Costruzione di un modello logistico relativo ad un e-commerce francese

Martina Chiesa 837484, Carlo Saccardi 839641, Davide Valoti 846737

19/11/2020

Importiamo nuovamente il dataset di partenza relativo all'e-commerce francese e ci concentriamo sulla distinzione delle due categorie: venditori e non venditori, tramite la costruzione di un modello logistico a fini interpretativi.

Analisi valori mancanti

```
##      identifierHash                type             country            language 
##                   0                   0                   0                   0 
##   socialNbFollowers     socialNbFollows socialProductsLiked      productsListed 
##                   0                   0                   0                4944 
##        productsSold    productsPassRate      productsWished      productsBought 
##                   0                   0                3208                   0 
##              gender    civilityGenderId       civilityTitle            hasAnyApp 
##                   0                   0                   0                   0 
##       hasAndroidApp           hasIosApp   hasProfilePicture  daysSinceLastLogin 
##                   0                   0                   0                   0 
##           seniority   seniorityAsMonths    seniorityAsYears         countryCode 
##                   0                   0                   0                   0
```

Sono presenti 4944 valori mancanti per *productsListed* e 3208 per *productsWished*, quindi rimuoviamo le osservazioni che presentano NA, in modo tale che il modello lavorerà solo sui dati completi.

Variabile target

```
##     0     1 
## 91332  1920

##          0          1 
## 0.97941063 0.02058937
```

Creiamo una variabile dipendente binaria *venditore*, che assume valore 1 in caso di soggetti che hanno venduto almeno un prodotto e 0 se si tratta di individui che non hanno mai venduto ed eliminiamo la variabile *productsSold* dal dataset. Appare evidente che non c'è equilibrio tra le due categorie di individui, infatti i venditori costituiscono solo il 2% degli utenti registrati. Procediamo a questo punto con la ripulitura del dataset da variabili contatore (*identifierHash*) e con un numero elevato di livelli (*countryCode* e *country*). Inoltre, trasformiamo in fattore la variabile *civilityGenderId* che assume valori 1, 2 e 3, come ricodifica di *civilityTitle*.

Modello iniziale

Il modello iniziale con tutte le variabili genera errore nella stima, dunque verifichiamo la presenza di collinearità, zero variance e separation.

Valutazione status delle variabili

```
##               variable q_zeros p_zeros q_na p_na q_inf p_inf    type unique
## 1                 type       0    0.00    0    0     0     0  factor      1
## 2             language       0    0.00    0    0     0     0  factor      5
## 3    socialNbFollowers       0    0.00    0    0     0     0 integer     86
## 4      socialNbFollows      37    0.04    0    0     0     0 integer     82
## 5  socialProductsLiked   78222   83.88    0    0     0     0 integer    410
## 6       productsListed   91626   98.26    0    0     0     0 integer     64
## 7     productsPassRate   92374   99.06    0    0     0     0 numeric     70
## 8       productsWished   84461   90.57    0    0     0     0 integer    273
## 9       productsBought   88137   94.51    0    0     0     0 integer     70
## 10              gender       0    0.00    0    0     0     0  factor      2
## 11    civilityGenderId       0    0.00    0    0     0     0  factor      3
## 12       civilityTitle       0    0.00    0    0     0     0  factor      3
## 13           hasAnyApp       0    0.00    0    0     0     0  factor      2
## 14       hasAndroidApp       0    0.00    0    0     0     0  factor      2
## 15           hasIosApp       0    0.00    0    0     0     0  factor      2
## 16   hasProfilePicture       0    0.00    0    0     0     0  factor      2
## 17  daysSinceLastLogin       0    0.00    0    0     0     0 integer    699
## 18           seniority       0    0.00    0    0     0     0 integer     19
## 19   seniorityAsMonths       0    0.00    0    0     0     0 numeric     19
## 20    seniorityAsYears       0    0.00    0    0     0     0 numeric      6
## 21           venditore   91332   97.94    0    0     0     0 numeric      2
```

Notiamo che per molte variabili la percentuale di 0 è molto elevata, ad esempio per *productsPassRate* è presente il 99% di zeri nel dataset. Inoltre, si osserva una variabile che presenta un solo livello (*type*), quindi la rimuoviamo, poichè affetta da zero variance.

Collinearità quantitative

Il modello costruito con le sole variabili numeriche genera il seguente warning:

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

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

Potrebbe sussistere un problema di collinearità o separation.

```
## All Individual Multicollinearity Diagnostics Result
##
##                               VIF    TOL           Wi           Fi Leamer
## socialNbFollowers          3.1191 0.3206 1.975906e+04 2.195475e+04 0.5662
## socialNbFollows            4.2429 0.2357 3.023673e+04 3.359673e+04 0.4855
## socialProductsLiked        3.4931 0.2863 2.324612e+04 2.582930e+04 0.5350
## productsListed             1.3283 0.7528 3.061003e+03 3.401151e+03 0.8677
## productsPassRate           1.3468 0.7425 3.233419e+03 3.592726e+03 0.8617
## productsWished             1.3664 0.7318 3.416441e+03 3.796087e+03 0.8555
## productsBought             1.2329 0.8111 2.171955e+03 2.413309e+03 0.9006
## daysSinceLastLogin         1.0871 0.9198 8.125755e+02 9.028714e+02 0.9591
## seniority            4714070.6094 0.0000 4.395446e+10 4.883881e+10 0.0005
## seniorityAsMonths    4623101.5506 0.0000 4.310625e+10 4.789635e+10 0.0005
## seniorityAsYears       26111.7463 0.0000 2.434592e+08 2.705131e+08 0.0062
##                              CVIF Klein  IND1   IND2
## socialNbFollowers          5.9283     1 0e+00 1.2055
## socialNbFollows            8.0641     1 0e+00 1.3562
## socialProductsLiked        6.6391     1 0e+00 1.2664
## productsListed             2.5246     0 1e-04 0.4385
## productsPassRate           2.5597     0 1e-04 0.4569
```

```
## productsWished            2.5970     0 1e-04 0.4758
## productsBought            2.3434     0 1e-04 0.3352
## daysSinceLastLogin        2.0663     0 1e-04 0.1422
## seniority           8959726.4456     1 0e+00 1.7744
## seniorityAsMonths   8786827.4908     1 0e+00 1.7744
## seniorityAsYears      49628.8926     1 0e+00 1.7743
##
## 1 --> COLLINEARITY is detected by the test
## 0 --> COLLINEARITY is not detected by the test
##
## productsListed , seniority , seniorityAsMonths , coefficient(s) are non-significant may be d
ue to multicollinearity
##
## R-square of y on all x: 0.4615
##
## * use method argument to check which regressors may be the reason of collinearity
## ===================================
```

Le ultime tre variabili presentano valori molto elevati di VIF e pari a 0 di tolleranza. Dunque, rimuoviamo *seniority* dal modello, ma il problema permane, quindi ripetiamo il calcolo delle diagnostiche. A questo punto, procediamo con la rimozione di *seniorityAsYears* in accordo con le soglie di VIF e TOL nuovamente calcolate, ma il modello continua a presentare problemi. Togliamo ora *socialNbFollows* perchè, nonostante il VIF presenti un valore accettabile secondo la nostra soglia, il TOL è minore di 0.3 e anche la colonna Klein indica multicollinearità. Ora l'algoritmo converge e il modello viene stimato (*modello_numeriche <- glm(venditore ~ socialNbFollowers + socialProductsLiked + productsListed + productsPassRate + productsWished + productsBought + daysSinceLastLogin + seniorityAsMonths, data = data_numeric, family = binomial)*). Tuttavia, genera il seguente warning:

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

Collinearità qualitative

```
##    X1              Row            Column df p.value Chi.Square.norm
## 1   1         language            gender  4   0.000    5.724324e-03
## 2   2         language  civilityGenderId  8   0.000    3.252329e-03
## 3   3         language     civilityTitle  8   0.000    3.252329e-03
## 4   4         language         hasAnyApp  4   0.000    8.394698e-03
## 5   5         language     hasAndroidApp  4   0.000    2.166337e-02
## 6   6         language         hasIosApp  4   0.000    6.061634e-03
## 7   7         language hasProfilePicture  4   0.000    1.422144e-03
## 8   8           gender  civilityGenderId  2   0.000    1.000000e+00
## 9   9           gender     civilityTitle  2   0.000    1.000000e+00
## 10 10           gender         hasAnyApp  1   0.000    7.843245e-03
## 11 11           gender     hasAndroidApp  1   0.000    1.968852e-03
## 12 12           gender         hasIosApp  1   0.000    5.323514e-03
## 13 13           gender hasProfilePicture  1   0.166    2.054262e-05
## 14 14 civilityGenderId     civilityTitle  4   0.000    1.000000e+00
## 15 15 civilityGenderId         hasAnyApp  2   0.000    8.312527e-03
## 16 16 civilityGenderId     hasAndroidApp  2   0.000    3.063704e-03
## 17 17 civilityGenderId         hasIosApp  2   0.000    5.373178e-03
## 18 18 civilityGenderId hasProfilePicture  2   0.000    5.436791e-03
## 19 19    civilityTitle         hasAnyApp  2   0.000    8.312527e-03
## 20 20    civilityTitle     hasAndroidApp  2   0.000    3.063704e-03
## 21 21    civilityTitle         hasIosApp  2   0.000    5.373178e-03
## 22 22    civilityTitle hasProfilePicture  2   0.000    5.436791e-03
## 23 23        hasAnyApp     hasAndroidApp  1   0.000    1.428188e-01
## 24 24        hasAnyApp         hasIosApp  1   0.000    7.724537e-01
## 25 25        hasAnyApp hasProfilePicture  1   0.000    1.945901e-02
## 26 26    hasAndroidApp         hasIosApp  1   0.000    9.862512e-03
```

```
## 27 27    hasAndroidApp hasProfilePicture  1   0.000    1.490561e-03
## 28 28        hasIosApp hasProfilePicture  1   0.000    1.759137e-02
```

Calcoliamo il chi-quadro normalizzato tra ogni combinazione di coppie di variabili qualitative e osserviamo che le variabili *civilityGenderId*, *gender* e *civilityTitle* presentano perfetta associazione, pertanto ne eliminiamo due.

Modello completo

Il modello completo presenta ancora problemi (*error:glm.fit: fitted probabilities numerically 0 or 1 occurred*), quindi procediamo con il calcolo del test LRT.

```
## Single term deletions
##
## Model:
## venditore ~ language + socialNbFollowers + socialProductsLiked +
##     productsListed + productsPassRate + productsWished + productsBought +
##     civilityTitle + hasAnyApp + hasAndroidApp + hasIosApp + hasProfilePicture +
##     daysSinceLastLogin + seniorityAsMonths
##                     Df Deviance    AIC     LRT  Pr(>Chi)
## <none>                   8097.9 8135.9
## language             4   8262.7 8292.7  164.83 < 2.2e-16 ***
## socialNbFollowers    1   8412.1 8448.1  314.19 < 2.2e-16 ***
## socialProductsLiked  1   8126.4 8162.4   28.49 9.435e-08 ***
## productsListed       1   8506.2 8542.2  408.33 < 2.2e-16 ***
## productsPassRate     1   9735.1 9771.1 1637.22 < 2.2e-16 ***
## productsWished       1   8122.7 8158.7   24.78 6.415e-07 ***
## productsBought       1   8115.0 8151.0   17.12 3.506e-05 ***
## civilityTitle        2   8296.5 8330.5  198.65 < 2.2e-16 ***
## hasAnyApp            1   8101.8 8137.8    3.97   0.04624 *
## hasAndroidApp        1   8098.2 8134.2    0.29   0.59173
## hasIosApp            1   8098.0 8134.0    0.11   0.74051
## hasProfilePicture    1   8114.5 8150.5   16.63 4.540e-05 ***
## daysSinceLastLogin   1   8762.1 8798.1  664.17 < 2.2e-16 ***
## seniorityAsMonths    1   8099.0 8135.0    1.15   0.28429
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notiamo valori molto elevati per questo test, quello maggiore è assunto dalla variabile *productsPassRate*, quindi indaghiamo la separation tra questa variabile e quella dipendente.

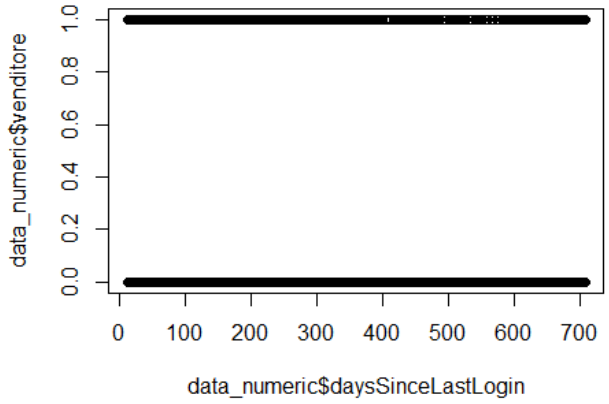
```
##        0    25    28    31    33    35    37    40  41.6    42    43    50
##   0 91332     0     0     0     0     0     0     0     0     0     0     0
##   1  1042     5     1     1     8     1     2     2     1     1     1    54
##
##       53    55    57    58    60    61    62  62.5    63    64    66  66.6
##   0    0     0     0     0     0     0     0     0     0     0     0     0
##   1    2     2     3     3    13     2     4     1     3     2    61     1
##
##       68    69  69.2    70    71    72  72.7    74    75    76    77  77.7
##   0    0     0     0     0     0     0     0     0     0     0     0     0
##   1    1     1     1     5     6     1     1     3    42     5     8     1
##
##       78    79    80    81    82  82.6    83  83.3    84    85  85.7    86
##   0    0     0     0     0     0     0     0     0     0     0     0     0
##   1    4     2    17     4     2     1    23     3    10    18     1     3
##
##       87    88  88.8    89  89.1    90  90.9    91  91.6    92  92.3  92.8
##   0    0     0     0     0     0     0     0     0     0     0     0     0
```

```
##   1    15    14     1     5     1    22     4     9     1    11     3     1
##
##        93    94    95    96  96.2  96.4    98  98.7    99   100
##   0     0     0     0     0     0     0     0     0     0     0
##   1    12     8     5     5     1     1     7     1     1   413
```

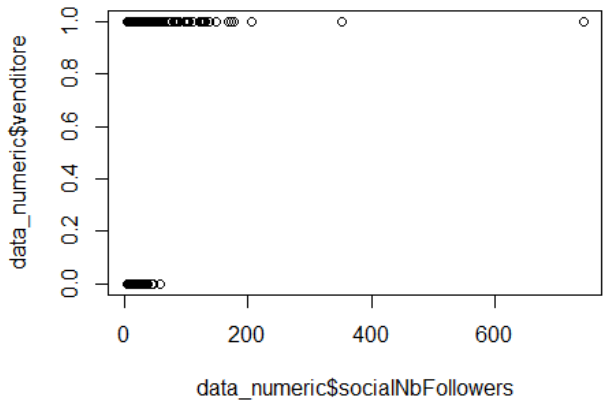
Sono presenti numerosi 0 nella matrice, quindi c'è quasi perfetta separation. Questo risultato rispecchia le nostre aspettative, poichè un non venditore non può avere una percentuale maggiore di zero di prodotti la cui descrizione è coerente con il bene venduto. Stimiamo il modello escludendo questa variabile e osserviamo che i warning sussistono, quindi ripetiamo il calcolo del test LRT sulle variabili restanti e il valore più alto per il test è presentato da *productsListed* (1193.49).

```
##         0     1     2     3     4     5     6     7     8     9    10    11
##   0 90690   459   114    27    12    13     4     3     1     2     2     1
##   1   936   306   147   117    73    48    38    32    27    18    19    13
##
##        12    13    14    15    16    17    18    19    20    21    22    23
##   0     1     0     0     0     0     1     0     0     0     0     0     1
##   1    12    11     6     6     4     6     8     6     7     6     6     7
##
##        24    25    26    27    28    31    32    33    34    35    37    38
##   0     0     1     0     0     0     0     0     0     0     0     0     0
##   1     1     7     2     3     1     2     2     1     1     3     1     2
##
##        39    40    42    44    46    47    48    49    50    51    54    55
##   0     0     0     0     0     0     0     0     0     0     0     0     0
##   1     2     2     1     1     1     1     1     1     1     1     2     3
##
##        56    60    62    63    66    82    84    96   102   113   117   122
##   0     0     0     0     0     0     0     0     0     0     0     0     0
##   1     1     1     2     1     1     1     1     1     1     1     2     1
##
##       123   185   202   244
##   0     0     0     0     0
##   1     1     1     1     1
```

Anche questa matrice presenta numerosi 0, e quindi c'è ancora quasi perfetta separation. Il modello senza *productsListed* non viene ancora stimato e il test LRT ha ancora valori molto elevati per *daysSinceLastLogin* (1906.62) e *socialNbFollowers* (1412.34), quindi proviamo ad osservarne i plot:



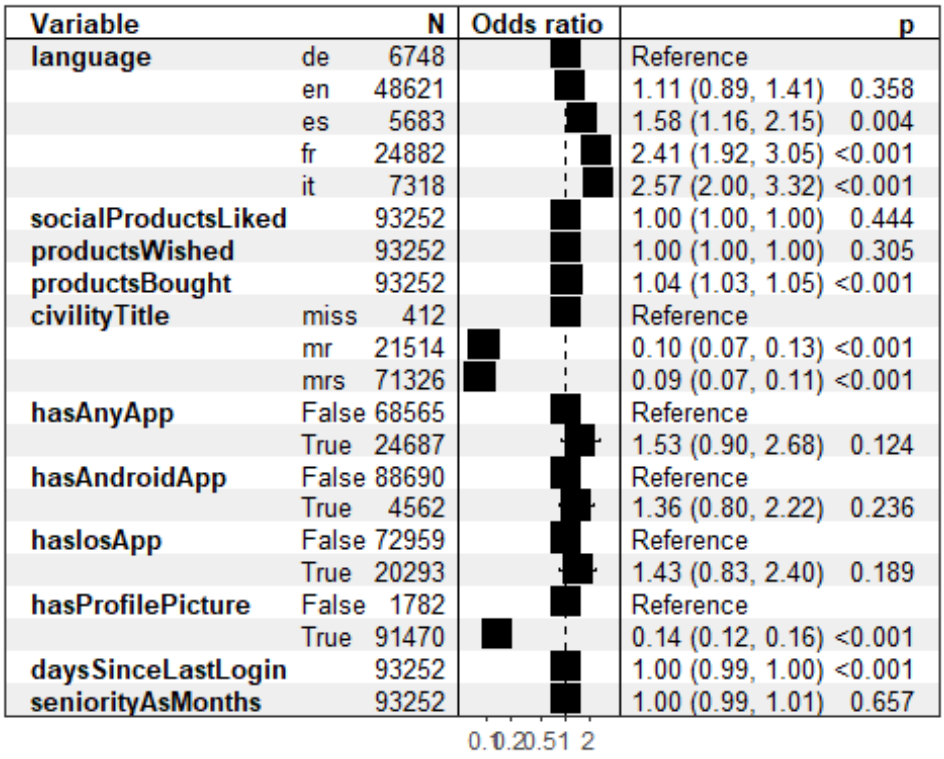
Non notiamo differenze significative.



Questo grafico, differente dal precedente, evidenzia come a valori alti del numero di followers corrispondano soggetti appartenenti alla categoria dei venditori. Proviamo quindi a rimuovere questa variabile.

```
## Call:
## glm(formula = venditore ~ language + socialProductsLiked + productsWished +
##     productsBought + civilityTitle + hasAnyApp + hasAndroidApp +
##     hasIosApp + hasProfilePicture + daysSinceLastLogin + seniorityAsMonths,
##     family = binomial, data = b)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6167  -0.1292  -0.0874  -0.0598   3.5959
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            1.259e+00  5.124e-01   2.456  0.01403 *
## languageen             1.078e-01  1.174e-01   0.919  0.35828
## languagees             4.589e-01  1.573e-01   2.918  0.00353 **
## languagefr             8.777e-01  1.179e-01   7.444 9.76e-14 ***
## languageit             9.430e-01  1.295e-01   7.284 3.23e-13 ***
## socialProductsLiked    4.316e-05  5.633e-05   0.766  0.44355
## productsWished        -4.366e-04  4.257e-04  -1.026  0.30505
## productsBought         4.100e-02  6.057e-03   6.770 1.29e-11 ***
## civilityTitlemr       -2.349e+00  1.457e-01 -16.124  < 2e-16 ***
## civilityTitlemrs      -2.464e+00  1.383e-01 -17.816  < 2e-16 ***
## hasAnyAppTrue          4.260e-01  2.770e-01   1.538  0.12402
## hasAndroidAppTrue      3.072e-01  2.593e-01   1.185  0.23606
## hasIosAppTrue          3.575e-01  2.719e-01   1.315  0.18858
## hasProfilePictureTrue -1.967e+00  6.747e-02 -29.159  < 2e-16 ***
## daysSinceLastLogin    -4.986e-03  1.092e-04 -45.651  < 2e-16 ***
## seniorityAsMonths      2.052e-03  4.625e-03   0.444  0.65725
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 18711  on 93251  degrees of freedom
## Residual deviance: 12341  on 93236  degrees of freedom
## AIC: 12373
##
## Number of Fisher Scoring iterations: 8
```

Il modello converge e viene stimato. Si tratta di un modello che presenta la variabile *venditore* in funzione di 11 covariate qualitative e quantitative, i coefficienti stimati sono 16, ma preferiamo interpretarli come incrementi (o decrementi) moltiplicativi.



Variable		N	Odds ratio	p
language	de	6748	Reference	
	en	48621	1.11 (0.89, 1.41)	0.358
	es	5683	1.58 (1.16, 2.15)	0.004
	fr	24882	2.41 (1.92, 3.05)	<0.001
	it	7318	2.57 (2.00, 3.32)	<0.001
socialProductsLiked		93252	1.00 (1.00, 1.00)	0.444
productsWished		93252	1.00 (1.00, 1.00)	0.305
productsBought		93252	1.04 (1.03, 1.05)	<0.001
civilityTitle	miss	412	Reference	
	mr	21514	0.10 (0.07, 0.13)	<0.001
	mrs	71326	0.09 (0.07, 0.11)	<0.001
hasAnyApp	False	68565	Reference	
	True	24687	1.53 (0.90, 2.68)	0.124
hasAndroidApp	False	88690	Reference	
	True	4562	1.36 (0.80, 2.22)	0.236
haslosApp	False	72959	Reference	
	True	20293	1.43 (0.83, 2.40)	0.189
hasProfilePicture	False	1782	Reference	
	True	91470	0.14 (0.12, 0.16)	<0.001
daysSinceLastLogin		93252	1.00 (0.99, 1.00)	<0.001
seniorityAsMonths		93252	1.00 (0.99, 1.01)	0.657

Il plot riproduce tutti i diversi odds calcolati sul modello completo. Notiamo alcune attitudini dirette, ma non particolarmente elevate, prossime ad 1. Ad esempio, l'attitudine ad essere venditore, comprando 2 prodotti è 1.04 volte l'attitudine ad essere venditore comprando un solo prodotto. Invece è presente attitudine inversa tra l'avere il titolo di mr o mrs e l'essere venditore rispetto al livello di riferimento di essere miss. L'odds più elevato è relativo alla lingua italiana, che risulta essere a parità delle altre condizioni, la più associata allo stato di venditore rispetto alle altre lingue.

```
## [1] 0.3404397
```

Il modello spiega il 34% della variabilità totale.

Accuracy

```
##          predicted
## observed     0     1
##        0 91115   217
##        1  1653   267
```

Creiamo un'ulteriore variabile dummy: *venditore2*. Questa assume valore 1 se i valori fittati sono superiori a 0.5, 0 diversamente, secondo cui è possibile classificare in due gruppi i soggetti. Confrontiamo tramite la funzione table in valori percentuali la classificazione dei soggetti secondo il modello stimato e la reale condizione di essere venditori o meno.

```
##          predicted
## observed           0           1
##        0 0.977083601 0.002327028
##        1 0.017726161 0.002863209
```

La maggior parte dei soggetti sono stati classificati correttamente, ovvero quasi il 98% degli utenti non è venditore e viene classificato come tale dal modello.

```
## [1] 0.9799468
```

Il valore della precisione è molto alto, ma in questo contesto era prevedibile data l'elevata presenza di zeri nel dataset. Il modello risulta comunque migliore del modello 'naive' che stima ogni osservazione come non venditore, questo infatti avrebbe come accuracy: 91332/93252=0.9794106.