

GBM per prevedere il Churn Telco

Mariangela Tafuri, Vincenzo Picarelli, Paolo Simari

Master Data Science, UNICAL

6 Luglio 2022

Introduzione

Il fenomeno “Churn” inteso come tasso di abbandono è la percentuale di clienti o abbonati che smettono di usare i servizi offerti da una azienda per un periodo di tempo.

Per ovviare al fenomeno, si vuole avviare una nuova strategia di marketing per una azienda in ambito Telco che faccia diminuire il tasso di *churn*.

Scopo di questo progetto, quindi, è analizzare il fenomeno con i dati a disposizione e prevedere il comportamento di abbandono di clienti per sviluppare la strategia di marketing appropriata.

Per tale scopo si utilizza come approccio Machine Learning l’algoritmo *Gradient Boosting Machine* (GBM).

Introduzione - Dataset

Il dataset è composto da 7032 osservazioni per 20 variabili.

Esso include le informazioni dei clienti:

- Clienti che hanno disdetto il contratto nell'ultimo mese, variabile dipendente “Churn” di tipo binario.
- Servizi che il cliente aveva attivato nel suo contratto (telefono, internet, streaming TV. . .)
- Informazioni relative al contratto del cliente: da quanto era cliente, tipo di contratto, tipo di pagamento, prezzo per mese, ecc. . .
- Informazioni demografiche del cliente: sesso, età, partner.

Introduzione - GBM

L'idea del GBM si basa sul concetto che il miglior modello possibile è quello che, combinato con modelli precedenti, impara da questi ultimi e minimizza l'errore di previsione generale.

Il GBM coinvolge tre elementi:

1. una *loss function* da ottimizzare, il cui tipo dipende dal problema che si vuole risolvere (regressione, classificazione. . .);
2. un agente che impara lentamente, come un albero decisionale;
3. un modello che si adatti e che impari dagli agenti precedenti per minimizzare la *loss function*.

Introduzione - GBM

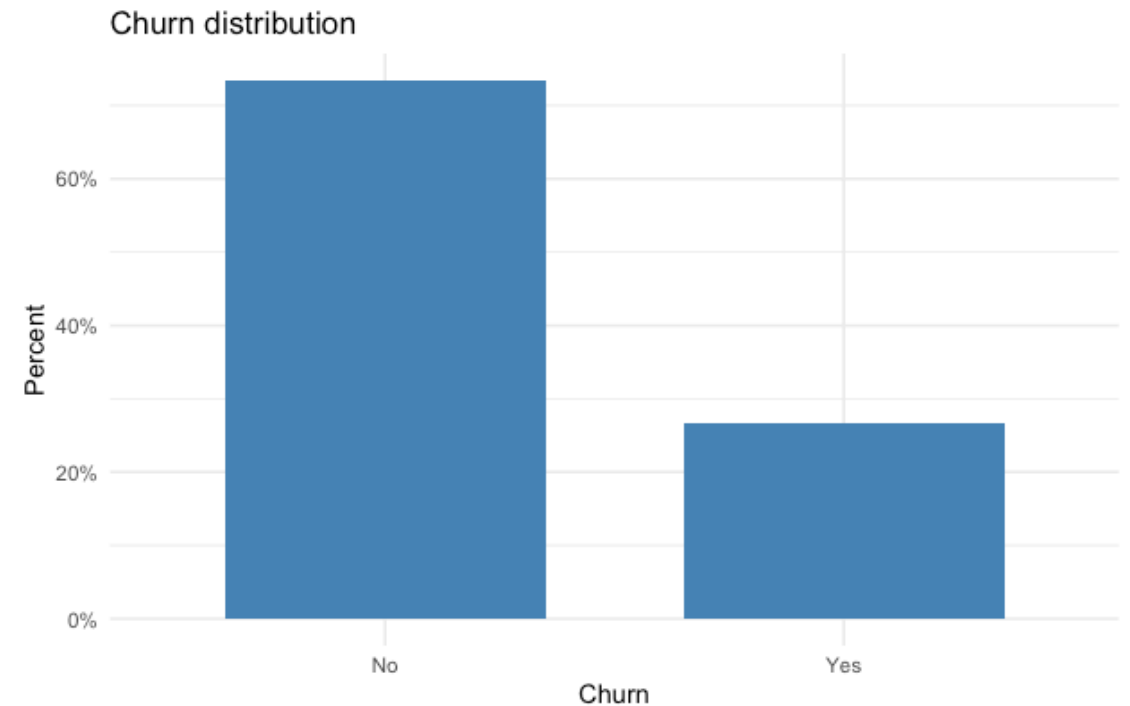
I parametri di ottimizzazione che si usano per il GBM e che possono essere stabiliti preliminarmente attraverso *cross validation*, sono:

1. il numero di alberi che contiene il GBM;
2. il parametro di apprendimento (*shrinkage*) controlla il grado di apprendimento dell'algoritmo. Di solito assume valore 0.1 o 0.01, in base al problema. Il valore più piccolo viene utilizzato quando si vogliono raggiungere performance migliori;
3. il numero di split controlla la complessità degli alberi;
4. numero minimo di osservazioni per nodo.

Data preprocessing

Tale fase comprende:

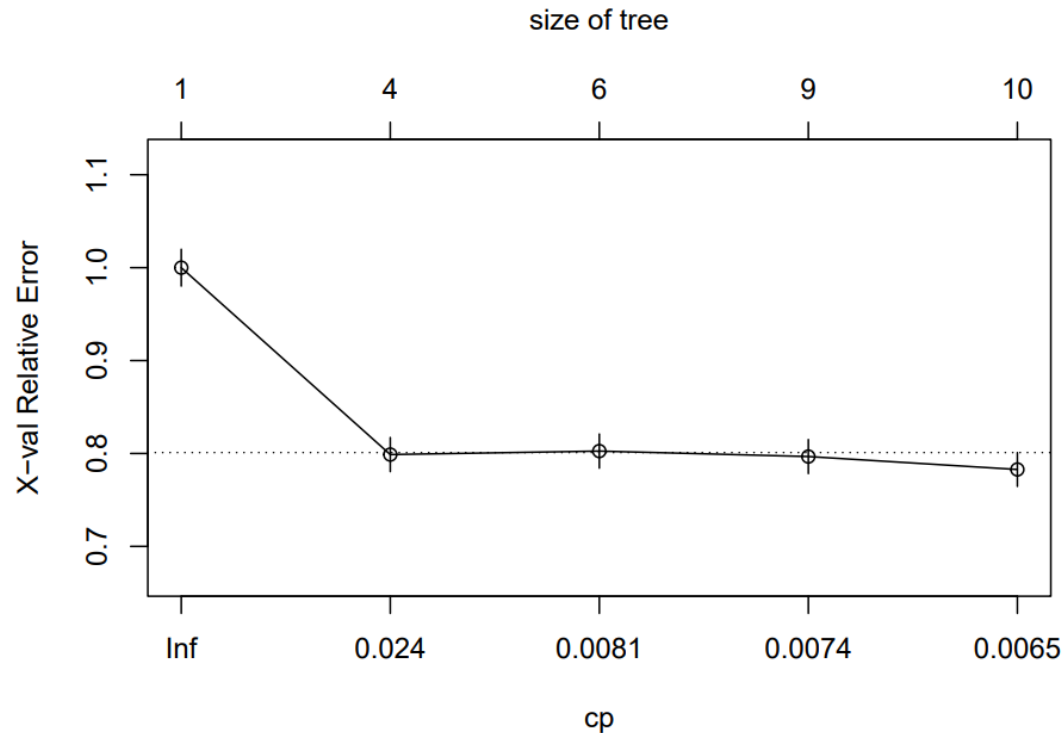
- *Summary* delle variabili;
- Controllo degli NA (0.16 % delle osservazioni che contiene *missing* viene eliminato);
- Standardizzazione delle variabili numeriche;
- Aggiustamento del tipo di variabili (variabili categoriche che erano tipizzate come numeriche)
- Stabilire le variabili indipendenti e quella dipendente.



La distribuzione della variabile dipendente è del 73.4% “No” e 26.6% “Yes”.

Si ricorda che lo scopo è predire i “Yes”, cioè i clienti che abbandonano.

Fase di adattamento



Adattamento di un albero di classificazione

Per stabilire il numero ideale di nodi terminali, si adatta un albero di classificazione su tutto il dataset e si sceglie il numero di nodi secondo le *cross validation*. Si stabilisce un valore di controllo del parametro di complessità a 0.006 tale per cui, si cerca di costruire un albero saturo.

In base ai valori del parametro di complessità e alla stima di *cross validation*, si ritiene opportuno considerare negli iperparametri un numero di nodi terminali pari a 3, 5 oppure 8.

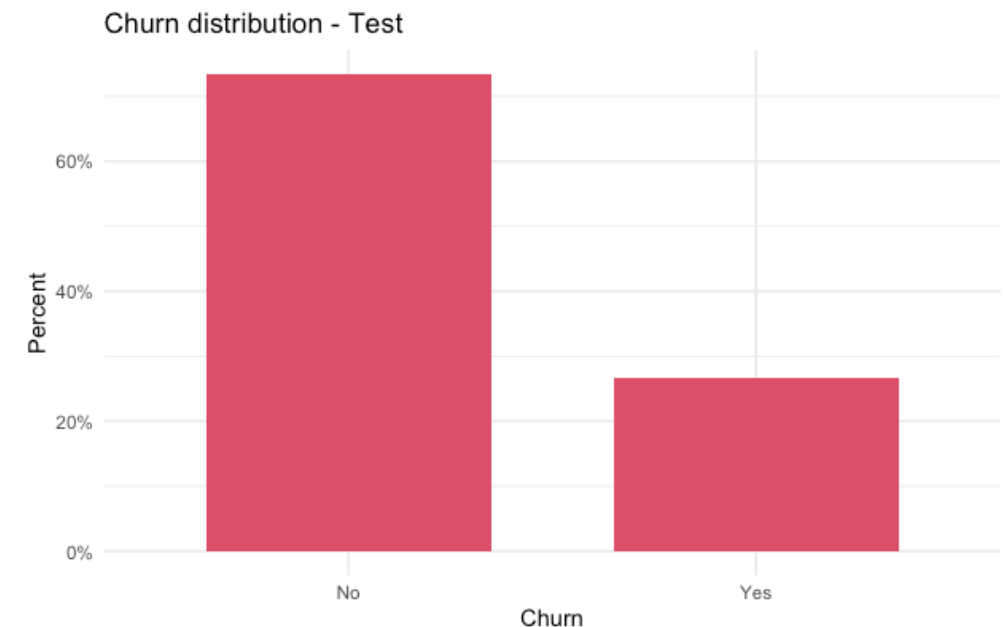
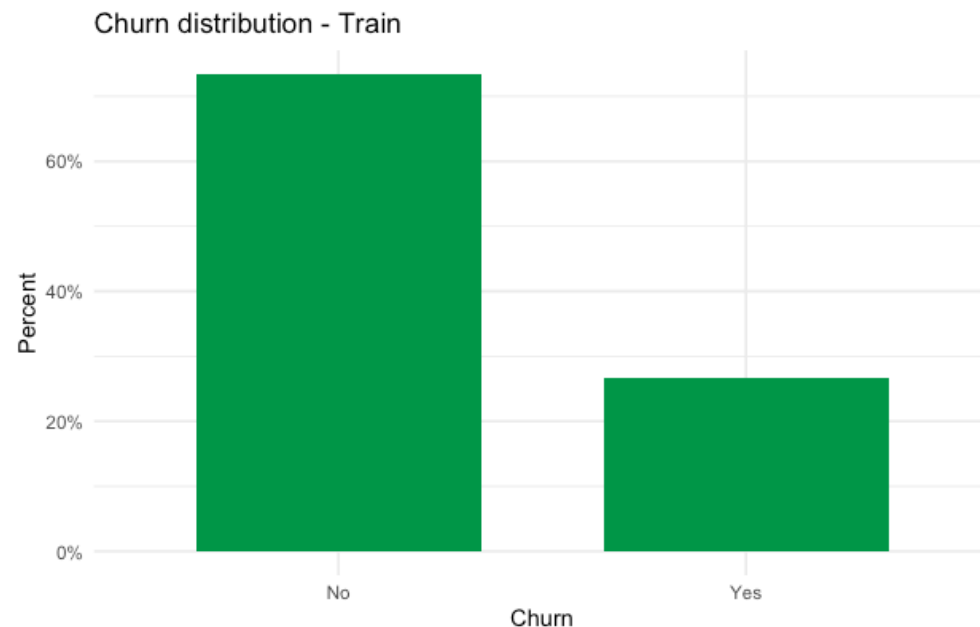
Fase di adattamento

Fase di *train*

Si divide il dataset in train-set e test-set.

Il train rappresenta il 70% del dataset totale e i clienti scelti a caso nel train vengono selezionati senza ripetizione.

Si nota che è mantenuta la giusta proporzione della variabile dipendente nei due dataset.



Fase di adattamento

GBM 1

Si istanzia un classificatore GBM con il metodo della *repeated cross validation*. Nella funzione di controllo, vengono stabiliti il numero di *fold* (10), il numero di ripetizione della *cross validation* e la funzione di tipo classificazione.

Nella fase di *train* si stabilisce la funzione, il metodo (gbm), la metrica (ROC) e la funzione di controllo.

```
# Set up control function for training

ctrl <- trainControl(method = "repeatedcv",
                     number = 10, # 10 fold
                     repeats = 5, # x validation viene ripetuta 5 volte
                     summaryFunction = twoClassSummary,
                     classProbs = TRUE)

# Build a standard classifier using a gradient boosted machine

train$y <- as.factor(train$y)
test$y <- as.factor(test$y)

set.seed(3)
GBM_1 <- train(y ~ .,
               data = train,
               method = "gbm",
               verbose = FALSE,
               metric = "ROC",
               trControl = ctrl)
```

Fase di adattamento

GBM 1 – *best tune*

Il miglior modello GBM 1 è composto da 150 alberi, una profondità massima di 2 *split*, un numero minimo di osservazioni in ciascun nodo terminale pari a 10 e un valore di *shrinkage* 0.1 di default.

```
GBM_1$bestTune # miglior modello GBM
```

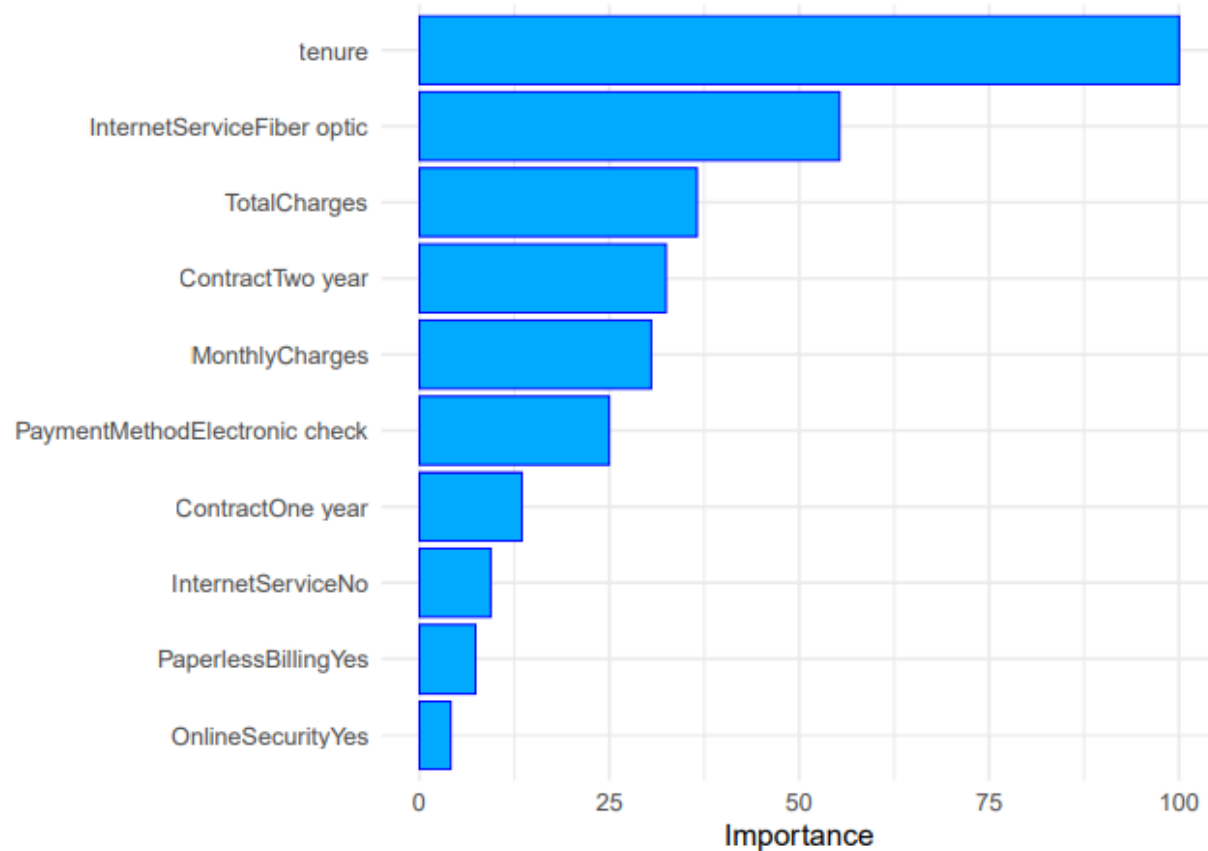
```
##   n.trees interaction.depth shrinkage n.minobsinnode  
## 6      150                2        0.1             10
```

Fase di test

GBM 1 – *performance*

GBM 1 sul dataset di test ha un livello di *accuracy* pari a 0.80 e di *precision* pari a 0.65.

Di seguito viene riportata la *feature importance* per ciascuna variabile. Viene determinata calcolando la relativa influenza di ciascuna variabile: se quella variabile è stata selezionata per la suddivisione durante il processo di costruzione dell'albero e quanto l'errore al quadrato (su tutti gli alberi) è migliorato (diminuito) di conseguenza.



Fase di adattamento

GBM 2

Siccome si vuole predire bene il verificarsi della disdetta del contratto (Churn = Yes), che in questo dataset ha una minore probabilità di verificarsi (0.26), si cerca di aumentare la precisione del modello.

Per tanto, si stabiliscono degli iperparametri: - numero di nodi terminali (3, 5, 8), - numero di alberi presenti nel modello da 50 a 1500 con passo 50. Inoltre si stabilisce un tasso di *learning rate* pari a 0.01 e un numero minimo di osservazioni all'interno dei nodi terminali pari a 20.

La combinazione tra questi genera 90 modelli differenti, tra cui si sceglie quello che ha la ROC più alta.

```
# Set-up 10-fold cross-validation
ctrl_grid <- trainControl(method='cv',
                          number=10,
                          search="grid",
                          summaryFunction = twoClassSummary,
                          classProbs = TRUE)

# Generate the parameters' grid
gbmGrid <- expand.grid(interaction.depth = c(3, 5, 8),
                      n.trees = (1:30)*50,
                      shrinkage = 0.01,
                      n.minobsinnode = 20)
```

```
# Fit the model over the grid
set.seed(3)

GBM_2 <- train(y ~ .,
               data = data,
               method = "gbm",
               metric='ROC',
               trControl = ctrl_grid,
               verbose = FALSE,
               tuneGrid = gbmGrid)
```

Fase di adattamento

GBM 2 – *best tune*

Il miglior modello GBM 2 è composto da 700 alberi, una profondità massima di 5, un numero di osservazioni in ciascun nodo terminale pari a 20 e un valore di *shrinkage* 0.01. Si nota che in base al parametro di learning rate più basso il modello impara più lentamente, infatti il numero di alberi necessari per costruire il modello ottimale è maggiore.

```
GBM_2$bestTune
```

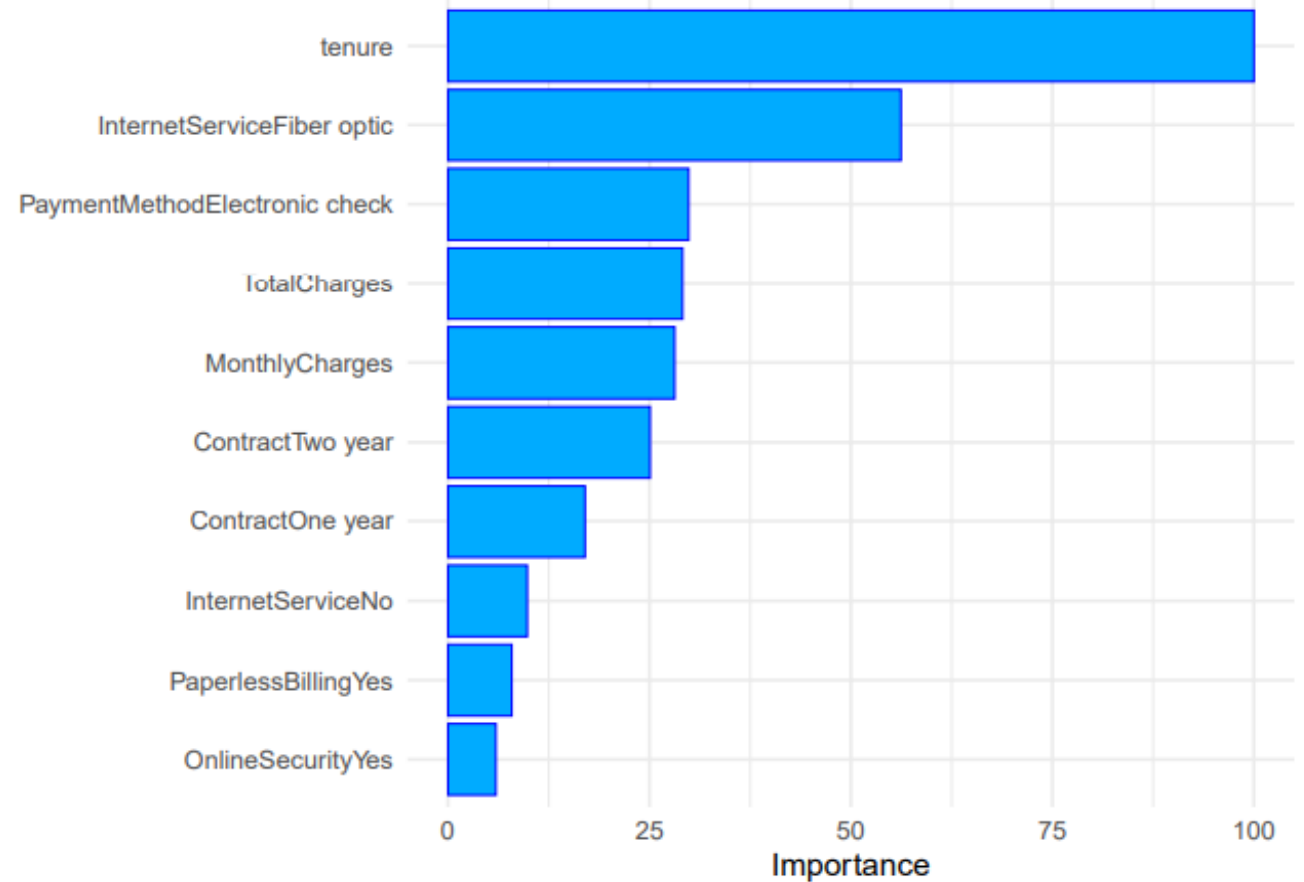
```
##      n.trees interaction.depth shrinkage n.minobsinnode  
## 44      700                5      0.01                20
```

Fase di test

GBM 2 – *performance*

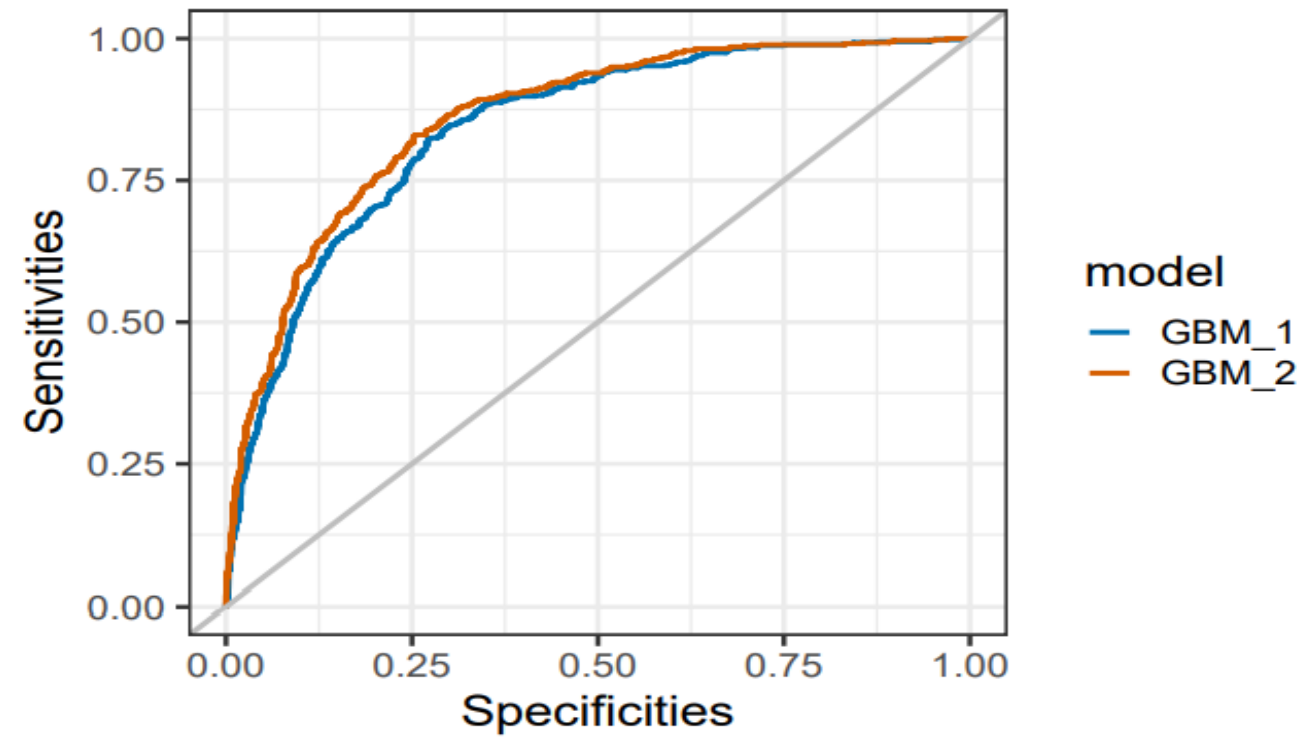
Il GBM 2 sul dataset di test ha un livello di *accuracy* pari a 0.81 e di *precision* pari a 0.69. Si riesce a predire l'abbandono del contratto per il 4% delle volte in più rispetto al GBM 1.

Di seguito viene riportata la *feature importance*:



ROC Curve

```
## $GBM_1  
## Area under the curve: 0.8415  
##  
## $GBM_2  
## Area under the curve: 0.8595
```



Conclusioni

In base ai due metodi di costruzione dei modelli GBM, si nota che quello che apprende più lentamente (GBM 2) performa meglio, in termini di precisione.

In questo caso, sulla base della precisione, cioè il tasso dei *true positive* sul totale dei *positive*, si decide quale metodo sia migliore. Il GBM 1 registra una precisione di 0.65; il GBM 2 registra una precisione di 0.69.

Nonostante la lentezza di computerizzazione nell'applicare un metodo più complesso, in base alla domanda iniziale, cioè predire con maggiore precisione i *churn* in un dataset abbastanza sbilanciato, è meglio investire in un modello che è in termini di computerizzazione e di struttura più complesso per avere una maggiore precisione previsionale.

Conclusioni

Per rispondere alla domanda, la strategia di marketing consigliata è quella di concentrarsi sulla “*tenure*”, cioè da quanti mesi il cliente è abbonato.

Un’idea potrebbe essere quella di premiare l’anzianità dei clienti con codici sconto o agevolazioni sul contratto.

Grazie per l'attenzione!

Mariangela Tafuri, Vincenzo Picarelli, Paolo Simari

Fonte dati: [Telco Customer Churn](#)