PROGETTO DATA MINING

di Millone A., Rossi S., Zanini V. ,Pedrini A.

Contesto

Il dataset scelto proviene dalla US Small Business Administration (US SBA), contiene 89000 osservazioni analizzate su 27 variabili che descrivono determinate caratteristiche informative ed economiche relative a piccole e medie imprese.

La US SBA è stata fondata nel 1953 sul principio di promuovere e assistere le piccole imprese nel mercato del credito statunitense. Le piccole imprese sono state una fonte primaria di creazione di posti di lavoro negli Stati Uniti; pertanto, favorire la formazione e la crescita di piccole imprese ha benefici sociali creando opportunità di lavoro e riducendo la disoccupazione. Ci sono state molte storie di successo di start-up che hanno ricevuto garanzie sui prestiti SBA come FedEx e Apple Computer. Tuttavia, ci sono state anche storie di piccole imprese e/o start-up che sono in default sui loro prestiti garantiti dall'SBA.

Le 27 variabili presenti nel dataset US SBA sono:

1. **LoanNr_ChkDgt: Identificatore Chiave primaria**
2. **Name:Nome del mutuatario**
3. **City: Città mutuataria**
4. **State: Stato mutuatario**
5. **Zip code: CAP del mutuatario**
6. **Bank: nome della banca**
7. **BankStatE: Stato della banca**
8. **NAICS: Codice del sist. di classificazione dell'industria**
9. **ApprovalDate: Data Impegno SBA emesso**
10. **ApprovalFY: Anno fiscale di impegno**
11. **Term: Durata del prestito in mesi**
12. **NoEmp: Numero di dipendenti aziendali**
13. **NewExist: Attività 1 = esistente, 2 = nuova**
14. **CreateJob: Numero di posti di lavoro creati**
15. **RetainedJob: Numero di posti di lavoro mantenuti**
16. **FranchiseCode: Franchising/nessun Franchising**
17. **UrbanRural: Urbano Rurale**
18. **RevLineCr: Linea di credito Si No**
19. **LowDoc: Proggramma di Prestito**
20. **ChgOffDate: La data in cui un prestito viene dichiarato inadempiente**
21. **DisbursementDate: Data di erogazione**
22. **DisbursementGross: Importo erogato**
23. **BalanceGross: Importo lordo in sospeso**
24. **MIS_Status: Stato del prestito addebitato→ *Variabile Target***
25. **ChgOffPrinGr: Importo addebitato**
26. **GrAppv: Importo lordo del prestito approvato dalla banca**
27. **SBA_Appv: Importo garantito di SBA del prestito approvato**

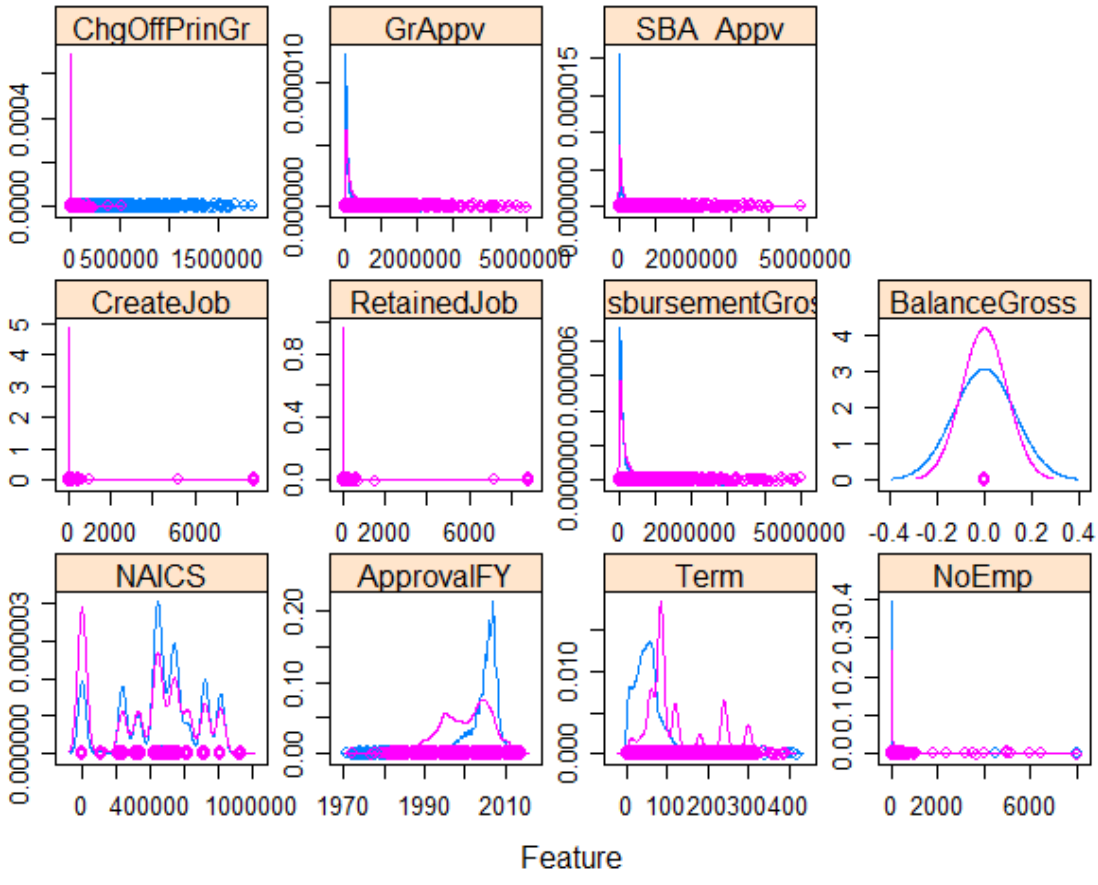
Il nostro obiettivo è individuare il miglior modello classificatore che identifichi correttamente le aziende che sono cattive pagatrici e non pagatrici (corretta classificazione del target: prestito restituito o no), che permetta poi di classificare correttamente nuove osservazioni.
Il target del modello sarà la variabile binaria MIS_Status, che rappresenta lo Stato del prestito addebitato (Non Pagato= Chgoff, Pagato per intero = Pif)

Prima di svolgere l'analisi procediamo ad eliminare le seguenti variabili:
RevLineCr (Linea di credito revolving) *Chgoffdate* (data in cui un prestito viene dichiarato inadempiente), *Disurbementdate* (Data di erogazione), in quanto non sono discriminanti/significative per la nostra analisi.
Si decide di togliere, anche la variabile *Chgoffpringr* in quanto si tratta di una variabile target "mascherata" (covariata che fa il ruolo della target), poiché rappresentava l'ammontare del debito e per tutti i soggetti che avevano restituito il prestito, aveva come valore 0.

Dalle statistiche descrittive e dalle distribuzioni di frequenza si nota che le variabili numeriche sono tutte distribuite asimmetricamente: *createjob, retainjob, NoEmp* (n° di posti di lavoro creati o mantenuti, numero di dipendenti aziendali), *disurbementgross e GrAppv e SBA_Appv*, e hanno range molto grandi. Anche la variabile *Term* (durata del prestito in mesi) risulta asimmetrica. le altre variabili numeriche sono binarie e le rimanenti sono categoriche

	min	max	mediana	media	3° quartile
Disurbementgross	4000	5.000.000	100.000	201.202	238.000
GrAppv	1000	5.000.000	90.000	192.900	225.000
SBA-appv	500	4.869.000	60.960	149.890	175.00
create job	0	8.800	0	9	1
retainjob	0	8.800	1	12	4
NoEmp	0	8.000	4	11	10
Term	0	421	84	111	120

Osserviamo le distribuzioni univariate rispetto alle classi del target (rosa=pagatore, blu= non pagatore). Notiamo che solo la variabile Term (durata del prestito in mesi) sembra discriminare le osservazioni tra le classi del target. Altre variabili come ad esempio *createjob, retainjob, NoEmp* al contrario non discriminano tra le classi del target.

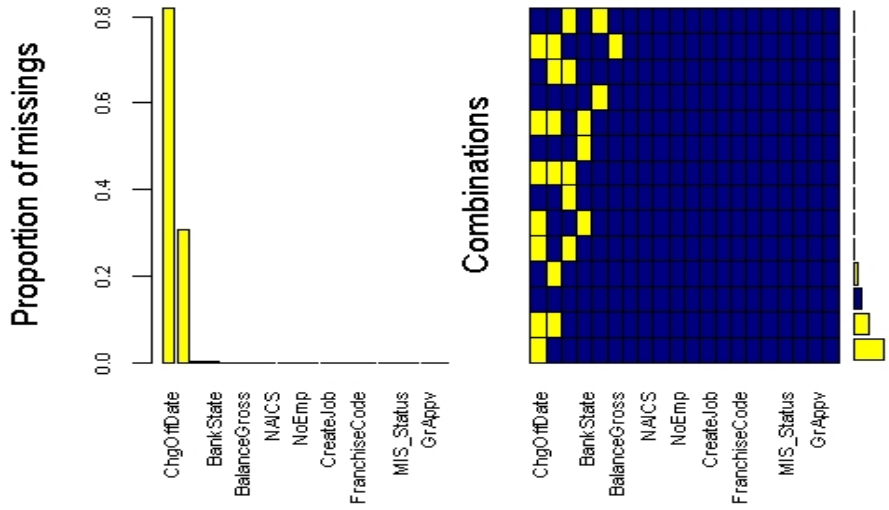


Alfine di classificare in modo corretto i soggetti come “pagatori” e “non pagatori” è necessario svolgere i seguenti passaggi :

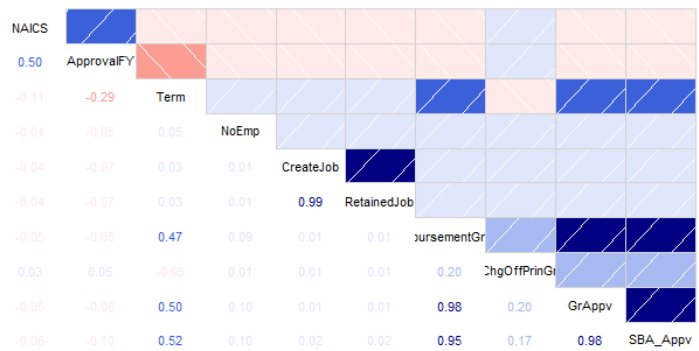
1. **Preprocessing e Training dataset** → si fanno tanti modelli, stiamo tunando il modello
2. **Validation dataset** → si valutano performance del modello su dati indipendenti (metriche di assesment, metriche robuste, cfr stat. comp.); →creazione di modelli classificativi
3. **Valutazione performance della classificazione del modello migliore** dello step 2; Applicare il miglior modello ai dati nuovi.-→con modello migliore decidere soglia
4. Utilizzare la nuova soglia nel dt **score** ai fini di avere la massima classificazione possibile.

Preprocessing: Analisi dati mancanti

Dai grafici sottostanti si può osservare che le variabili *RevLineCr* (Linea di credito revolving) e *Chgoffdate* (data in cui un prestito viene dichiarato inadempiente) sono caratterizzate da molti dati mancanti (entrambe erano già state eliminate perché non significative), inoltre si osserva che la variabile *Balancegross* (che rappresenta importo lordo in sospeso) ha varianza nulla 0, cioè 0 variance (ha tutti valori numerici 0=nessun lordo in sospeso) quindi viene eliminata perché creerebbe problemi alla nostra analisi.



Preprocessing: Collinearità



A fianco il grafico delle correlazioni tra le variabili numeriche. Dalla matrice si evidenziano forti correlazioni bivariate tra *create job* e *retainjob* (cioè tra numeri di posti lavoro creati e numero di lavori mantenuti), e tra *disurbementgross* e GrAppv e SBA_Appv (importo erogato, importo approvato dalla banca e garantito da SBA).
Queste variabili certamente sono collineari.

Esite un buon legame anche tra la variabile *Term* (durata del prestito in mesi) e le variabili *disurbementgross*, *GrAppv e SBA_Appv* (importo erogato, importo approvato dalla banca e garantito da SBA), come da attese infatti al crescere dell’importo erogato, approvato o garantito cresce anche la durata del prestito.

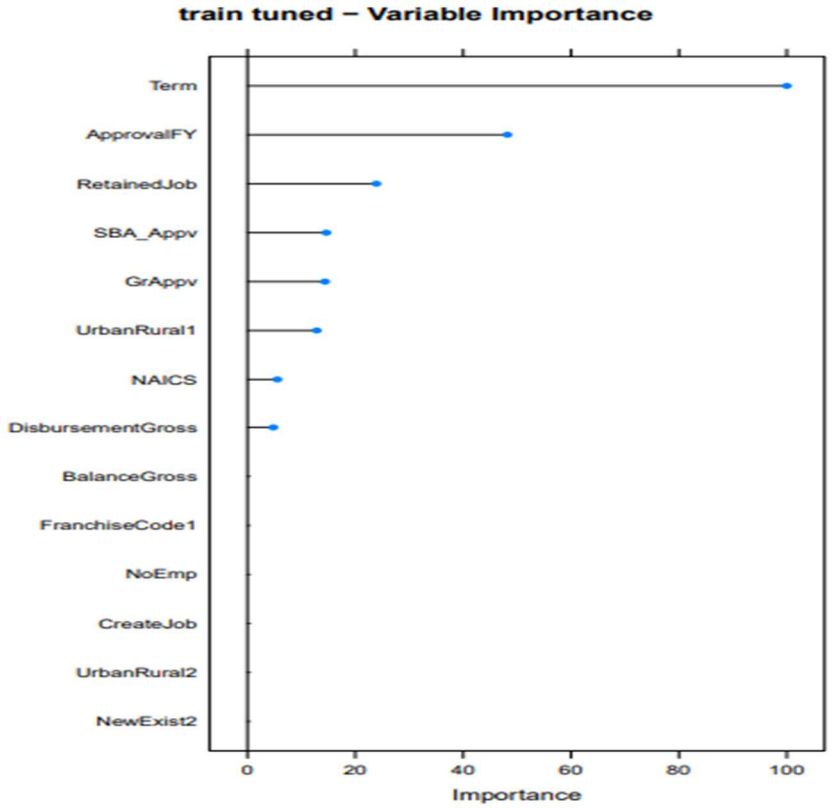
Prima di iniziare un progetto di data mining è utile controllare se dt è rappresentativo del contesto in esame ed eventualmente fare aggiustamenti con le prior; in un contesto economico come quello del credito,in cui vengono erogati prestiti a piccole medie imprese, la percentuale dei non pagatori normalmente si aggira intorno al 20 % e i pagatori sono circa l'80%; nel nostro caso i non pagatori risultano essere il 17,6% e i pagatori l' 82,4%, quindi non dobbiamo fare aggiustamenti con le prior.

Dopo aver tolto variabili inutili o confondenti e verificato che dt è rappresentativo, svolgiamo l'imputazione per i dati mancanti con il metodo bagging, che è un metodo robusto che prende ogni covariata con dati mancanti e la pone come target, rilancia una serie di alberi per cui alla fine il valore imputato sarà il valore previsto del target di questa variabile
Dopo questi passaggi dividiamo il dataset in dati di traininig (60%) e dati di test (40%)

STEP 1: COSTRUZIONE dei MODELLI DI CLASSIFICATIVI

Per selezionare le covariate si usa un metodo non parametrico, un modello ad albero (*tecnica machine learning albero*), *cioè* un metodo non analitico che permette di individuare le variabili più importanti ed eliminare eventualmente il problema della collinearità e 0 variance.
Una Variabile è importante, quando la sua importanza è data dalla somma dei $\Delta Gini$ relativi a lei, cioè la somma dei decrementi "impurità" che la variabile genera nei vari piani dell'albero.

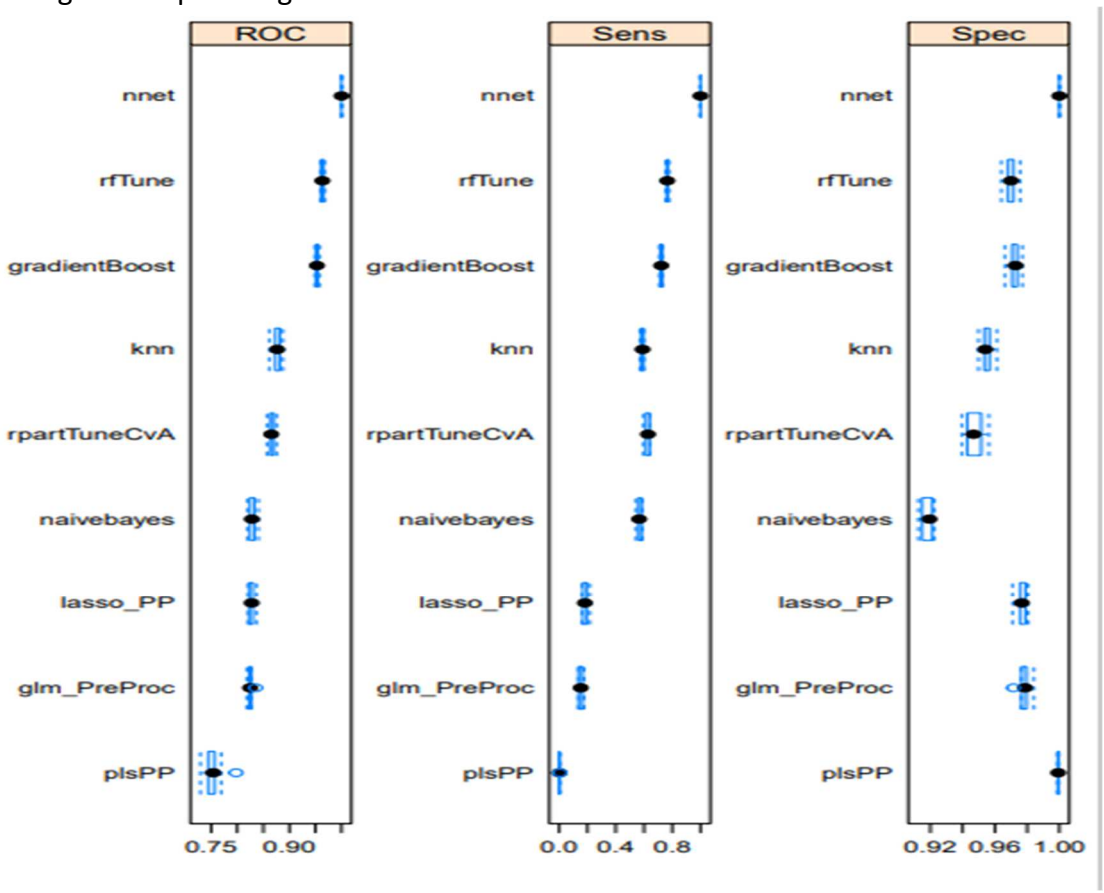
Il grafico mostra le variabili importanti :



Con questa procedura abbiamo selezionato le variabili importanti, prendendo queste variabili creeremo un nuovo dt di training che servirà per i modelli non robusti.

Adesso procediamo a svolgere i vari algoritmi di classificazione, ogni algoritmo è stato sviluppato con la tecnica CROSS VALIDATION a 10 Fold ;questa procedura consiste nel dividere il dt in uso in 10 parti uguali: 9 parti saranno utilizzate come training (quindi per far applicare il modello) e 1 parte come validation test, in quanto visualizzerà i valori previsti del target di tutte le 9 Fold, in questo modo si ottengono valori previsti del target non distorti/robusti. Per tunare i modelli si utilizza la metrica ROC.

I modelli applicati sono: Logistico, Pls, Lasso, Albero (all'inizio), Random Forest , Gradient Boosting, Reti Neurali, Knn e Naive Bayes.
Qui si seguito si riporta il grafico BW PLOT con metriche ROC SENS E SPEC.

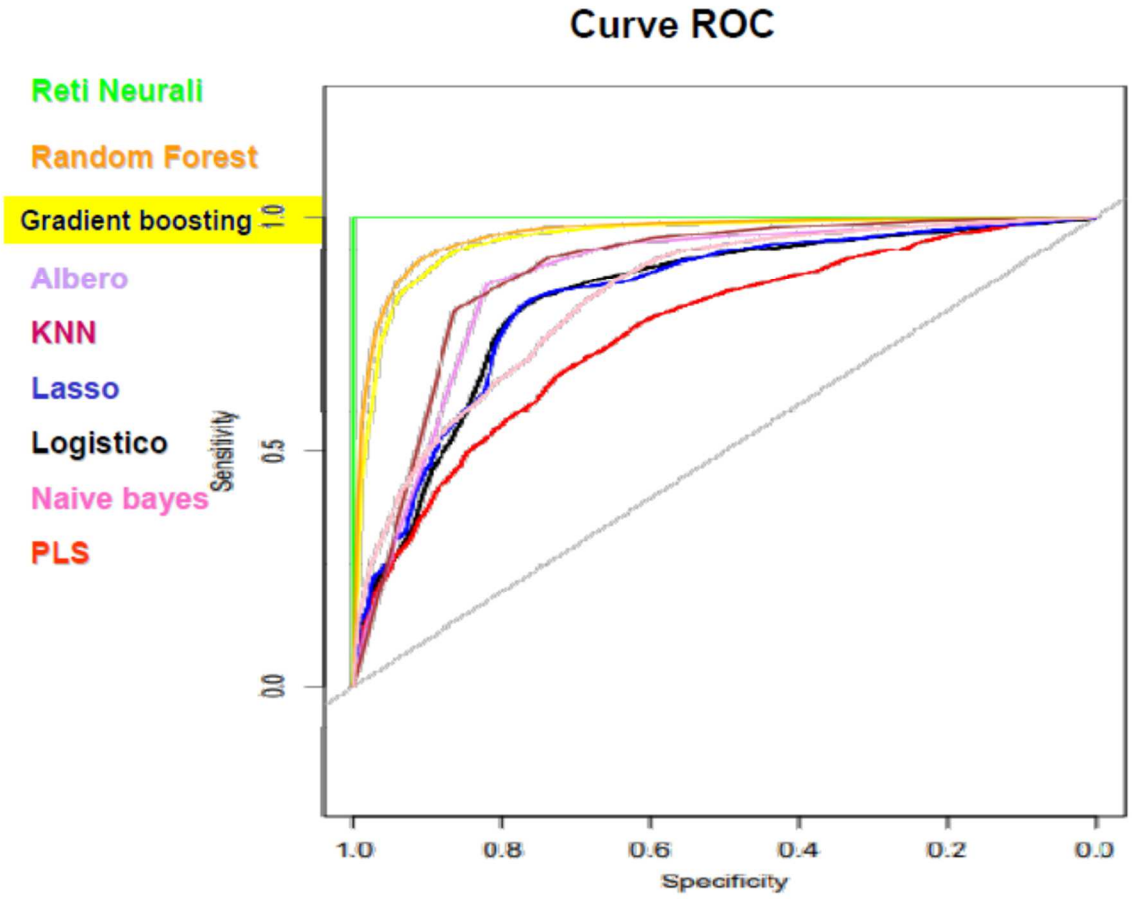


Da una prima visione di confronto dei vari modelli quelli che preformano meglio sono i modelli basatI su RETI NEURALI e ALBERI, invece quelli come PLS non ottengono in tutte le metriche performance positive.
Da notare che tutti i modelli ottengo valore di specificità superiori a .90, cioè individuano bene la quota di soggetti *non pagatori* che sono effettivamente *non pagatori* ,i **Veri Negativi** .
Mentre ottengono valori più contenuti di sensitività cioè la quota di soggetti *pagatori* che sono effettivamente *pagatori*, i **Veri Positivi**; solo i modelli alberi e rete neurale mostrano valori di sensitività superiori a 0.80.
Per valutare se ci sono differenze significative tra i modelli, è stata applicato la statitica t-test (corretta con metodo Bonferroni), da cui risulta che:

- per quanto riguarda la specificità non ci sono differenze significative tra i diversi modelli,
- il modello PLS ha sensitività significativamente più bassa rispetto ai modelli Random Forest, Gradient Boosting, Reti Neurali e Knn .

STEP 2: VALUTAZIONE DELLE PERFORMANCE CLASSIFICATIVE DEI VARI MODELLI

Dopo aver creato i modelli vediamo come questi si adattano al Validation dataset, per verificare la loro capacità classificativa, in termini di sensibilità e sensibilità, su dati indipendenti.
Sulla base dei risultati ottenuti (confusion matrix), si costruiscono dei grafici che mostrano le Curve ROC, che sono una rappresentazione grafica in base alla combinazione delle metriche sensitività e il complemento a uno della specificità.
Curve ROC: è la metrica che si preferisce perché valuta la bontà classificativa di ogni modello per tutte le possibili soglie. la curva Roc mostra come variano specificità e sensitività, coè come varia la percentuale della corretta classificazione degli eventi al variare del tasso di misclaffificazione.



Dalle curve Roc si conferma che i migliori modelli performanti sono quelli basati sulle reti neurali e alberi.
Infatti se la curva Roc cresce in modo repentino, come nel caso di Reti Neurali, Ramdom Forest e Gradient Boosting significa che c'è una buona capacità di classificazione e basso errore di misclassificazione anche all'aumentare della soglia (in breve quando la curva ROC sta sull'asse Y → modello migliore).
Quando la curva ROC sta sulla diagonale, il modello a cui corrisponde è il peggiore (vedi PLS) .
Si calcola specificity e sensitivity per ogni soglia, la ROC è la rappresentazione di questi due valori al variare della soglia. Il modello deve lavorare bene con soglie alte, allora è un buon modello.
Ogni curva Roc ha un'area sottostante chiamata AUC che rappresenta quanto il modello performa bene, più l'area è grande più performa, cioè classifica bene i nostri dati.

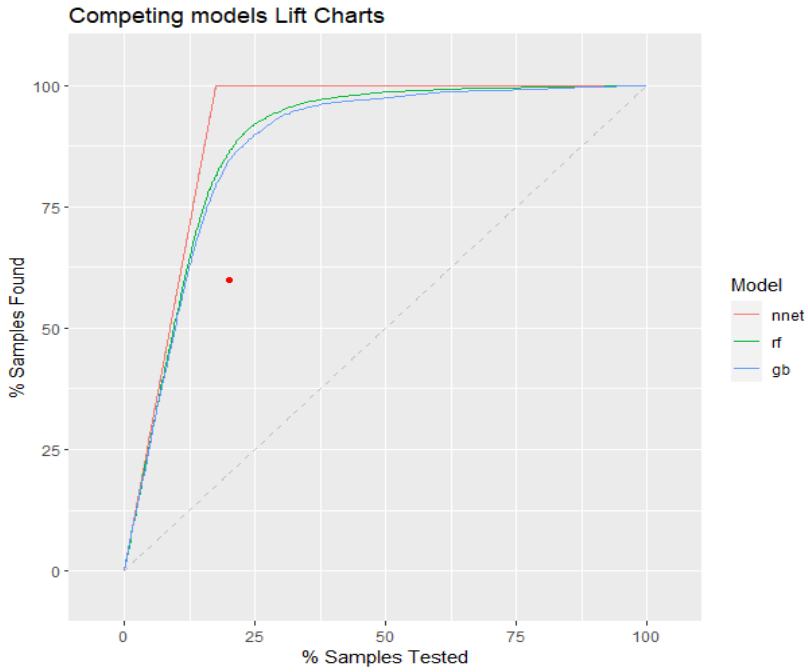
Le Curve Lift sono la metrica più interessante, è il valore cumulativo: percentuale di risposte catturate nei primi x decili. Più la curva della cumulata cresce, più il modello avrà delle ottime performance classificative. Solitamente si scelgono i primi decili.

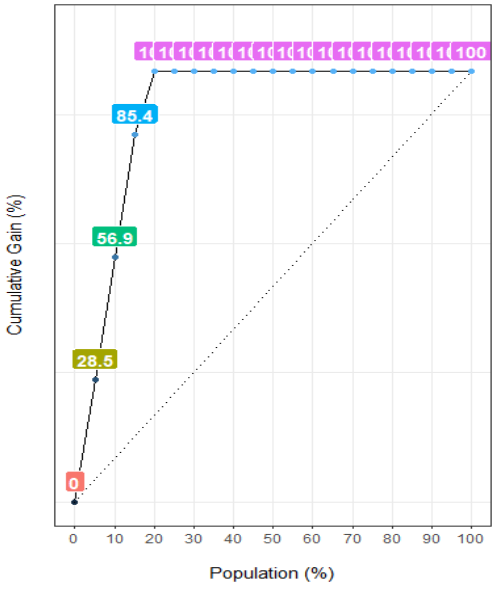
Le Curve Lift indicano la percentuale di corretti non pagatori (evento principale dello studio in generale) per ogni porzione di popolazione scelta. Il grafico qui a fianco è relativo ai modelli basati su alberi e reti neurali.

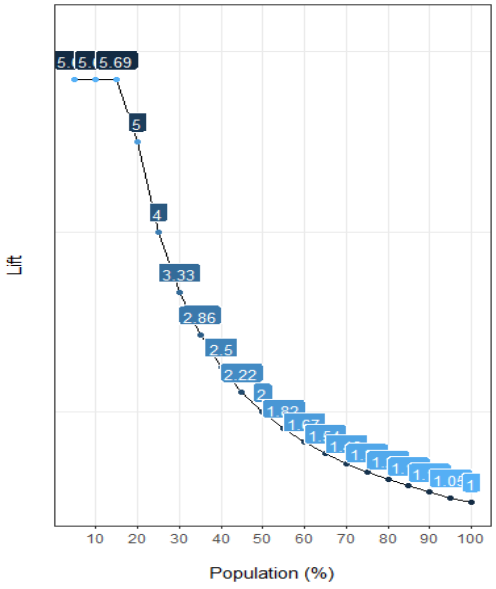
nnet=RETI NEURALI

rf=RANDOM FOREST

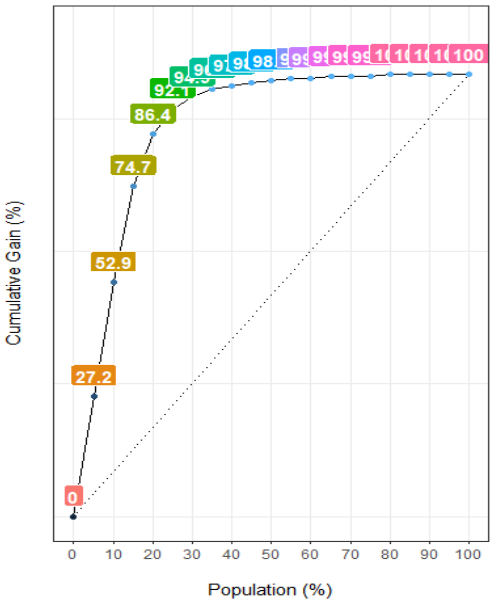
gb=GRADIENT BOOSTING

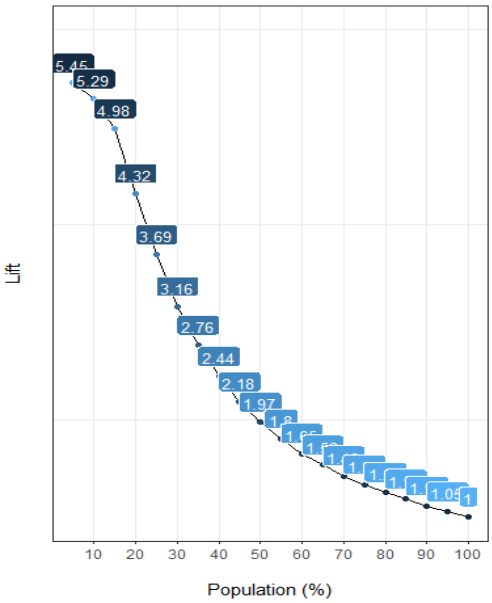




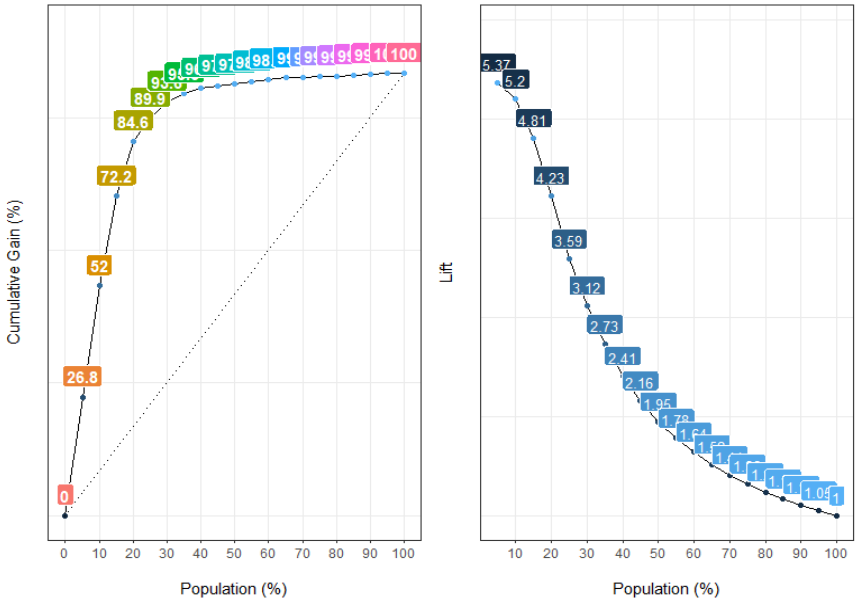


← RETI NEURALI





← RANDOM FOREST



←**GRADIENT BOOSTING**

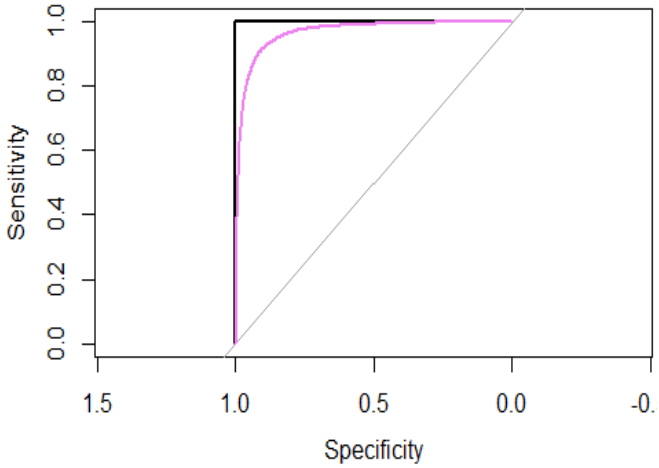
Per confrontare le curve lift e decidere il miglior modello, si prende una porzione di popolazione (secondo o terzo decile) e si vede la percentuale di Gain (% risposte catturate→veri negativi), il miglior modello sarà quello con il valore di Gain maggiore (→maggiore specificità).

Modelli	Popolazione	Gain	Score point
Reti neurali	20/100	100.00	0.0000546
Random forest	20/100	86.4	0.320
Gradient boosting	20/100	84.6	0.3116

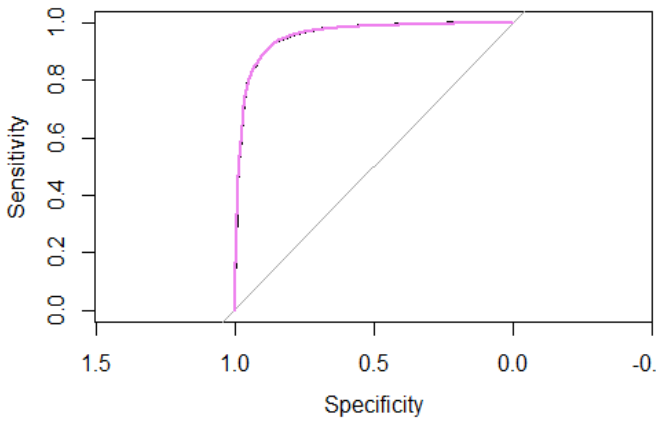
Nella tabella qui a fianco sono riportati i valori numerici delle lift relativi al 20% della popolazione. Se prendessi un modello basato sulle Reti Neurali riuscirei ad individuare il 100% dei soggetti non pagatori, la performance scenderebbe all'86.4% se utilizzassi la Random Forest e all' 84.6% se utilizzassi il Gradient Boosting.

In base ai valori delle lift, i modelli migliori sono Reti Neurali, Gradient Boosting o Random Forest.

Per i modelli Gradient Boosting o Random Forest verifico se esiste un fenomeno di overfitting: plotto le curve Roc del medesimo modello, prima su train e poi su validation test per vedere se sono uguali o diverse, se nel grafico risultano sovrapposte non c'è overfitting



Graficamente si può vedere che le curve Roc basate sul modello Random Forest non sono sovrapposte, presentano differenze in termini di distanza e hanno valori diversi. Numericamente questo si può notare anche dal valore dell'area AUC: Random Forest su dati di train mostra una AUC:1, Random Forest su dati di vtest mostra una AUC:0.965

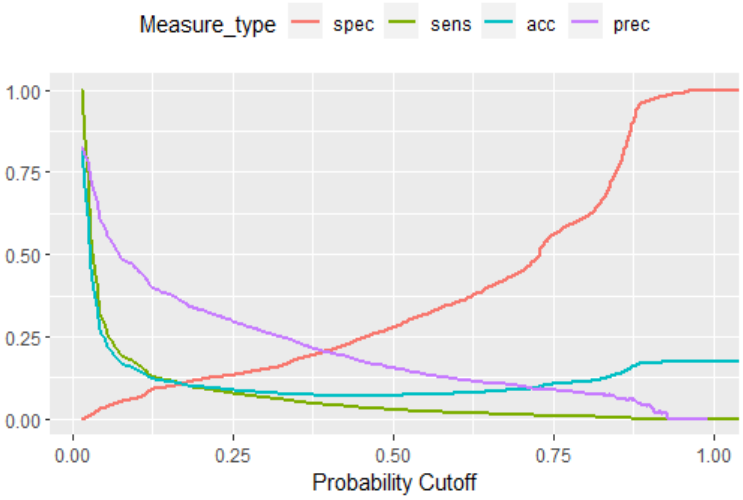


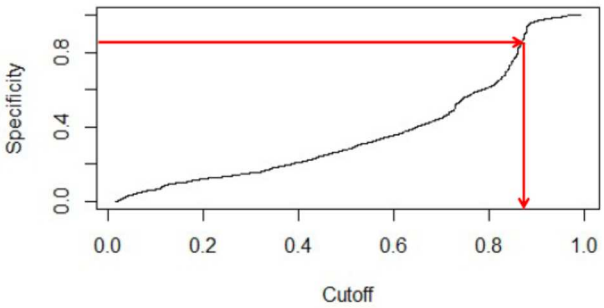
Graficamente non c'è differenza; le curve Roc basato sul modello ad Gradient Boosting sono sovrapposte,non presentano differenze in termini di distanza, hanno gli stessi valori per tutto il grafico. Numericamente questo si può notare anche dal valore dell'area AUC:
Gradient Boosting su dati di train mostra una AUC:0.954,
Gradient Boosting su dati di vtest mostra una AUC:0.955

Si decide per il modello Gradient Boosting perché non presenta il fenomeno dell'overfitting che invece è presente per il modello Random Forest.

STEP 3: VALUTARE LA PERFORMANCE CLASSIFICATIVA DEL MODELLO SCELTO (GRADIENT BOOSTING)

Ora valutiamo la soglia ottimale per il modello *Gradient Boosting*, consigliata dalle lift utilizzando Criterio Statistico, che confronta le diverse metriche su diverse soglie.
Si sceglie la soglia in modo che il modello soddisfi i nostri obiettivi, nel nostro caso si deve scegliere la soglia ottimale a cui corrisponde la migliore Specifity.
Per il modello Gradient Boosting, la soglia che ottimizza la Specificy 0.85.





Soglia scelta in base alla Specifity (0.85)

Tenendo conto della soglia (0.85) della metrica di interesse (specificità), possiamo visualizzare nella confusion matrix , come il modello Gradient Boosting classifica nelle varie celle l'algoritmo Vp,Fp,Vn,Fn (cioè come i nostri soggetti vengono classificati).

Confusion matrix relativa al validation test Gradient Boosting

	Reference	
Prediction	NO	YES
NO	1531	104
YES	4766	29436

Metrica	**Valore**
Accuracy	**.864**
Sensitività	**.243**
Specificity	**.996**

STEP 4: SCORE NUOVI DATI

Nell'ultimo step applichiamo il miglior modello classificatore a nuovi dati per valutare la sua performance. Il dataset di score contiene 3500 osservazioni (10% del dataset). Replicando il modello su nuovi dati di score 2493 imprese su 3500 vengono miss-classificate, Però rispetto i nostri obiettivi i veri negativi vengono tutti classificati correttamente 1007 specificity =.1
questa nuova matrice è stata creata sulla base di una nuova soglia (0.85) ottenuta dal modello Gradient Boosting.

Confusion matrix relativa al score data Gradient Boosting

	Reference	
Prediction	NO	YES
NO	1007	2493
YES	0	0

Metrica	**Valore**
Accuracy	**.288**
Sensitività	**0**
Specificity	**1**