observations deleted due to missingness)
## Multiple R-squared:  0.03224,   Adjusted R-squared:  0.0315 
## F-statistic: 43.86 on 16 and 21066 DF, p-value: < 2.2e-16
summary(Profit0)

##      Min. 1st Qu. Median Mean 3rd Qu. Max. NA's 
## -5643.0 -30.0  23.0 144.8 206.0 27086.0 5238
plot(Profit0)

```



```

m=which(Profit0<=-5000)
M=which(Profit0>=5000)
d=Pilgrim[-m,]
dtab=d[-M, ]
summary(dtab)

##          ID           X9Profit        X9Online        X9Age
##  Min.    : 1   Min.   :-221.0   Min.   :0.0000   Min.   :1.000
##  1st Qu.: 7909  1st Qu.: -34.0   1st Qu.:0.0000   1st Qu.:3.000
##  Median  :15817  Median  :  9.0   Median  :0.0000   Median  :4.000
##  Mean    :15817  Mean    :111.4   Mean    :0.1219   Mean    :4.046
##  3rd Qu.:23725  3rd Qu.:164.0   3rd Qu.:0.0000   3rd Qu.:5.000
##  Max.    :31634  Max.    :2071.0  Max.    :1.0000   Max.    :7.000
##                               NA's    :8288
##          X9Inc         X9Tenure       X9District      X0Profit
##  Min.    :1.000   Min.    : 0.16   Min.    :1100   Min.   :-846.0
##  1st Qu.:4.000   1st Qu.: 3.75   1st Qu.:1200   1st Qu.: -30.0
##  Median  :6.000   Median  : 7.41   Median  :1200   Median  : 23.0
##  Mean    :5.458   Mean    :10.16   Mean    :1203   Mean    :142.2
##  3rd Qu.:7.000   3rd Qu.:14.75   3rd Qu.:1200   3rd Qu.:206.0
##  Max.    :9.000   Max.    :41.16   Max.    :1300   Max.    :6263.0
##  NA's    :8260    NA's    :5238
##          X0Online      X9Billpay      X0Billpay
##  Min.    :0.000   Min.    :0.0000   Min.   :0.00
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.00
##  Median  :0.000   Median  :0.0000   Median  :0.00

```

```
##  Mean    :0.199   Mean    :0.0167   Mean    :0.03
##  3rd Qu.:0.000   3rd Qu.:0.0000   3rd Qu.:0.00
##  Max.    :1.000   Max.    :1.0000   Max.    :1.00
##  NA's    :5219
```

```

{r decision tree} # library(tree) # h.tree<-tree(as.factor(Retain)
~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+ #    IncomeGiven+Tenure+Di
# summary(h.tree) #

{r decision tree2} # plot(h.tree,lwd=3) # text(h.tree,pretty=0,cex=1.2,c
# h.tree # setup<-tree.control(length(Retain), mincut = 2,
minsize = 6, mindev = 0.0005) # h.tree<-tree(as.factor(Retain)
~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+ #    IncomeGiven+Tenure+Di
# summary(h.tree) # plot(h.tree,lwd=3) # text(h.tree,pretty=0,cex=1.2,c
#

{r decision tree cv} # cv.hitters =cv.tree(h.tree)# ,FUN=prune.tree
) # cv.hitters # plot(cv.hitters$size,cv.hitters$dev,type="b",
lwd=3,col="blue", # xlab="Nodi terminali", ylab="RSS",main="Cost
complexity pruning" ) #

{r decision tree predict} # prune.hitters=prune.tree(h.tree,
best =10) # plot(prune.hitters,lwd=3) # text(prune.hitters
,pretty =0,cex=1.2,col="blue") # P1<-predict(prune.hitters)
# P1<-data.frame(P1) ## semplicemente per osservare meglio #
## cosa abbiamo nell'oggetto P1 # head(P1) #

{r decision tree balanced} # library(rpart) # library(ROSE) #
Retain = as.factor(Retain) # data_balanced_over <- ovun.sample(as.factor(
~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District
method = "over",N = 52792)$data # setup<-tree.control(52792,
mincut = 2, minsize = 6, mindev = 0.00003) # h.tree.balanced<-rpart(Retain
~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District
data=data_balanced_over, control=setup) # summary(h.tree.balanced)
# plot(h.tree.balanced,lwd=3) # text(h.tree.balanced,pretty=0,cex=1.2,c
# pred.treebal <- predict(h.tree.balanced, newdata = data_balanced_over)
#  # # #summary(as.factor(Retain)) # #data_balanced_over <-
ovun.sample(as.factor(Retain) ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+
# # IncomeGiven+Tenure+District1100+District1200, method =
"over",N = 5000)$data # #table(data_balanced_over$Retain) #

```

Bilanciamento dataset

```

library(ROSE)

## Warning: package 'ROSE' was built under R version 3.6.3
## Loaded ROSE 0.0-3
Retain = as.factor(Retain)
data_balanced_over <- ovun.sample(as.factor(Retain) ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200, n=52792, replace=TRUE)
## Warning in (function (formula, data, method, subset, na.action, N, p = 0.5, : Transformations of variables are needed
## New data have been generated by using non-transformed variables.
##
library(tree)
setup<-tree.control(52792, mincut = 2, minsize = 6, mindev = 0.001)
tree.Pilgrim = tree(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200, control = setup)
summary(tree.Pilgrim)

##
## Classification tree:
## tree(formula = Retain ~ Profit9 + Online9 + AgeZero + AgeGiven +
##       IncomeZero + IncomeGiven + Tenure + District1100 + District1200,
##       data = data_balanced_over, control = setup)
## Variables actually used in tree construction:
## [1] "AgeZero"      "IncomeZero"    "Tenure"        "Profit9"
## Number of terminal nodes:  10
## Residual mean deviance:  1.107 = 58440 / 52780
## Misclassification error rate: 0.2602 = 13738 / 52792
set.seed(1)
train = sample(1:nrow(data_balanced_over), nrow(data_balanced_over)*0.7)
df.train=data_balanced_over[train,]
df.test = data_balanced_over[-train,]
target.train = data_balanced_over[train,]$Retain
target.test = df.test$Retain
setup<-tree.control(52792-length(target.test),mincut = 2, minsize = 6, mindev = 0.001)
tree.Pilgrim = tree(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District1200, control = setup)
tree.pred = predict(tree.Pilgrim, df.test, type="class", probability=TRUE)
#table.res = table(tree.pred,target.test)
#table.res
#accuracy = (table.res[1,1]+table.res[2,2])/sum(table.res)
#accuracy
library(caret)

## Loading required package: lattice
## Loading required package: ggplot2
cf = confusionMatrix(data = tree.pred, reference = target.test, mode = "prec_recall")
cf

## Confusion Matrix and Statistics
##
##          Reference
## Prediction   1     0
##             1 6367 2657
##             0 1508 5306
##

```

```

##                               Accuracy : 0.737
##             95% CI : (0.7301, 0.7439)
##      No Information Rate : 0.5028
##      P-Value [Acc > NIR] : < 2.2e-16
##
##                               Kappa : 0.4745
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##                               Precision : 0.7056
##                               Recall : 0.8085
##                               F1 : 0.7535
##                               Prevalence : 0.4972
##                               Detection Rate : 0.4020
##      Detection Prevalence : 0.5698
##      Balanced Accuracy : 0.7374
##
##      'Positive' Class : 1
##
precision <- posPredValue(tree.pred,target.test, positive="1")
recall <- sensitivity(tree.pred,target.test, positive="1")

#F1 <- (2 * precision * recall) / (precision + recall)
#precision
#recall
#F1
#summary(tree.Pilgrim)

```

## Logistic Regression

```

glm.out = glm(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+
  IncomeGiven+Tenure+District1100+District1200,family=binomial(logit), data=df.train)
summary(glm.out)

##
## Call:
## glm(formula = Retain ~ Profit9 + Online9 + AgeZero + AgeGiven +
##     IncomeZero + IncomeGiven + Tenure + District1100 + District1200,
##     family = binomial(logit), data = df.train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.9441   -0.8553   -0.4541    0.7142    2.1917
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.623e+00  3.871e-02  41.917  < 2e-16 ***
## Profit9     -1.889e-04  4.859e-05  -3.887  0.000101 ***
## Online9     -2.826e-02  3.757e-02  -0.752  0.451896
## AgeZero     -6.471e-02  9.857e-03  -6.565 5.21e-11 ***
## AgeGiven    -7.856e-01  6.944e-02 -11.313  < 2e-16 ***
## IncomeZero   -5.712e-02  6.614e-03  -8.636  < 2e-16 ***

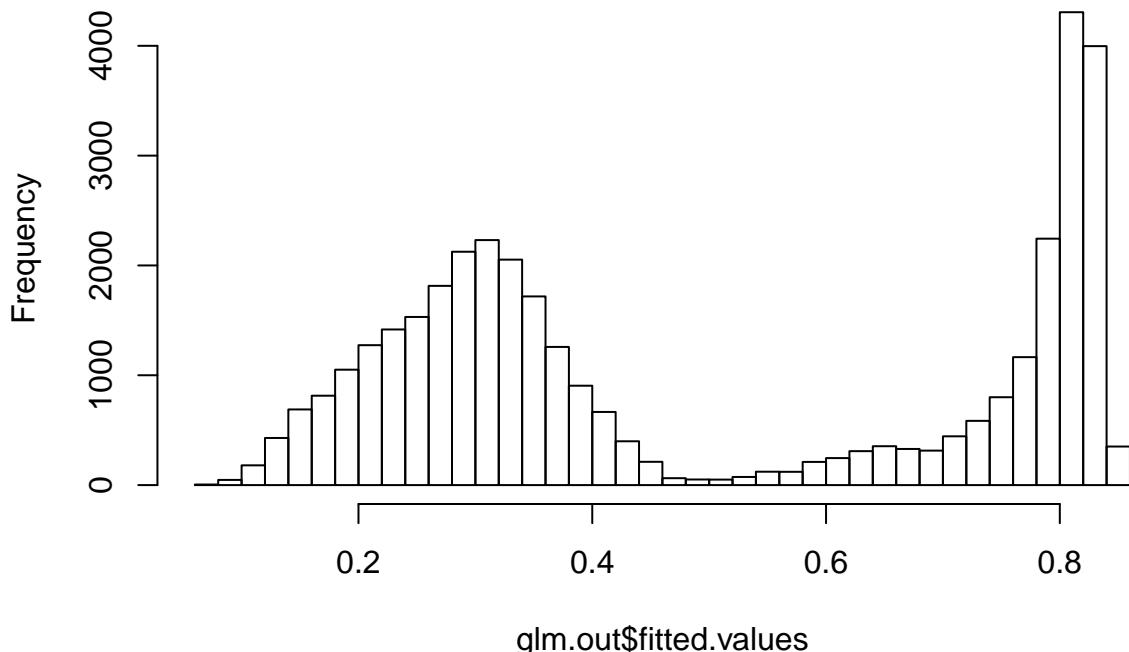
```

```

## IncomeGiven -8.133e-01 6.738e-02 -12.072 < 2e-16 ***
## Tenure      -3.548e-02 1.655e-03 -21.437 < 2e-16 ***
## District1100 1.065e-01 4.921e-02  2.164 0.030436 *
## District1200 -3.450e-02 3.561e-02 -0.969 0.332676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 51229  on 36953  degrees of freedom
## Residual deviance: 41052  on 36944  degrees of freedom
## AIC: 41072
##
## Number of Fisher Scoring iterations: 4
hist(glm.out$fitted.values,nclass=50, main="Histogram of Logistic Regression")

```

## Histogram of Logistic Regression



```

pred=predict(glm.out, df.test, type="response")
logit.pred = rep(0, dim(df.test)[1])
logit.pred[pred > .5] = 1
logit.pred=factor(logit.pred,labels=c("1","0"))
cf = confusionMatrix(data = logit.pred, reference = target.test, mode = "prec_recall", positive="1")
cf

## Confusion Matrix and Statistics
##
##          Reference

```

```

## Prediction      1      0
##                 1 6361 2690
##                 0 1514 5273
##
##                         Accuracy : 0.7346
##                         95% CI : (0.7276, 0.7414)
##     No Information Rate : 0.5028
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                         Kappa : 0.4695
##
##     Mcnemar's Test P-Value : < 2.2e-16
##
##                         Precision : 0.7028
##                         Recall : 0.8077
##                         F1 : 0.7516
##                         Prevalence : 0.4972
##                         Detection Rate : 0.4016
##     Detection Prevalence : 0.5715
##     Balanced Accuracy : 0.7350
##
##     'Positive' Class : 1
##
precision <- posPredValue(logit.pred,target.test, positive="1")
recall <- sensitivity(logit.pred,target.test, positive="1")
specificity=specificity(logit.pred,target.test, positive="1")
cat('Specificity:',specificity, '\n')

## Specificity: 0.6621876

```

## KNN

```

set.seed(1)
train.control=trainControl(method="repeatedcv", number=3, repeats=1)
fit=train(Retain ~ Profit9+Online9+AgeZero+AgeGiven+IncomeZero+IncomeGiven+Tenure+District1100+District2100,
          method="knn",
          tuneGrid=expand.grid(k=c(4,8,10,15)),
          trControl=train.control,
          metric="Accuracy",
          data=df.train)
fit

## k-Nearest Neighbors
##
## 36954 samples
##      9 predictor
##      2 classes: '1', '0'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold, repeated 1 times)
## Summary of sample sizes: 24636, 24637, 24635
## Resampling results across tuning parameters:
##
```

```

##   k   Accuracy   Kappa
##   4   0.7466583  0.4935831
##   8   0.7333442  0.4668774
##  10  0.7304758  0.4611070
##  15  0.7289605  0.4579898
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 4.

library(class)
knn_pred <- knn(df.train,df.test,cl=target.train,k=2)
cf = confusionMatrix(data = knn_pred, reference = target.test, mode = "prec_recall")
cf

## Confusion Matrix and Statistics
##
##             Reference
## Prediction    1     0
##           1 6543  270
##           0 1332 7693
##
##             Accuracy : 0.8989
##                 95% CI : (0.894, 0.9035)
## No Information Rate : 0.5028
## P-Value [Acc > NIR] : < 2.2e-16
##
##             Kappa : 0.7975
##
## Mcnemar's Test P-Value : < 2.2e-16
##
##             Precision : 0.9604
##             Recall : 0.8309
##             F1 : 0.8909
##             Prevalence : 0.4972
## Detection Rate : 0.4131
## Detection Prevalence : 0.4302
## Balanced Accuracy : 0.8985
##
## 'Positive' Class : 1
##
precision <- posPredValue(knn_pred,target.test, positive="1")
recall <- sensitivity(knn_pred,target.test, positive="1")
specificity=specificity(knn_pred,target.test, positive="1")
cat('Specificity:',specificity, '\n')

## Specificity: 0.9660932

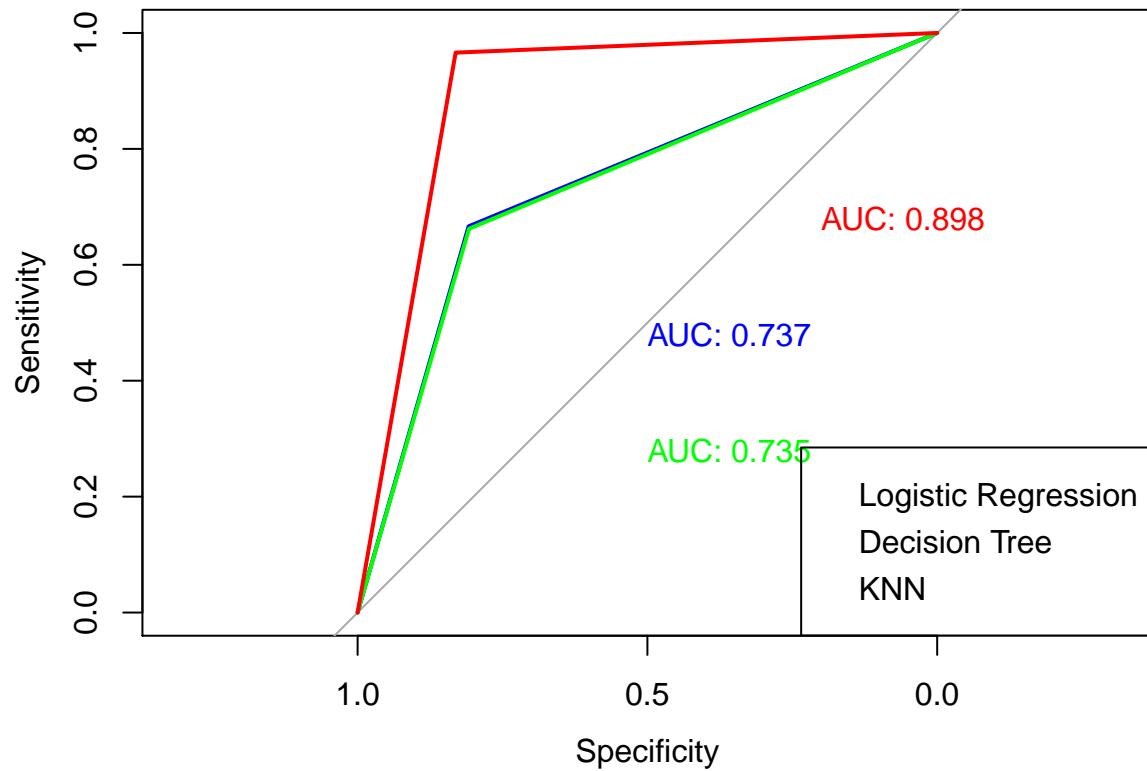
library(pROC)

## Warning: package 'pROC' was built under R version 3.6.3

treeROC = roc(target.test,factor(tree.pred, ordered = TRUE), plot=TRUE, print.auc=TRUE, col="blue")
logit_roc=plot.roc(target.test,factor(logit.pred, ordered = TRUE), add=TRUE, col="green", print.auc=TRUE)
knn_roc = plot.roc(target.test,factor(knn_pred, ordered = TRUE), add=TRUE, col="red", print.auc=TRUE, p

legend("bottomright", legend = c("Logistic Regression", "Decision Tree", "KNN"), col=c("green", "blue",

```



```
detach(Pilgrim)
```