Homeworks Business Analytics
Homework1:
Customer retention per il caso Pilgrim Bank

Enrico Agrippino (291272), Teresa Di Renzo (290523)

June 2021

Sommario

Con questo homework ci proponiamo di confrontare diversi metodi di classificazione per il caso Pilgrim Bank B. In particolare, il nostro obiettivo è prevedere se i clienti lasceranno la banca l'anno successivo, sulla base della loro età (*Age*), del loro salario (*Income*), del numero di anni da cui sono clienti (*Tenure*), del profitto che rendono alla banca (*Profit*) e della scelta di passare ai servizi online o meno (*Online*).

In fase di preparazione dei dati, abbiamo diviso il dataset in training e test set, divisione unica per tutti i metodi, in modo che il confronto sulle performance sia equo. Abbiamo poi effettuato la outlier detection, individuando 9 osservazioni con valori di profitto relativi all'anno 2000 molto incoerenti con quelli del 1999. Per ogni metodo confronteremo quindi l'impatto che gli outliers hanno sugli indicatori di performance, in particolare la AUC. In tutti i casi del dataset "pulito" dagli outliers questa è risultata superiore a quella del dataset originale.

I metodi di classificazione che abbiamo scelto sono la regressione logistica, la random forest e il KNN. Per ogni metodo abbiamo seguito gli stessi passaggi:

1. Sul dataset senza outliers abbiamo fatto variare il tipo (numerico o fattoriale) di variabile per i predittori età e salario, che nel dataset sono numeri interi corrispondenti a fasce numeriche.
2. In seguito abbiamo optato per la scelta che dava i migliori risultati in termini di AUC.
3. Abbiamo poi applicato il modello scelto al punto precedente anche sul data set originale contenente gli outliers, confrontando le due curve ROC.
4. Abbiamo riportato le matrici di confusione per i modelli scelti al punto 2., effettuando alcune considerazioni in merito.

In conclusione abbiamo confrontato i tre metodi usati: il metodo che ha prodotto i migliori risultati in termini di AUC è stata la regressione logistica.

Il codice usato è scritto in linguaggio R, ed è presente nella cartella condivisa (*HW1_Agrippino_Di_Renzo*) sotto il nome di *HW1.r*.

Indice

1 Introduzione

1.1 Preparazione dei dati

Osserviamo che per molte righe del dataset i valori di *Income* o *Age* sono Not Available, di conseguenza decidiamo di imputare tali valori con la media, creando le variabili *IncomeAverage* e *AgeAverage*. La scelta di utilizzare la media, e non un altro valore come per esempio lo 0, è dovuta al fatto che in alcuni modelli considereremo tali variabili come quantitative. Come emergerà nelle sezioni successive, è estremamente importante tenere traccia dei clienti di cui la banca non dispone delle informazioni sull'età o sul salario, di conseguenza creiamo le variabili binarie *AgeGiven* e *IncomeGiven*.

Costruiamo poi due variabili binarie *District1100* e *District1200* relative al distretto di appartenenza del cliente (il terzo distretto presente nel dataset, il *1300*, lo si può ricavare dalle altre due variabili)

La variabile *Retain*, oggetto della classificazione, viene creata a partire dalla presenza o meno della variabile *X0Profit*, relativa al profitto che il cliente ha portato alla banca nel 2000. Nello specifico,

$$\text{se} \quad X0Profit == \text{na} \Rightarrow Retain = 0$$

in quanto la banca ha perso il cliente.

Tutte le variabili create vengono dunque inserite all'interno del dataset originario.

1.2 Divisione in training e test set

Procediamo ora alla divisione del dataset in training e test set, in proporzione rispettivamente il 75% e 25%. La partizione dei dati viene fatta in modo casuale tramite la funzione *createDataPartition*, che permette di stratificare sulla variabile oggetto della classificazione: *Retain*. Otteniamo così due dataset con cardinalità:

- **Training set**: 23726
- **Test set**: 7908

1.3 Outliers detection

Analizziamo adesso la presenza di eventuali outliers: plottiamo i valori di *Profit* per l'anno 1999 e per l'anno 2000. Come si può vedere nella figura seguente (Figura 1), ci sono alcuni valori di *X0Profit* totalmente incoerenti con quelli dell'anno precedente.

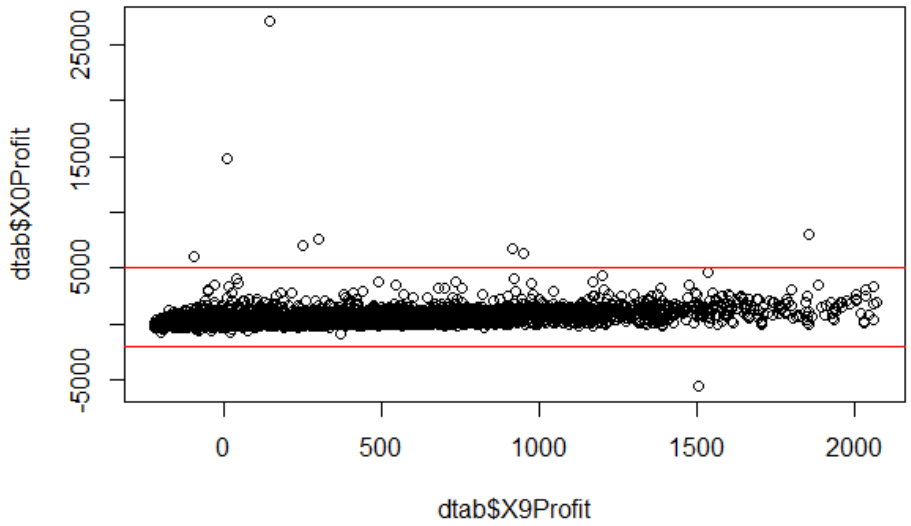

Figura 1: R Plot dei profitti procurati dai clienti alla banca nell'anno 1999 (asse delle ascisse) e 2000 (asse delle ordinate). Le due righe rosse orizzontali corrispondono ai valori di 5000 e -2000: valori al di fuori di questo range vengono da noi considerati outliers.

E' bene ricordare che, sebbene il profitto dell'anno 2000 non venga utilizzato come predittore, esso viene utilizzato nella creazione della variabile binaria *Retain*, oggetto della classificazione. Per queste ragioni, abbiamo deciso di creare un altro dataset in cui eliminiamo i 9 outliers individuati tramite il grafico, conservando anche il dataset originale. Per ogni metodo di classificazione scelto, potremo così confrontare le performance sul dataset con e senza outliers.

1.4 Considerazioni sull'importanza dei predittori

Tutti i predittori presenti nel dataset sono significativi per la predizione di *Retain*? Per rispondere a questa domanda usiamo la funzione di R *filterVarImp* del pacchetto *caret*. Tale funzione calcola la curva ROC per ogni predittore considerato singolarmente: per un problema a due classi, come nel nostro caso, viene applicata una serie di cutoffs al predittore per prevedere la classe. Per ogni cutoff vengono calcolati poi la sensitività e la specificità, e in seguito la curva ROC e l'area al di sotto della stessa, l'AUC.

Vediamo i risultati:

	AUC
X9Profit	0.5517259
X9Online	0.5083683
X9Tenure	0.6366981
AgeAverage	0.5446779
AgeGiven	0.7366330
IncomeAverage	0.5537205
IncomeGiven	0.7360194
District1100	0.5064914
District1200	0.5118495

Figura 2: R output della funzione *filterVarImp*.

Si possono fare alcune interessanti osservazioni:

- L'informazione sul distretto ha un'importanza praticamente nulla nella predizione, di conseguenza non la utilizzeremo.
- Anche l'informazione se il cliente abbia deciso di utilizzare i servizi online (*X9Online*) non sembra essere utile a predire se il cliente abbandonerà la banca l'anno successivo, tuttavia abbiamo scelto di utilizzare questa variabile nei modelli predittivi per via della sua rilevanza nel business case.
- Le variabili *AgeGiven* e *IncomeGiven* sono le più significative, con una AUC elevata: si potrebbe pensare che il cliente di cui la banca non dispone delle informazioni sull'età o il salario (perchè non ha voluto fornirle oppure perchè potrebbe non essere una persona fisica ma un'azienda, per esempio) sia più propenso ad abbandonare la banca l'anno successivo. Tale intuizione verrà confermata nella sezione relativa alla regressione logistica.

E' bene ricordare che la funzione *filterVarImp* fornisce un'informazione generica sull'importanza dei predittori, senza considerare come questi si comportano insieme. Un'analisi più approfondita della significatività dei predittori verrà fatta nella prossima sezione.

2 Regressione logistica

Vediamo il primo metodo implementato: la regressione logistica. Abbiamo innanzitutto analizzato quale scelta sul tipo di variabile (categorica o numerica) relativa a *AgeAverage* e *IncomeAverage* porti a ottenere un'AUC maggiore. Tale scelta si è rivelata essere *AgeAverage* fattoriale e *IncomeAverage* numerico, con una AUC = 0.7869. Abbiamo poi provato tale modello sul dataset con outliers, ottenendo una AUC = 0.7828, e confrontato le due curve ROC. In conclusione, abbiamo effettuato alcune considerazioni sulla matrice di confusione relativa al modello migliore, provando anche a cambiare la soglia di probabilità in modo da migliorare gli indici di sensitività e di precisione.

2.1 Confronto fra modelli: variabili fattoriali o numeriche?

Vediamo il confronto fra modelli che differiscono per il tipo dei predittori *AgeAverage* e *IncomeAverage*. Quando uno di questi due predittori è considerato come fattoriale, non usiamo più la corrispondente variabile binaria che ci fornisce l'indicazione se il dato originale era un missing data (*AgeGiven* e *IncomeGiven*), in quanto tale informazione è già contenuta nel fattore corrispondente alla media.

Abbiamo fatto variare i modelli come segue:

1) Tutti i predittori vengono considerati numerici.

2) Sia *AgeAverage* che *IncomeAverage* categorici.

3) *AgeAverage* categorico, mentre per il salario abbiamo usato solo *IncomeGiven*, senza utilizzare *IncomeAverage*: tale scelta, come verrà illustrato in seguito, è giustificata dai risultati ottenuti nel modello 2).

4) *AgeAverage* fattoriale, *IncomeAverage* numerico.

Vediamo il summary dei primi due modelli, in modo da poter effettuare alcune considerazioni.

```
> summary(lr_allnum)

Call:
glm(formula = Retain ~ X9Profit + X9Online + AgeAverage + AgeGiven +
    IncomeAverage + IncomeGiven + X9Tenure, family = binomial(logit),
    data = dtr)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7641   0.2947   0.3926   0.4662   1.1967

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)   -5.458e-01  1.016e-01  -5.372 7.80e-08 ***
X9Profit       2.334e-04  8.305e-05   2.811  0.00494 **
X9Online       1.405e-01  6.103e-02   2.303  0.02129 *
AgeAverage     7.744e-02  1.848e-02   4.190 2.79e-05 ***
AgeGiven       1.127e+00  8.840e-02  12.752  < 2e-16 ***
IncomeAverage  3.721e-02  1.210e-02   3.077  0.00209 **
IncomeGiven    1.043e+00  8.808e-02  11.842  < 2e-16 ***
X9Tenure       3.591e-02  2.774e-03  12.943  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 21400  on 23721  degrees of freedom
Residual deviance: 17591  on 23714  degrees of freedom
AIC: 17607

Number of Fisher Scoring iterations: 5
```

Figura 3: Summary della funzione *glm* di R che implementa la regressione logistica in cui tutti i predittori vengono considerati numerici.

Dal summary del primo modello (Figura 3) possiamo osservare come la variabile *X9Online* conservi una, seppur debole, significatività. Esistono tuttavia variabili molto più significative: come già accennato nella sezione introduttiva, il modello ci dice che i clienti di cui abbiamo informazioni sull'età e sul salario sono molto più propensi a restare nella banca anche l'anno successivo, infatti i coefficienti di *AgeGiven* e *IncomeGiven* sono positivi e addirittura maggiori di 1 (ricordiamo che *Retain*= 1 corrisponde all'avere mantenuto il cliente).
Discorso analogo per la variabile *X9Tenure*: più aumentano gli anni da cui il cliente è affiliato alla banca e meno è probabile che il cliente ci lasci l'anno successivo.

Possiamo inoltre dedurre dal modello che la fedeltà del cliente aumenti, sebbene non marcatamente, con l'età anagrafica: il coefficiente di *AgeAverage* è positivo, anche se molto piccolo, ma anch'esso altamente significativo.

La significatività del predittore *Profit* è invece meno marcata e questo scontenterà i manager della banca, in quanto dal segno del coefficiente sembra che più aumenti il profitto, più sia alta la probabilità che il cliente resti tale anche l'anno successivo.

```
Coefficients:
                                        Estimate Std. Error z value Pr(>|z|)
(Intercept)                            1.533e+00  1.392e-01  11.009  < 2e-16 ***
X9Profit                               2.192e-04  8.325e-05   2.633  0.00845 **
X9Online                               1.439e-01  6.115e-02   2.353  0.01861 *
factor(AgeAverage)2                    2.365e-01  1.323e-01   1.787  0.07390 .
factor(AgeAverage)3                    6.257e-01  1.335e-01   4.685 2.79e-06 ***
factor(AgeAverage)4                    7.191e-01  1.371e-01   5.244 1.57e-07 ***
factor(AgeAverage)4.04604840436924    -6.159e-01  1.387e-01  -4.440 8.98e-06 ***
factor(AgeAverage)5                    6.177e-01  1.470e-01   4.202 2.65e-05 ***
factor(AgeAverage)6                    4.579e-01  1.531e-01   2.991  0.00278 **
factor(AgeAverage)7                    6.472e-01  1.537e-01   4.212 2.53e-05 ***
factor(IncomeAverage)2                -8.274e-02  1.662e-01  -0.498  0.61859
factor(IncomeAverage)3                 2.835e-03  1.199e-01   0.024  0.98113
factor(IncomeAverage)4                 6.147e-03  1.253e-01   0.049  0.96088
factor(IncomeAverage)5                -6.954e-02  1.237e-01  -0.562  0.57391
factor(IncomeAverage)5.45877722158046 -9.448e-01  1.172e-01  -8.060 7.66e-16 ***
factor(IncomeAverage)6                 2.069e-01  1.095e-01   1.890  0.05882 .
factor(IncomeAverage)7                 3.321e-02  1.186e-01   0.280  0.77943
factor(IncomeAverage)8                -1.273e-01  1.321e-01  -0.964  0.33506
factor(IncomeAverage)9                 2.871e-01  1.309e-01   2.193  0.02828 *
X9Tenure                               3.594e-02  2.773e-03  12.964  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 21400  on 23721  degrees of freedom
Residual deviance: 17539  on 23702  degrees of freedom
AIC: 17579
```

Figura 4: Summary della funzione *glm* di R che implementa la regressione logistica in cui sia *AgeAverage* che *IncomeAverage* sono considerati categorici.

Commentiamo adesso il summary del secondo modello, in cui sia l'età che il salario vengono considerate variabili categoriche (Figura 4). Per *X9Profit*, *X9Online* e *X9Tenure* valgono le considerazioni effettuate in precedenza. Per quanto riguarda gli altri predittori, è interessante notare come non tutte le età siano significative allo stesso modo, mentre tutte (tranne il livello corrispondente alla media, cioè ai NA del dataset originale) abbiano coefficiente positivo. Ciò conferma l'ipotesi che i clienti di cui la banca non dispone delle informazioni angarafiche siano quelli che con più probabilità lasceranno la banca.

Per quanto riguarda *IncomeAverage*, invece, l'unico livello significativo è quello corrispondente ai missing data imputati con la media, di conseguenza possiamo dedurre che conoscere il salario del cliente non ci dia molte indicazioni sulla sua fedeltà, mentre molto importante è sapere se tale informazione sia disponibile o meno. Queste considerazioni giustificano la scelta effettuata nel modello 3 di non inserire all'interno dei predittori la variabile *IncomeAverage* ma solo *IncomeGiven*. Come si vedrà nelle prossime righe tale scelta infatti non penalizzerà le performance del modello.

2.1.1 Qual è la scelta migliore?

Per rispondere a tale domanda riportiamo i risultati delle performance dei quattro metodi presentati, in particolare l'indice AUC, l'*Area Under the Curve* della curva ROC.

```
  auclr_1   auclr_2   auclr_3   auclr_4
0.7869585 0.7838001 0.7844142 0.7865696
```

Figura 5: R output dei valori di AUC per i quattro modelli di Regressione Logistica descritti in precedenza.

Come si può notare, il modello migliore è risultato essere il primo, in cui tutti i predittori sono considerati come numerici, sebbene di poco rispetto al quarto. E' interessante osservare come l'AUC del terzo modello sia superiore al secondo, nonostante nel terzo abbiamo utilizzato un predittore in meno, *Income Average*: questo per i motivi analizzati alla fine del paragrafo precedente.

2.2 Confronto del modello migliore trovato con e senza outliers

Andiamo adesso a verificare l'impatto che hanno gli outliers sulle performance del modello migliore trovato al punto precedente, cioè quello con tutti i predittori numerici.

In Figura 6 abbiamo riportato le curve ROC del modello applicato ai dataset con e senza outliers. Data la scarsa cardinalità del numero di outliers, appena 9 su un dataset di 31634 osservazioni, come ci si poteva aspettare le due curve ROC sono quasi indistinguibili. La differenza è resa evidente dal calcolo della AUC. Considerando anche gli outliers, infatti, l'indice di performance peggiora, passando da 0.7869 a 0.7828. Ciò sembra suggerire di non considerare le osservazioni che presentano valori di *X0Profit* eccessivamente elevati o bassi.

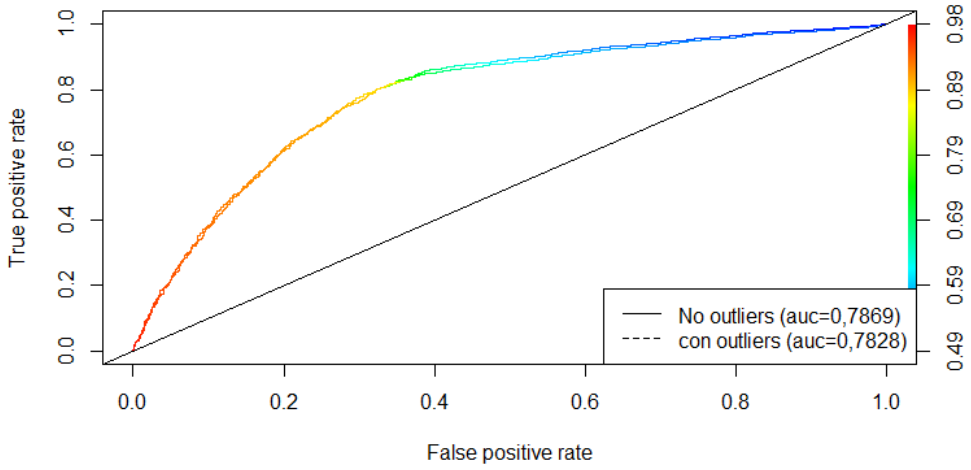

Figura 6: Curve ROC della regressione logistica con tutti i predittori numerici applicata al dataset senza e con outliers.

2.3 Confusion matrix

In conclusione della sezione dedicata al metodo della regressione logistica, analizziamo la matrice di confusione data dal modello migliore applicato al dataset senza outliers.

Per chiarezza riportiamo nella figura seguente le definizioni degli indici calcolati sulla matrice di confusione, alcuni dei quali useremo in seguito. Nel nostro caso, con il termine *positive* consideriamo i clienti che lasceranno la banca (classe 0), il vero target della classificazione.

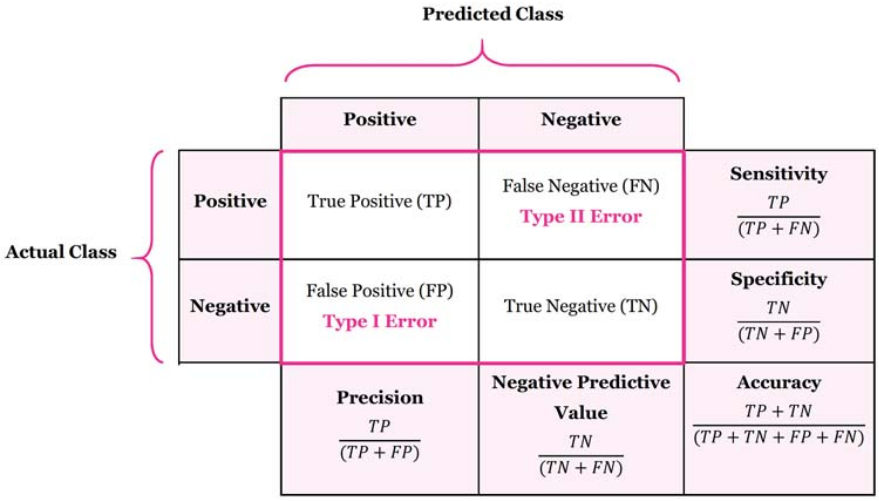

Figura 7: Definizioni indici confusion matrix

Vediamo dapprima la matrice di confusione ottenuta settando la soglia di probabilità che un'osservazione appartenga alla classe *Retain*= 1 al valore standard 0.5.

	predizioni	
real	0	1
0	69	1208
1	67	6559

Figura 8: Confusion matrix con soglia di probabilità fissata a 0.5

Come si può notare dalla figura 8, il metodo tende ad assegnare alla quasi totalità del test set la classe Retain = 1, di conseguenza la sensitività nell'individuare i clienti che abbandoneranno la banca è pessima (5.4%), un po' meno la precisione (50.7%), ma è chiaro che il metodo fallisce l'obiettivo di prevedere i clienti insoddisfatti.

Proviamo quindi a cambiare la soglia di probabilità, anche alla luce della curva ROC in figura 6. Settando tale soglia a 0.70, otteniamo la seguente matrice di confusione:

	predizioni	
real	0	1
0	778	499
1	947	5679

Figura 9: Confusion matrix con soglia di probabilità fissata a 0.7

Possiamo notare come la sensitività sia notevolmente aumentata, passando al 60.9%, mentre la precisione è scesa solo in parte rispetto a prima: 45.1%. L'accuratezza è passata da 83.9% a 81.7%.

Supponendo che la banca sia più interessata a individuare i clienti che la abbandoneranno rispetto a quelli fedeli, ecco che questa seconda confusion matrix diventa molto più apprezzabile della precedente. Di concerto con il management della banca, si potrebbe quindi scegliere una soglia di probabilità che gestisca al meglio il trade off fra sensitività e precisione: è facile ipotizzare che per la banca sia meglio dare incentivi anche a chi in realtà non abbia intenzione di cambiare (bassa precisione) rispetto a non individuare e quindi perdere i clienti insoddisfatti (bassa sensitività).

3 Random forest

Il secondo metodo di classificazione analizzato è la *classification random forest*. I parametri usati dall'algoritmo sono quelli standard della funzione di R *randomForest* del pacchetto omonimo, cioè:

- 2 variabili fra cui l'algoritmo sceglie ad ogni split
- 500 alberi per ogni foresta

In questo caso la scelta che ha portato al maggior valore di AUC è stata quella di trattare sia l'età che il salario come predittori categorici, con una AUC= 0.7046, inferiore a quella ottenuta dalla regressione logistica. Anche in questo caso, inoltre, il data set con outliers performa peggio del dataset pulito.

Utilizzando la funzione *randomForest::importance* abbiamo poi ottenuto ulteriore conferma della scarsa utilità dell'informazione sull'utilizzo dei servizi online per predire la fedeltà dei clienti.

Nei paragrafi seguenti vedremo più in dettaglio i risultati ottenuti.

3.1 Confronto tra modelli

Le possibili scelte su come considerare le variabili *AgeAverage* e *IncomeAverage* sono riassunte nel seguente elenco:

1. Tutti i predittori sono considerati numerici.
2. Sia l'età che il salario sono considerate variabili categoriche.
3. *AgeAverage* fattoriale, *IncomeAverage* numerico.
4. *AgeAverage* numerico, *IncomeAverage* fattoriale.

3.1.1 Qual è il modello migliore?

Riportiamo nella figura seguente (10) i valori di AUC ottenuti per tutti e quattro le scelte di modellizzazione presentate sopra.

```
aucrf_confronto
  aucrf_1   aucrf_2   aucrf_3   aucrf_4
0.6870744 0.7046135 0.6932348 0.6843346
```

Figura 10: Valori di AUC dei quattro modelli di random forest sul dataset senza outliers

In questo caso, contrariamente alla regressione logistica, il modello migliore è quello ottenuto considerando sia l'età che il salario come variabili fattoriali. Questo può essere spiegato dalle caratteristiche dell'algoritmo alla base degli alberi di classificazione (split successivi in modo da ridurre le impurità dei nodi ottenuti dallo split), in cui le variabili categoriche vengono trattate più efficacemente.

3.2 Confronto del modello con e senza outliers

Una volta trovato il modello migliore, lo applichiamo anche al dataset originale contenente i 9 outliers individuati in precedenza. Vediamo dunque le curve ROC nei due casi nella figura seguente (11).

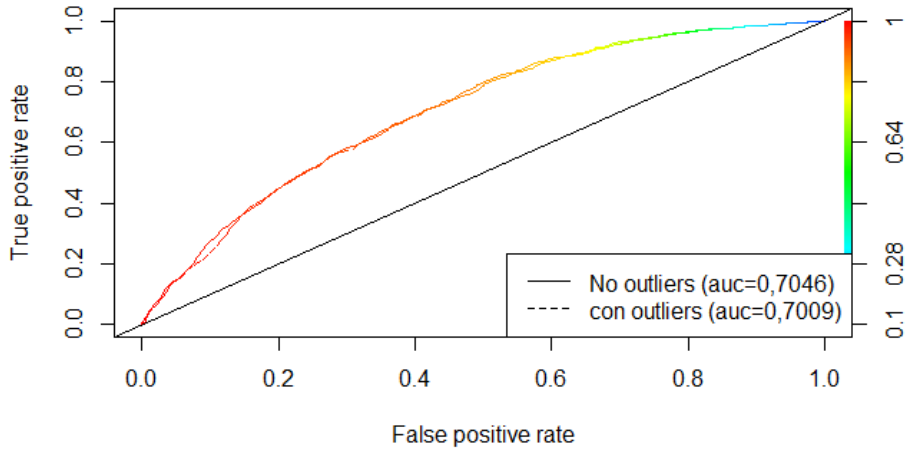

Figura 11: Curve ROC del modello con sia età che salario fattoriali applicato al dataset orginale (linea tratteggiata) e a quello pulito (linea continua)

Le due curve sono leggermente più distinguibili che nel caso precedente, sebbene risultino ovviamente quasi sovrapponibili (ricordiamo che gli outliers sono 9 su un dataset di 31634 osservazioni). Ancora una volta calcolando le due AUC emerge come la scelta di eliminare gli outliers abbia pagato in termini di performance.

3.3 Confusion matrix

Riportiamo anche in questo caso la matrice di confusione relativa al modello migliore applicato al dataset senza outliers. Di seguito la confusion matrix ottenuta fissando la soglia di probabilità a 0.5.

real	predrf_2 0	1
0	235	1042
1	203	6423

Figura 12: Confusion matrix con soglia di probabilità fissata a 0.5

Come nel caso precedente osserviamo che con questa soglia di probabilità il metodo tende ad assegnare alla quasi totalità del dataset la classe *Retain*= 1, con il risultato di avere un pessimo indice di sensitività: 18.4%.
Aumentiamo dunque tale soglia di probabilità: osservando la curva ROC ed effettuando alcune prove, abbiamo scelto come soglia 0.82. Otteniamo:

real	predizioni 0	1
0	522	755
1	862	5764

Figura 13: Confusion matrix con soglia di probabilità fissata a 0.82

La sensitività è salita al 40.8%, mentre la precisione è scesa al 37.7%. Notiamo che la coperta è più corta che nel caso precedente della regressione logistica, ciò è giustificato dal fatto che la AUC per la random forest è minore (0.704 vs 0.787).

3.4 Importanza dei predittori nella random forest

Purtroppo la random forest non permette di interpretare i risultati ottenuti quanto la regressione logistica, tuttavia possiamo ottenere un'indicazione dell'importanza che ciascun predittore ha avuto nello stimare il modello. In particolare, per ogni predittore viene calcolata la *Mean Decrease Accuracy* permutando i dati Out-Of-Bag: per ogni albero della foresta il prediction error dei dati OOB viene confrontato con quello ottenuto dopo aver permutato ogni predittore singolarmente. In seguito tale differenza viene mediata su tutti gli alberi, e normalizzata per la deviazione standard. Un procedimento analogo è seguito per stimare il decremento medio dell'indice di Gini, indicatore della impurità del nodo.

Tali risultati sono riassunti nei due grafici in figura 14

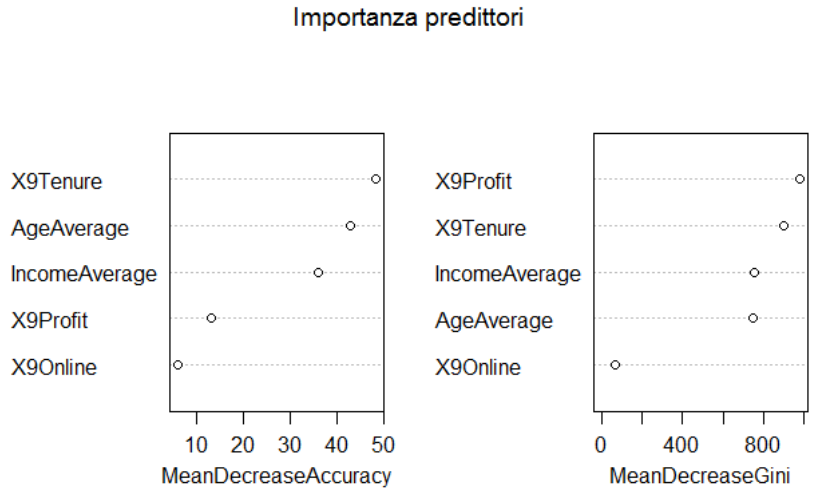

Figura 14: Plot ottenuto dalla funzione di R *varImpPlot*, che prende in input un modello di tipo random forest e restituisce l'importanza dei predittori.

E' interessante osservare come la variabile *Online* abbia molta poca influenza nel prevedere l'abbandono del cliente, così come avevamo notato già nella sezione introduttiva. Invece riveste grande importanza la variabile *Tenure*, cioè il numero di anni da cui il cliente ha aperto un conto.

4 KNN

Vediamo ora il metodo K Nearest Neighbors, noto come KNN.

Come in precedenza, abbiamo provato 4 diversi modelli sul dataset senza outliers facendo variare la scelta del tipo di variabili *Age* e *Income*, per poi confrontarli usando l'indice AUC e riprovando il modello migliore sull'intero dataset.

Per ogni modello provato, abbiamo scelto il valore di K (da 1 a 50) da passare all'algoritmo come quello che massimizza l'accuratezza.

Il modello migliore è risultato essere quello dove tutti i predittori sono considerati numerici, ottenendo un'AUC = 0.771. In questo caso il dataset con outliers ha dato risultati peggiori rispetto al dataset "pulito", sebbene come in precedenza la differenza fra i due sia minima.

Riportiamo adesso più in dettaglio il lavoro svolto, e alcune considerazioni sulla matrice di confusione del modello migliore ottenuto.

4.1 Confronto fra modelli: fattoriale o numerico?

Vediamo il confronto fra i modelli in cui abbiamo cambiato il tipo dei predittori *AgeAverage* e *IncomeAverage*. Come al solito, quando uno di questi due predittori è considerato come fattoriale, non usiamo più la corrispondente variabile binaria che ci fornisce l'indicazione se il dato originale fosse un missing data

(*AgeGiven* e *IncomeGiven*), in quanto tale informazione è già contenuta nel fattore corrispondente alla media.

E' bene ricordare che l'algoritmo del KNN lavora esclusivamente su dati numerici, quindi trasformare un predittore in categorico significa che l'algoritmo utilizzerà tante variabili numeriche con dominio $\{0, 1\}$ quanti sono i livelli del predittore, anzichè una sola variabile numerica.

Dato che il KNN si basa sul calcolo delle distanze euclidee fra le osservazioni del dataset, prima di applicare l'algoritmo abbiamo normalizzato le variabili in modo che avessero tutte la stessa importanza predittiva nel calcolo della distanza.

Anche in questo caso abbiamo eseguito quattro modelli diversi:

1. Tutti i predittori vengono considerati numerici.
2. Sia *AgeAverage* che *IncomeAverage* vengono considerati fattoriali.
3. *AgeAverage* fattoriale, *IncomeAverage* numerico.
4. *AgeAverage* numerico, *IncomeAverage* fattoriale.

Per ogni modello abbiamo scelto il valore di K provando tutti i valori da 1 a 50 e scegliendo il valore corrispondente alla massima accuratezza, di seguito riportiamo il grafico relativo al primo modello (figura 15):

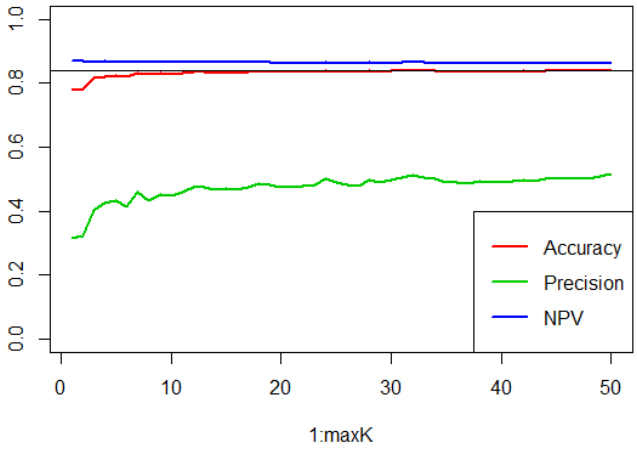

Figura 15: Plot dell'indice di accuratezza, del *True Positive Rate* e *False Positive rate*, dove Positive è associato a *Retain=1*, quindi al cliente mantenuto.

Come si può notare dalla figura, dopo i primi valori di K l'accuratezza si assesta, facendo sì che la curva sia quasi piatta.

4.1.1 Qual è la scelta migliore?

Ora che abbiamo spiegato come abbiamo ottenuto i modelli, andiamo ad analizzare i risultati in termini di AUC. Otteniamo indicatori di performance molto simili fra loro, la AUC più alta si ha per i modelli 1 e 3, come evidenziato dalla figura seguente.

```
  aucknn1   aucknn2   aucknn3   aucknn4
0.7710261 0.767958 0.7710261 0.7704342
```

Figura 16: Output di R contenente i valori di AUC per i quattro modelli descritti sopra.

4.2 Confronto del modello migliore trovato con e senza outliers

In questo paragrafo andremo ad applicare sul dataset originale (con outliers) il modello migliore trovato sulla base dei risultati riportati in figura 16: avendo un ex-aequo scegliamo arbitrariamente il primo modello, quello in cui tutti i predittori vengono considerati numerici. Come in precedenza, effettuiamo il confronto sulla base delle curve ROC. Come si vede dalla figura 17, la differenza fra le due è minima, sebbene anche in questo caso la AUC più alta sia quella ottenuta dal modello senza outliers.

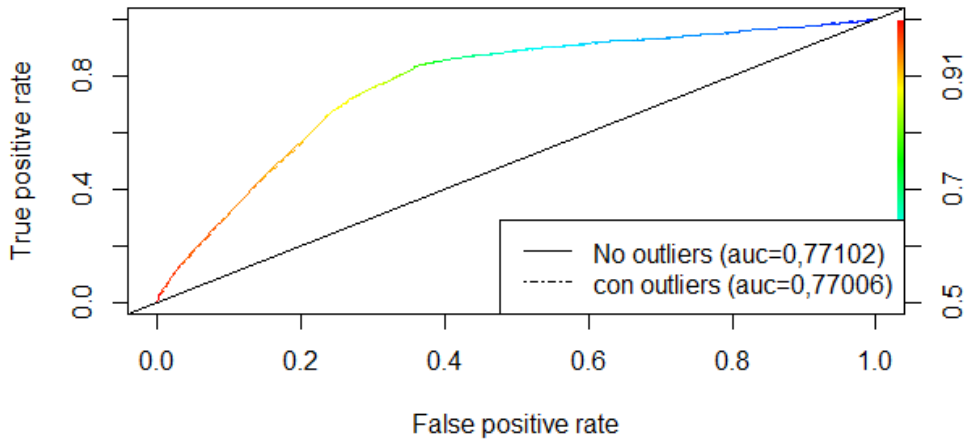

Figura 17: Curve ROC per il modello nel caso del dataset senza outliers (linea continua) e dataset originale (linea tratteggiata).

4.3 Confusion matrix

Commentiamo adesso la Confusion Matrix ottenuta a partire dal modello migliore (figura 18), dove l'osservazione test è stata assegnata alla classe più rappresentata fra i suoi vicini, scegliendo in maniera casuale in caso di pareggi.

Notiamo che nonostante la accuratezza sia elevata ($\approx 84\%$), il modello fallisce nel predirre i clienti che la banca ha perso (classe 0), con una sensitività molto bassa: 21.4%.

	knn_best1	
te.cl	0	1
0	273	1004
1	264	6362

Figura 18: Confusion matrix relativa ai risultati ottenuti considerando tutti i predittori come categorici e usando il dataset senza outliers. *knn_best1* rappresenta le predizioni del modello, mentre *te.cl* le classi reali.

Come nei casi precedenti, è sempre possibile intervenire per migliorare la sensitività cambiando la soglia di probabilità per l'assegnazione alla classe $Retain = 1$. Nel caso del metodo KNN, tale probabilità è data dalla percentuale dei K vicini dell'osservazione test che sono di classe 1. Vediamo dunque la matrice di confusione ottenuta ponendo la soglia di probabilità a 0.78:

real	predizioni 0	1
0	820	457
1	1120	5506

Figura 19: Confusion matrix con soglia di probabilità a 0.78

In questo caso la sensitività sale al 64.2%, mentre la precisione si attesta al 42.2%.

5 Considerazioni finali

Il metodo che ha restituito le performance migliori è stata la regressione logistica, che ha anche il notevole vantaggio di avere risultati facilmente interpretabili. Anche il KNN ha fornito risultati soddisfacenti, di poco inferiori a quelli della regressione logistica, sebbene non abbia la stessa interpretabilità. Si potrebbe dunque pensare di utilizzare i due metodi per una predizione più accurata. La random forest non ha avuto invece ottimi indicatori di performance: a nostro avviso tale metodo andrebbe scartato.

Dal punto di vista dell'interpretazione dei risultati, la nostra analisi ha riportato la notevole importanza delle variabili *AgeGiven* e *IncomeGiven* nel predire la customer retention. Bisognerebbe approfondire le ragioni per cui nel dataset ci sono dati non disponibili, tenendo presente che quando questo accade è molto più probabile che il cliente abbandoni la banca. L'informazione sull'Online non sembra invece ricoprire un ruolo decisivo: ci sentiamo dunque di affermare che è sbagliato dire che chi abbia attivato i servizi online sia più o meno fedele alla banca.