Progetto di machine learning: modelli di classificazione

Membri del gruppo:

Bancora Davide | M. 905588
Donato Benedetta | M. 905338
Dubini Emanuele | M. 904078

Introduzione

I canali di prenotazione alberghieri online hanno cambiato decisamente le possibilità di prenotazione e il comportamento dei clienti.
Un numero significativo di prenotazioni alberghiere viene annullato e le ragioni tipiche delle cancellazioni sono il cambio di programma, i conflitti di programmazione, ecc. Questo è spesso facilitato dalla possibilità di farlo gratuitamente o preferibilmente a basso costo, il che è vantaggioso per gli ospiti dell'hotel, ma è un fattore meno desiderabile per i gestori delle strutture.

Il dataset “Hotel Reservations” fornisce una serie di informazioni relative ai soggiorni effettuati dai clienti degli hotel situati in tutto il mondo, incluse date di prenotazioni, cancellazioni precedenti, tipo di camere, tipo di prenotazioni, informazioni sul pagamento, ecc.

Tale progetto ha lo scopo di sfruttare questo dataset per identificare i modelli di classificazione che possono contribuire a prevedere se un cliente cancellerà la prenotazione precedentemente effettuata. In particolare, verrà effettuata un'analisi esplorativa delle covariate del dataset e verranno studiati due modelli di classificazione: alberi binari (apprendimento supervisionato), Random Forest e Reti neurali.

L'obbiettivo finale di questo progetto è quello di trovare il miglior modello di classificazione che sia in grado di prevedere precisamente se una cancellazione verrà mantenuta o meno.

Descrizione del Dataset

Il dataset utilizzato in questo progetto, disponibile al seguente link (https://www.kaggle.com/datasets/ahsan81/hotel-reservations-classification-dataset), è composto da 19 attributi riguardanti i principali fattori che possano determinare o meno la cancellazione di una prenotazione di un soggiorno in un Hotel.

La struttura del dataset utilizzato è descritta dall’elenco seguente:

1. ***Booking_ID:*** identificativo univoco di ogni prenotazione.
2. ***no_of_adults:*** numero di adulti presenti nella prenotazione.
3. ***no_of_children***: numero di bambini presenti nella prenotazione.
4. ***no_of_weekend_nights:*** numero di notti del fine settimana previste dalla prenotazione (sabato - domenica).
5. ***no_of_week_nights:*** numero di notti settimanali previste dalla prenotazione (lunedì - venerdì).
6. ***type_of_meal_plan:*** tipologia di piano pasti prenotato dal cliente.
7. ***required_car_parking_space***: se è stato richiesto un posto auto durante la prenotazione.

8. ***room_type_reserved:*** tipologia di camera prenotata dal cliente.
9. ***lead_time:*** tempo trascorso tra la prenotazione e l'arrivo nella struttura.
10. ***arrival_year:*** anno di inizio soggiorno.
11. ***arrival_month:*** mese in cui inizia il soggiorno.
12. ***arrival_date:*** giorno in cui inizia il soggiorno.
13. ***market_segment_type:*** il segmento di mercato da cui proviene la prenotazione (Aviation, Complementary, Corporate, Offline, Online).
14. ***repeated_guest:*** se il cliente è già stato ospite nella struttura in precedenza.
15. ***no_of_previous_cancellations:*** numero di prenotazioni precedenti che sono state annullate dal cliente.
16. ***no_of_previous_bookings_not_canceled:*** numero di prenotazioni precedenti che non sono state annullate dal cliente.
17. ***avg_price_per_room:*** costo medio giornaliero della camera (espresso in euro).
18. ***no_of_special_requests:*** numero di richieste “speciali” desiderate dal cliente (es. piano alto, vista dalla camera, ecc.).
19. ***booking_status:*** indica se la prenotazione è stata annullata o non cancellata.

Scelte di design per la creazione del data set, eventuali ipotesi o assunzioni

Dopo aver importato il dataset all'interno di RStudio, sono stati esaminati e manipolati i dati assicurandone la loro correttezza ed integrità prima di effettuarne l'analisi. Nello specifico, sono state esaminate le dimensioni del dataset e la tipologia di ciascuna covariata.

La presenza di valori mancanti e/o nulli può influire negativamente sull'accuratezza delle analisi causando problemi durante l'elaborazione dei dati, ad esempio durante la fase di training di un modello. Di conseguenza è stato verificato se il dataset preso in esame contenesse valori nulli e/o valori mancanti. La verifica ha confermato l'assenza di tali valori, come di seguito illustrato:

```
> # Identificazione dei valori mancanti
> missing_values <- colSums(is.na(data))
> print(missing_values)
                          Booking_ID                        no_of_adults
                                   0                                   0
                      no_of_children                no_of_weekend_nights
                                   0                                   0
                   no_of_week_nights                   type_of_meal_plan
                                   0                                   0
          required_car_parking_space                 room_type_reserved
                                   0                                   0
                           lead_time                        arrival_year
                                   0                                   0
                       arrival_month                        arrival_date
                                   0                                   0
                 market_segment_type                      repeated_guest
                                   0                                   0
       no_of_previous_cancellations no_of_previous_bookings_not_canceled
                                   0                                   0
                 avg_price_per_room               no_of_special_requests
                                   0                                   0
                      booking_status
                                   0
```

Inoltre, considerato la colonna ***Booking_ID***, essa non ha alcun valore predittivo per il modello di classificazione e quindi è stata rimossa senza influire sulle prestazioni del modello. La sua presenza potrebbe causare errate previsioni basate sull'ID univoco, senza generalizzare bene sui dati di test.

In seguito, sono state convertite alcune delle covariate del dataset, tra cui la variabile target, in tipo *factor* utilizzando la funzione **`factor()`**. In particolare, vengono convertite le seguenti covariate: ***booking_status***, ***type_of_meal_plan***, ***room_type_reserved***, ***market_segment_type***, ***required_car_parking_space***, ***repeated_guest***.

In merito alle variabili ***type_of_meal_plan***, ***room_type_reserved***, ***market_segment_type***, ***required_car_parking_space***, ***repeated_guest***, abbiamo notato che queste covariate assumono valori che appartengono ad un insieme predefinito di categorie. Ad esempio, la covariata "required_car_parking_space" rappresenta la categoria "richiesta di posto auto" o "nessuna richiesta di posto auto" e i valori possibili sono "0" o "1". Un ulteriore esempio è la variabile "type_of_meal_plan" che può assumere 4 valori differenti, ciascuno dei quali rappresenta un tipo di piano pasti specifico. Analogamente, la variabile "booking_status" può assumere solo 2 valori, Canceled o Not_Canceled.

Una volta effettuate le operazioni sopra descritte, il dataset si presenta in questo modo:

```
> str(data)
'data.frame':   36275 obs. of  18 variables:
 $ no_of_adults                      : int  2 2 1 2 2 2 2 2 3 2 ...
 $ no_of_children                    : int  0 0 0 0 0 0 0 0 0 0 ...
 $ no_of_weekend_nights              : int  1 2 2 0 1 0 1 1 0 0 ...
 $ no_of_week_nights                 : int  2 3 1 2 1 2 3 3 4 5 ...
 $ type_of_meal_plan                 : Factor w/ 4 levels "Meal Plan 1",..: 1 4 1 1 4 2 1 1 1 1 ...
 $ required_car_parking_space        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ room_type_reserved                : Factor w/ 7 levels "Room_Type 1",..: 1 1 1 1 1 1 1 4 1 4 ...
 $ lead_time                         : int  224 5 1 211 48 346 34 83 121 44 ...
 $ arrival_year                      : int  2017 2018 2018 2018 2018 2018 2017 2018 2018 2018 ...
 $ arrival_month                     : int  10 11 2 5 4 9 10 12 7 10 ...
 $ arrival_date                      : int  2 6 28 20 11 13 15 26 6 18 ...
 $ market_segment_type               : Factor w/ 5 levels "Aviation","Complementary",..: 4 5 5 5 5 5 5 5 4 5 ...
 $ repeated_guest                    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ no_of_previous_cancellations      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ no_of_previous_bookings_not_canceled: int  0 0 0 0 0 0 0 0 0 0 ...
 $ avg_price_per_room                : num  65 106.7 60 100 94.5 ...
 $ no_of_special_requests            : int  0 1 0 0 0 1 1 1 1 3 ...
 $ booking_status                    : Factor w/ 2 levels "Canceled","Not_Canceled": 2 2 1 1 1 1 2 2 2 2 ...
>
```

Divisione Dataset in trainset e testset

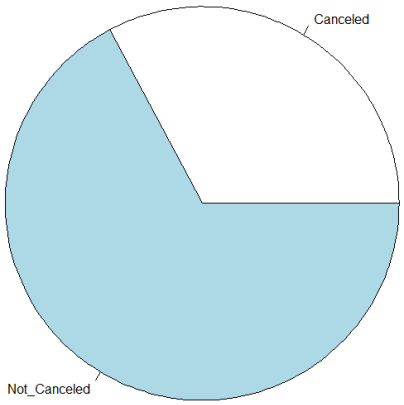
Dopo aver analizzato le covariate del dataset, abbiamo deciso di suddividere quest'ultimo in due parti: Trainset e Testset con una percentuale di partizione delle istanze 70%-30%.

Il set di test ha permesso di valutare come il modello si comporta su dati che non ha osservato durante l'addestramento ed inoltre, la divisione in trainset e testset permette di prevenire il fenomeno di *overfitting*; infatti, se un modello viene addestrato su troppi dati, può imparare a memoria i dati di train e non essere in grado di generalizzare bene sui nuovi dati.

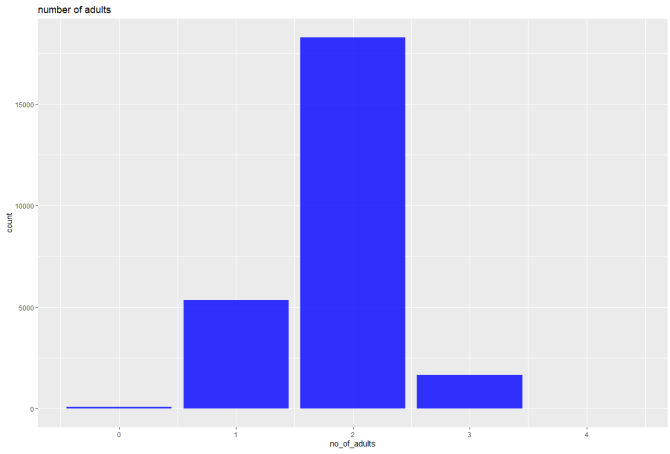
Descrizione del training set: analisi esplorativa delle covariate

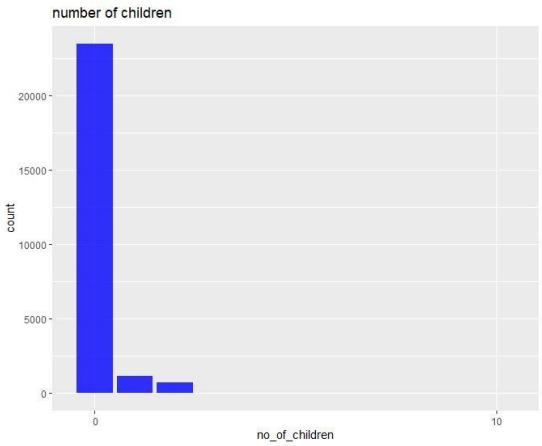
Analizzando i dati presenti all'interno del training set, abbiamo notato che le prenotazioni non cancellate sono superiori alle superiori cancellate, come di seguito mostrato:

Prenotazioni cancellate: 8320
Prenotazioni non cancellate: 17073



Continuando l'esplorazione del training set, la maggior parte delle prenotazioni presenta un numero di adulti pari a due. Inoltre, osservando il grafico relativo al numero di bambini presenti in ciascuna prenotazione, si può concludere che i soggiorni effettuati da coppie di adulti senza figli primeggiano su quelle composte da adulti e bambini.



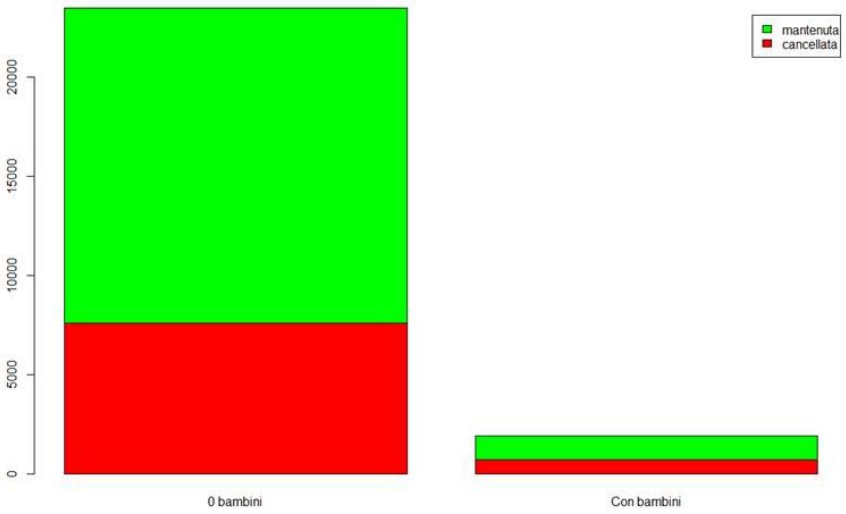


Ottenute queste informazioni, i passi successivi dell'analisi verificano se la presenza di uno o più bambini all'interno di una prenotazione possa essere motivo di cancellazione del soggiorno prenotato. Tale scelta è giustificata dal fatto che le esigenze dei bambini possono comportare imminenti cambiamenti di programma a causa della loro natura.

Per fare questo abbiamo inizialmente calcolato la quantità di prenotazioni con un numero di bambini maggiore di zero e successivamente calcolato la percentuale di cancellazione che corrisponde al 37,5%.

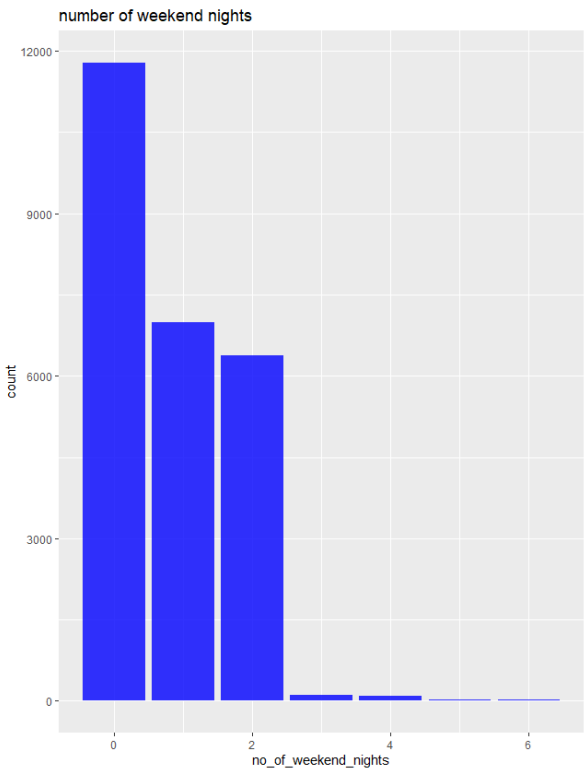
La stessa operazione è stata effettuata anche per le prenotazioni di soli adulti e la percentuale di prenotazioni cancellate è pari al 32,4%.

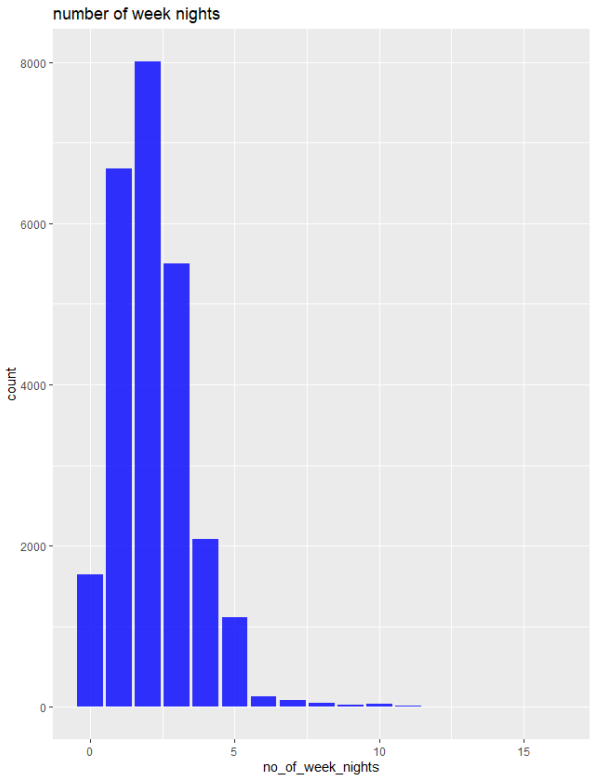
Di seguito riportiamo il barplot, che mostra quanto descritto:



A differenza di quanto ipotizzato prima di effettuare le analisi, la presenza o meno di bambini all'interno di una prenotazione influenza di poco sulla futura decisione di cancellazione del soggiorno.

Continuando con l'esplorazione dei dati andiamo ora a verificare come si distribuisce la permanenza degli ospiti negli Hotel:

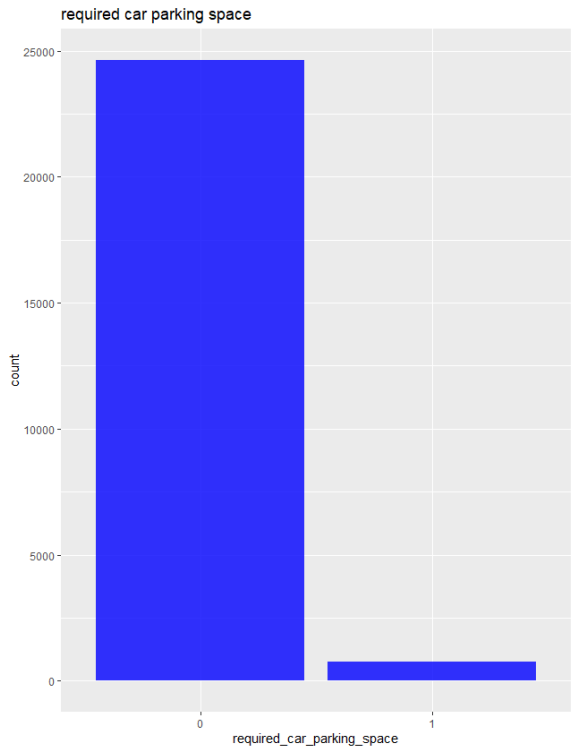




Dal primo barplot si osserva che più della metà dei soggiorni prenotati presenta almeno una notte; inoltre, osservando il secondo grafico, la maggior parte delle prenotazioni prevedono di riservare una stanza per due notti durante la settimana.

Osservando l'attributo relativo al piano pasti, "type_of_meal_plan", concludiamo che per la maggior parte delle prenotazioni è stato scelto "Meal Plan 1". Il dataset non fornisce informazioni utili relative ai vari piani pasti. L'attributo può assumere 4 valori (di cui uno è "Not Selected").

Proseguendo con l'analisi delle covariate, prendiamo ora in considerazione l'attributo "required_car_parking_space" che indica la richiesta di avere o meno il parcheggio riservato alla vettura dell'ospite: la maggior parte delle istanze assume valore pari a zero. Questo valore indica che il cliente non ha richiesto il posto auto durante la fase di prenotazione del soggiorno oppure possiamo supporre che alcune strutture alberghiere non dispongano di parcheggi riservati agli ospiti e di conseguenza il valore rimani pari a zero.

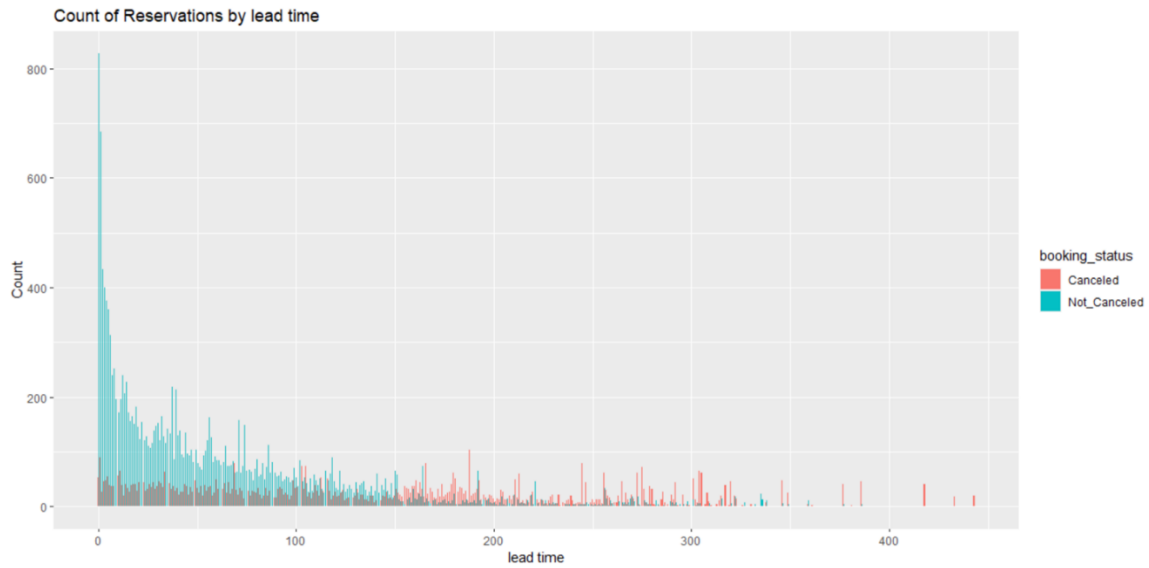


Osservando inoltre il grafico che mette in relazione la richiesta del parcheggio e la colonna target, possiamo osservare che dal momento in cui, all'interno della prenotazione viene richiesta la presenza di un parcheggio privato, quest'ultima ha una probabilità molto bassa di essere successivamente cancellata (circa 9,4%):

Riferendosi all'attributo "room_type_reserved", il tipo di camera prenotata più frequentemente è il Room_Type 1.

Siamo andati ora ad osservare il numero di giorni che intercorrono tra la prenotazione e il soggiorno notando che la maggior parte delle prenotazioni vengono effettuate entro circa tre mesi dal soggiorno. Queste informazioni sono state verificate tramite la seguente riga di codice:

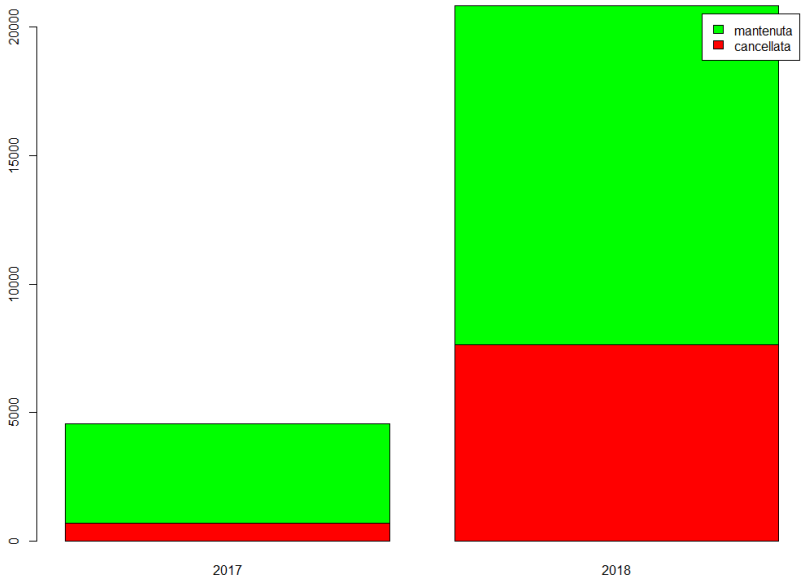
```
> nrow(train_set[train_set$lead_time<100, ])
```



Calcolando la relazione che esiste tra il target ed il “lead_time”, possiamo osservare che le prenotazioni avvenute nei tre mesi antecedenti al soggiorno hanno una probabilità di essere cancellate più bassa rispetto alle restanti prenotazioni.

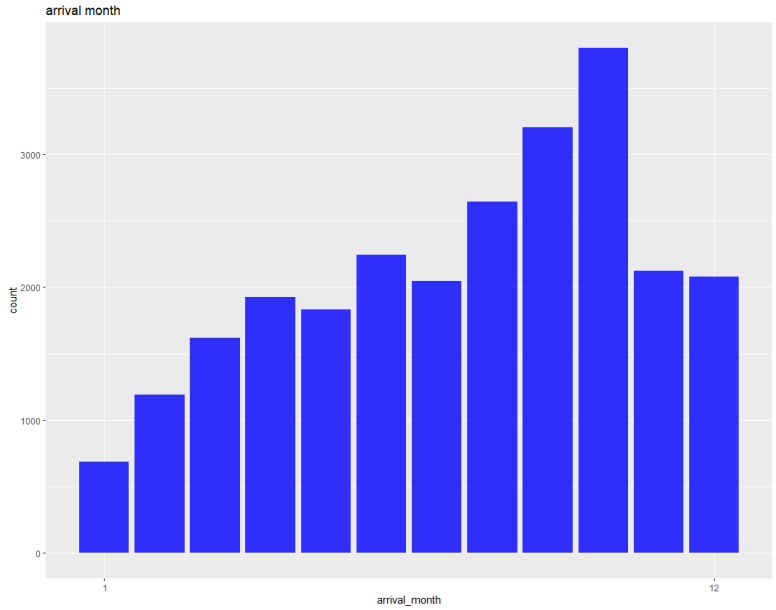
Osservando il plot relativo all’attributo “arrival_year” si può notare che la maggior parte delle prenotazioni è avvenuta nell’anno 2018. Con questo attributo abbiamo voluto verificare se l’anno di arrivo dei clienti presso la struttura ospitante fosse determinante per la scelta del target.

Il grafico ottenuto è il seguente:

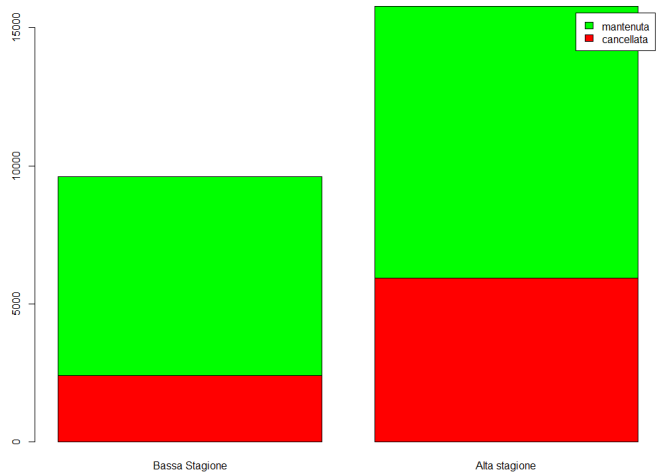


Analizzando questo nuovo grafico, abbiamo scoperto che la percentuale di prenotazioni cancellate nell’anno 2017 è pari al 14,9%, mentre nel 2018 la percentuale di cancellazioni è pari al 36,7%, concludendo che la differenza di percentuale non è così marcata.

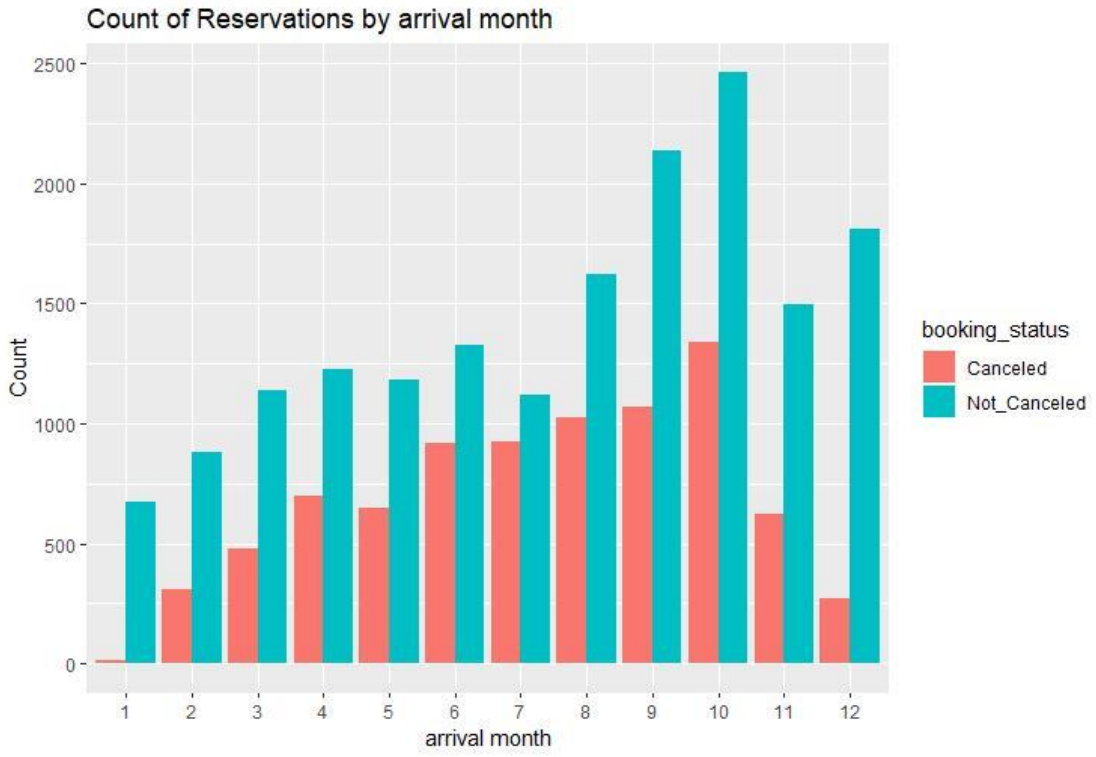
Analizzando nuovamente il dataset, notiamo che la maggior parte degli arrivi nelle strutture alberghiere è avvenuto dopo il mese di giugno, come riporta il grafico:



Ancora una volta abbiamo voluto verificare se l'attributo "arrival_month" fosse una covariata determinate per la scelta della colonna target: per fare ciò, abbiamo considerato che il periodo di alta stagione comprendesse i mesi da maggio ad ottobre.

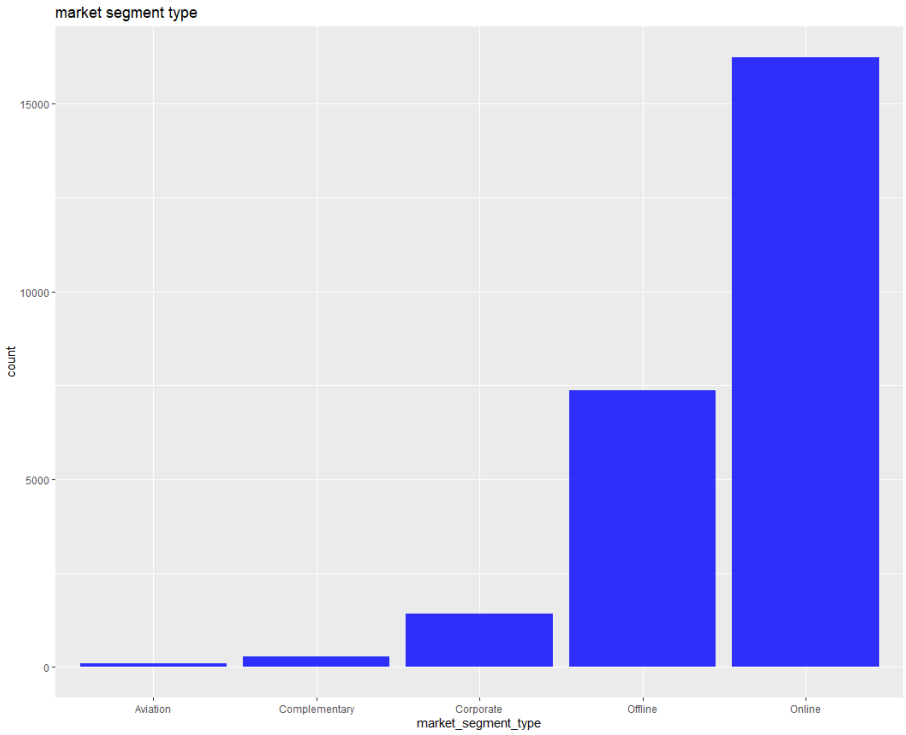


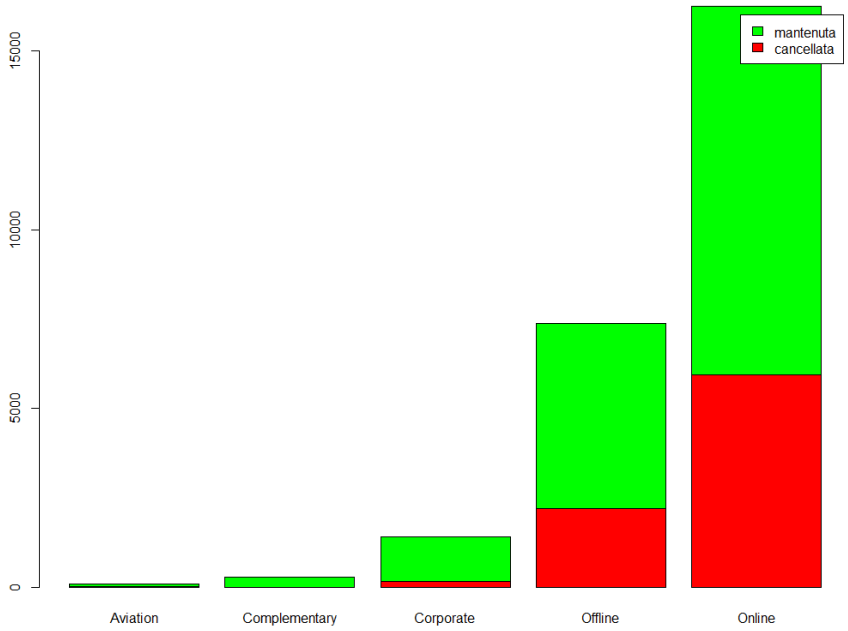
Dal grafico si evince che le prenotazioni cancellate nel periodo di alta stagione corrispondono al 37,6% mentre le prenotazioni cancellate nel periodo di bassa stagione al 24,9%.
Con questi dati non otteniamo informazioni determinanti per la scelta del target, ma realizzando il grafico che rappresenta le prenotazioni distribuite mese per mese (senza distinzione tra bassa e alta stagione) notiamo che le prenotazioni avvenute durante il mese di dicembre e gennaio, difficilmente verranno cancellate:



La successiva covariata analizzata “arrival date” non fornisce informazioni utili per classificare le istanze del dataset e per questo motivo non è stato riportato il grafico (comunque generabile tramite script).

Procedendo con l'esplorazione del trainset, abbiamo notato che una covariata rilevante per la scelta del target è "market_segment_type". Dal primo grafico si evince che la maggior parte delle prenotazioni vengono effettuate online e guardando la percentuale di prenotazioni cancellate in relazione al metodo di prenotazione del soggiorno, è possibile notare che queste hanno la percentuale di cancellazione più alta (36,5%), supponendo che questo alto valore sia dato dalla comodità di poter cancellare la prenotazione.



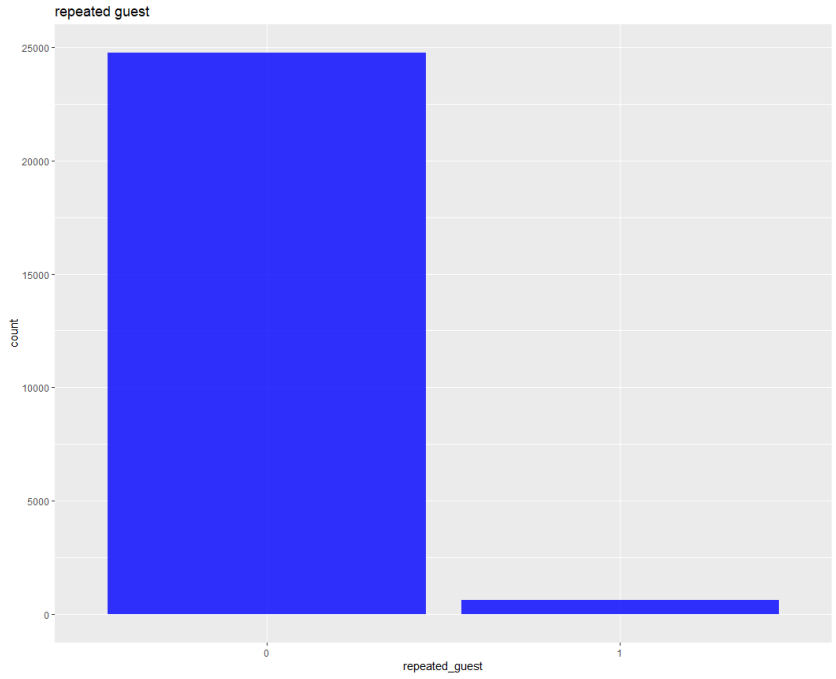


È possibile notare che le prenotazioni Complementary presentando una cancellazione pari allo 0%. Questo avviene perché sono prenotazioni offerte gratuitamente al cliente da parte della struttura alberghiera per promuoverla o per premiare i clienti fedeli e difficilmente il cliente le cancellerà.

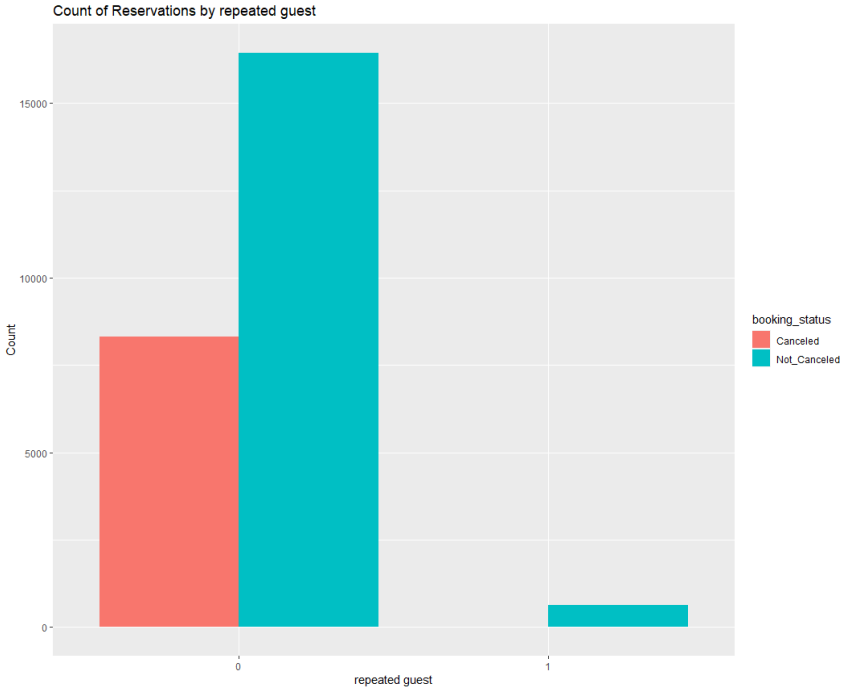
Anche le cancellazioni delle prenotazioni corporate, ossia quelle effettuate dalle aziende per i propri dipendenti, sono molto basse (pari a 10,8%).

Le prenotazioni che avvengono offline, hanno un tasso di cancellazione più basso rispetto alle prenotazioni avvenute online, ma comunque pari al 29.9%.

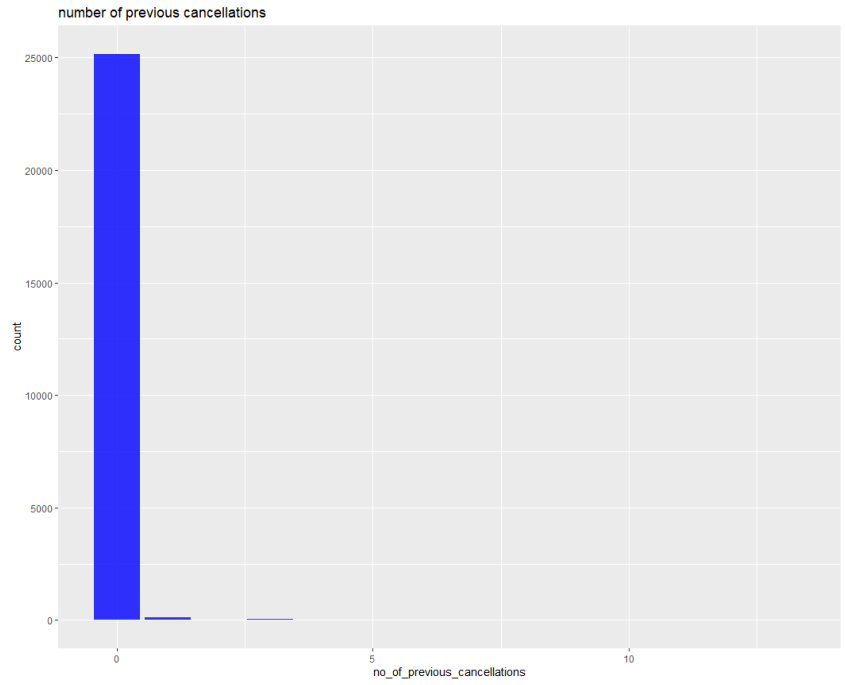
Analizzando il dataset, possiamo inoltre notare che la maggior parte delle prenotazioni viene effettuata da parte di clienti che non hanno mai soggiornato nella struttura alberghiera:



Analizzando le prenotazioni effettuate dai clienti abituali, concludiamo che il tasso di cancellazione è pari all'1,6%, mentre le prenotazioni effettuate dai nuovi clienti hanno una probabilità di essere cancellate pari al 33,6%.

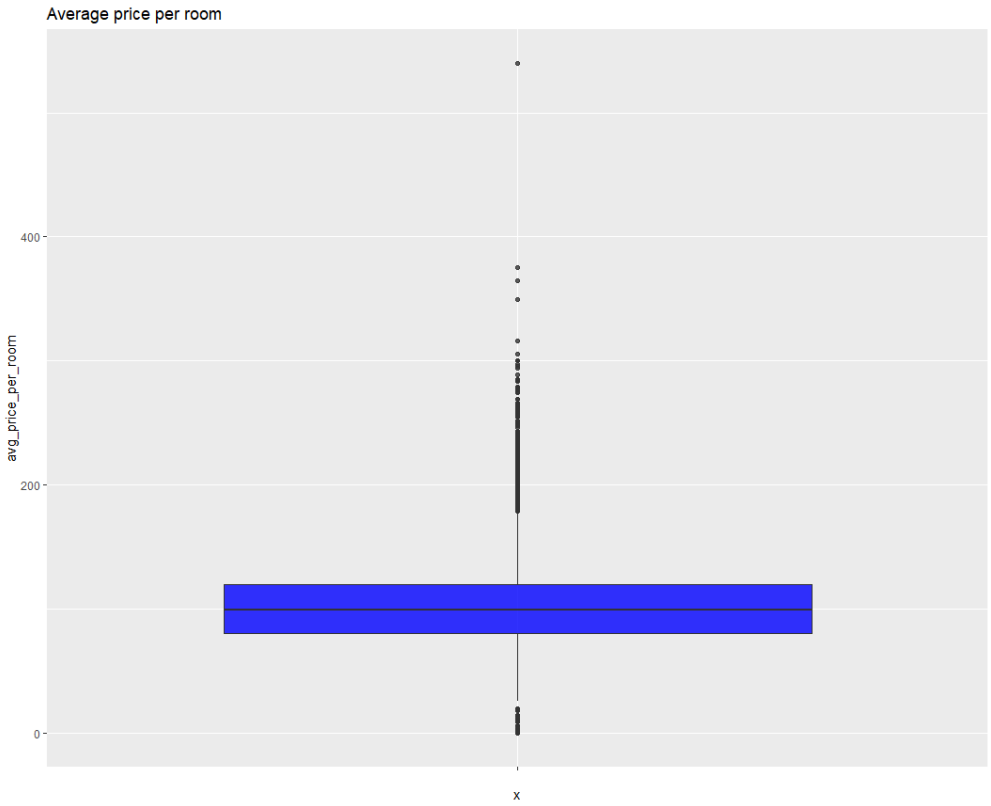


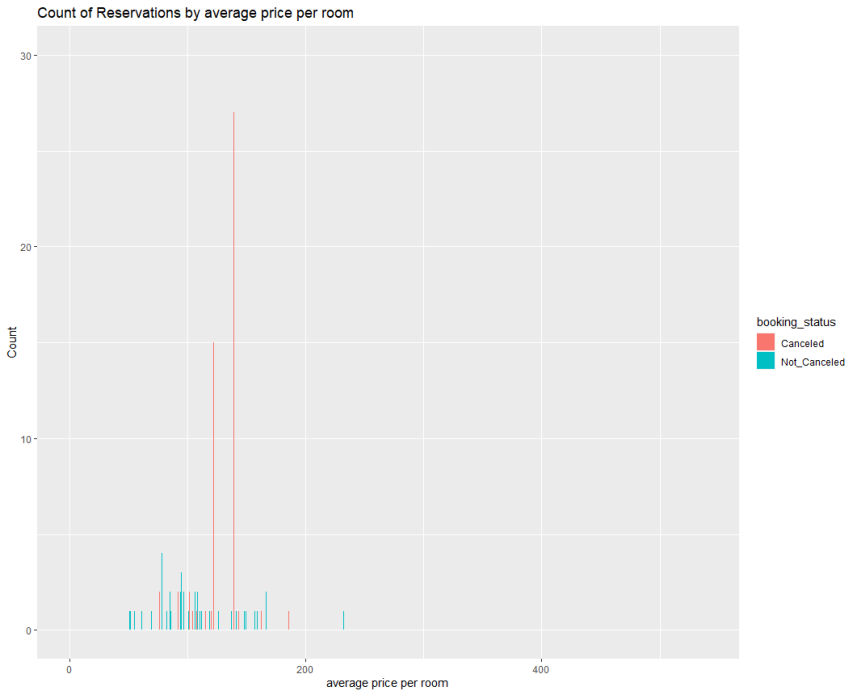
L'affidabilità del cliente potrebbe essere valutata in base al numero di cancellazioni da lui precedentemente effettuate. Nel dataset in questione, la maggior parte dei clienti non ha mai effettuato una cancellazione in precedenza.



Un ulteriore dato utile a verificare l'affidabilità del cliente riguarda il conteggio delle precedenti cancellazioni effettuate nella stessa struttura. Anche in questo caso, la maggior parte dei clienti non ha mai effettuato cancellazioni nella struttura in cui si recherà per il soggiorno. Questo dato è confermato dal fatto che il numero di repeted guest è basso.

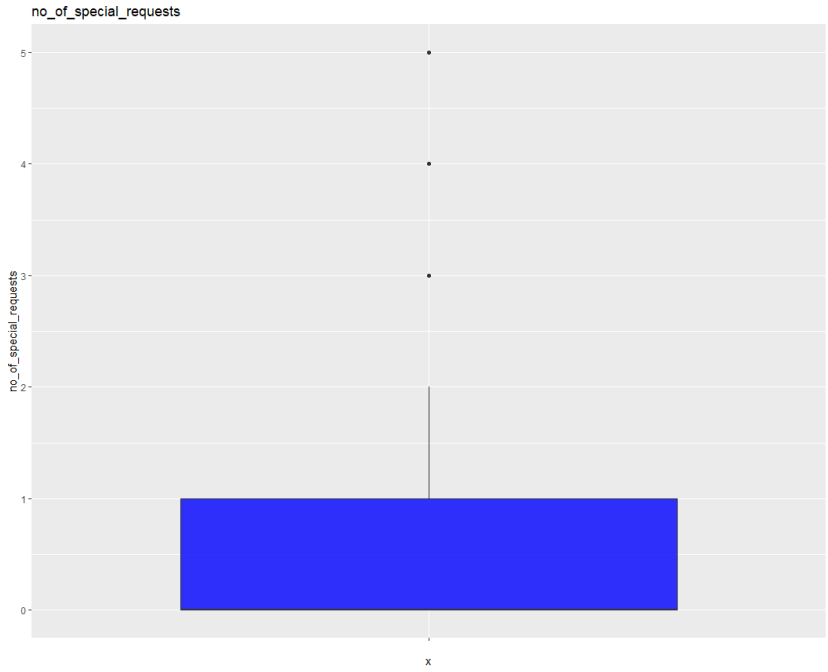
Analizzando ora il prezzo medio per camera possiamo affermare che questo si aggira intorno ai 100€:



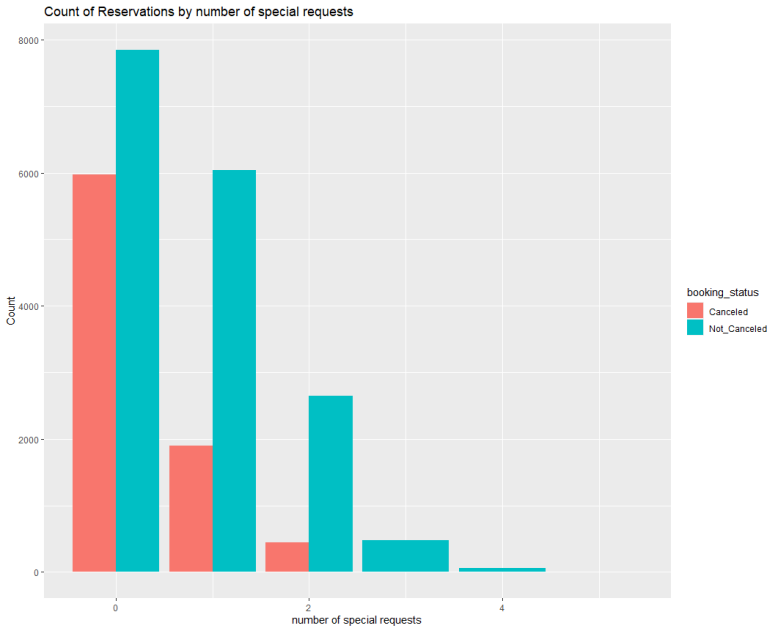


Da questo grafico possiamo notare che la maggior parte delle camere cancellate ha un prezzo medio compreso tra i 120€ a 150€.

L'ultimo attributo che andiamo ad analizzare riguarda il numero di richieste speciali che un cliente può richiedere alla struttura alberghiera. Gran parte delle prenotazioni non contengono richieste speciali o al più ne contengono una.



Un'importante osservazione ottenuta analizzando la relazione tra la covariata "no_of_special_request" e l'attributo target riguarda la percentuale di cancellazione delle prenotazioni: al crescere delle richieste speciali diminuisce la probabilità che il cliente cancelli la prenotazione (pari a 0% quando il cliente inserisce 3 o più richieste speciali); concludiamo quindi che i due attributi sono inversamente proporzionali tra loro.



Descrizione dei modelli di machine learning scelti

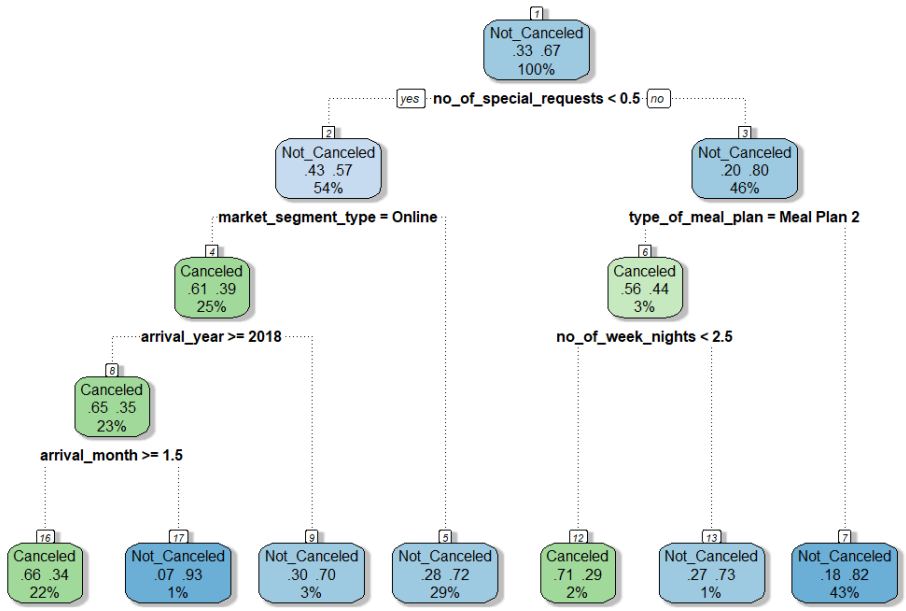
In questo studio sono state implementate tre differenti tecniche di classificazione con lo scopo di individuare la più adatta, sulla base dei dati disponibili: gli Alberi Decisionali, i Random Forest e le reti neurali.

Alberi decisionali

Abbiamo utilizzato il modello di classificazione degli alberi decisionali in quanto consente di avere una visualizzazione chiara e semplice dei risultati ottenuti e permette di gestire dati numerici e categorici presenti in larga scala nel dataset; in aggiunta, gli alberi decisionali sono in grado di individuare relazioni tra le diverse variabili del dataset con il target creando una gerarchia di decisioni che consentono di prevedere la classe di appartenenza delle istanze che altrimenti sarebbero difficilmente osservabili solo guardando i dati.

Gli alberi decisionali possono aiutare a identificare quali variabili sono più importanti per la classificazione delle prenotazioni in "cancellate" o "non cancellate". Ciò può essere utile per capire quali fattori influenzano maggiormente la decisione di cancellare una prenotazione e quindi permette di prendere future decisioni fondate.

Nella nostra analisi, dopo aver importato la libreria *rpart* ed aver creato il modello dell'albero, specificando quali attributi utilizzare per la costruzione del modello decisionale, effettuiamo le predizioni sul test-set. L'albero ottenuto è il seguente:



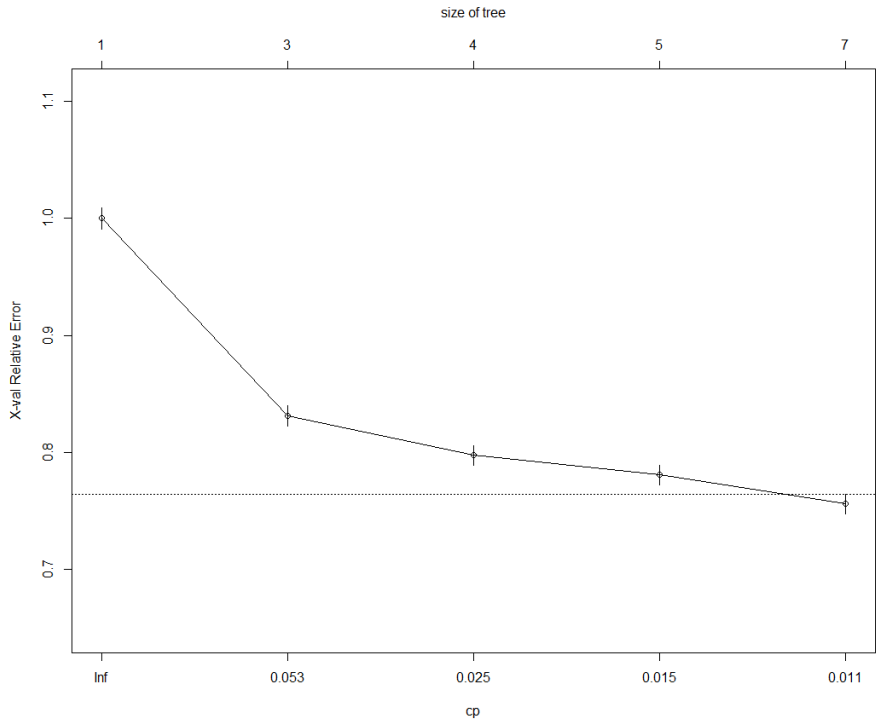
Analizzando il risultato, abbiamo riscontrato che il valore dell'accuratezza è pari al 75,28%, ottenendo la seguente matrice di confusione:

```
                Reference
Prediction      Canceled Not_Canceled
  Canceled          1748          873
  Not_Canceled      1817         6444

               Accuracy : 0.7528
                 95% CI : (0.7446, 0.7609)
    No Information Rate : 0.6724
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.398
```

Tali valori non risultano essere molto soddisfacenti, soprattutto se si osserva il valore Kappa che indica la concordanza tra le previsioni del modello e i risultati reali, rispetto alla concordanza che si otterrebbe scegliendo a caso il target; il valore da noi ottenuto è pari a 0,398 ed indica che il modello non è molto preciso.

Sarebbe interessante verificare la possibilità di semplificare il modello ottenuto riducendo la probabilità di overfitting senza peggiorare di molto le prestazioni: per far questo, dobbiamo analizzare il valore *cp* (complexity parameter) che indica la complessità dell'albero generato.

Il nostro obiettivo è quello di aumentare questo valore, per creare un albero decisionale più semplice pur cercando di mantenere un alto valore di accuratezza:



Sulla base delle assunzioni precedenti, il valore di complexity parameter è stato incrementato da 0,011 a 0,015 osservando un cambiamento minimo del valore di accuratezza rispetto al valore originale (cp = 0,011) pari al 74,55%, come mostrato nella figura seguente:

```
                Reference
Prediction      Canceled Not_Canceled
  Canceled          1608          813
  Not_Canceled      1957         6504

               Accuracy : 0.7455
                 95% CI : (0.7372, 0.7536)
    No Information Rate : 0.6724
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3704
```

L'Information Rate di 0,6724 indica che, se si scegliesse a caso, si otterrebbero correttamente il 67% delle previsioni. Il valore Kappa che si ottiene con il nuovo valore di *cp* è poco più basso rispetto al precedente.

```
Precision : 0.6642
   Recall : 0.4511
       F1 : 0.5373
```

Valori ottenuti considerando la classe positiva "Canceled"

```
Precision : 0.7687
   Recall : 0.8889
       F1 : 0.8244
```

Valori ottenuti considerando la classe positiva "Not_Canceled"

In base ai dati ottenuti, possiamo concludere che il modello di albero decisionale ha ottenuto risultati differenti nella previsione delle due classi "canceled" e "not canceled". Per la classe "canceled", la *precision* è di 0,6642, il che significa che il 66,42% delle volte il modello ha previsto una prenotazione annullata e questa è effettivamente stata annullata. Il *recall* è di 0,4511, il che significa che il modello ha previsto correttamente solo il 45,11% delle prenotazioni annullate. La *F1-score* di 0,5373 indica che la performance del modello in questa classe è mediocre.

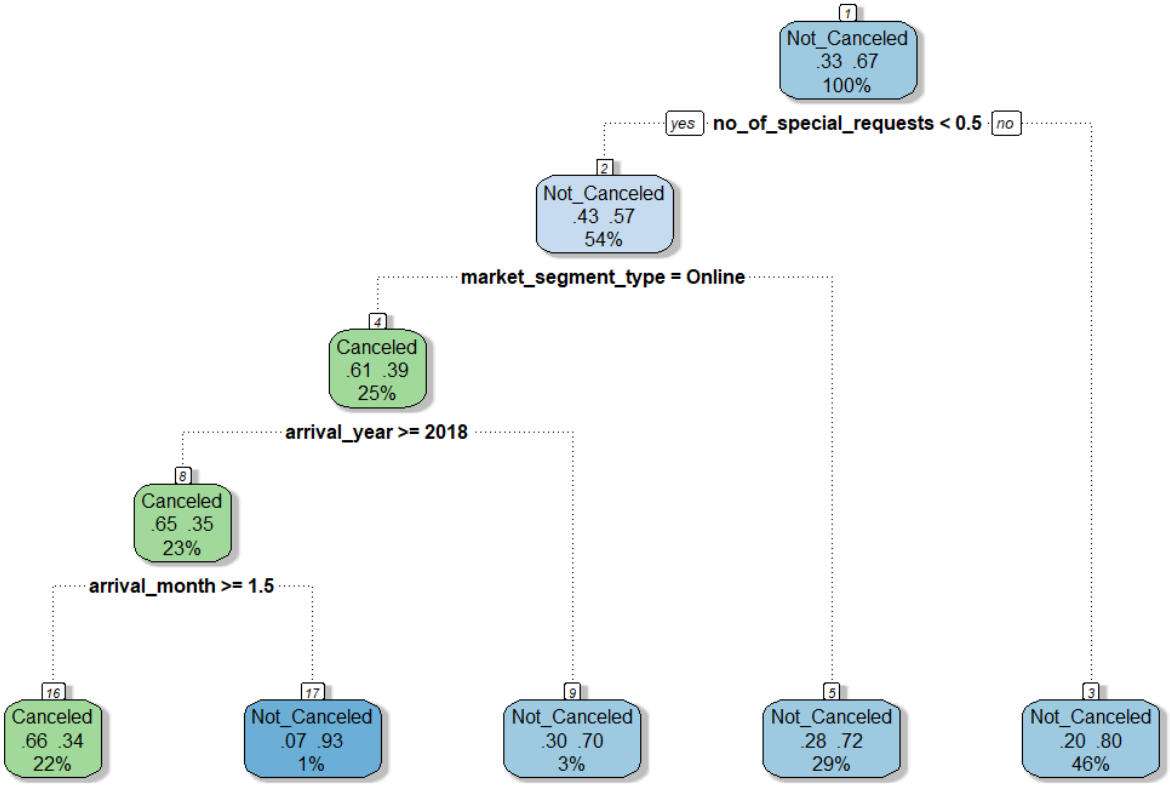
Per la classe "not canceled", la *precision* è di 0,7687, il che significa che il 76,87% delle volte che il modello ha previsto una prenotazione non annullata, questa non è stata effettivamente annullata. Il *recall* è di 0,8889, il che significa che il modello ha previsto correttamente l'88,89% delle prenotazioni non annullate. La *F1-score* di 0,8244 indica che la performance del modello in questa classe è molto buona.

In generale, possiamo concludere che il modello funziona meglio nella previsione di prenotazioni non cancellate (“Not Canceled”) rispetto a quelle successivamente annullate.

Calcolando la *Precision Macro Average* si ottiene un valore di 72,34% che rappresenta una media non pesata della misura di performance “Precision” di tutte le classi del target. Calcolando anche la *Recall Macro Average* otteniamo un valore pari a 68,55% mentre per la *F1 Macro Average* otteniamo un valore di 69,62%.

Riducendo ulteriormente il valore di complexity parameter a 0,025 si osserva un ulteriore cambiamento minimo nel valore dell'accuratezza e un'ulteriore diminuzione del valore Kappa.

Dopo avere analizzato i risultati di accuratezza e Kappa ottenuti con le matrici di confusione, abbiamo deciso che il *cp* pari a 0,015 rappresenta un giusto compromesso tra accuratezza (74.55%) e complessità dell'albero abbassando la probabilità di overfitting:



Come nell'albero ottenuto con *cp* pari a 0,011 possiamo notare che l'attributo che viene considerato inizialmente riguarda il numero di richieste speciali inserite dall'utente. Se questo valore è maggiore di 0, non è necessario verificare altri attributi in quanto l'albero classifica le prenotazioni come "Not_Canceled". Se, al contrario, il numero di richieste speciali corrisponde a 0, è necessario verificare ulteriori parametri: se la prenotazione non è stata effettuata online, allora l'albero classifica la prenotazione come "Not_Canceled", mentre se il cliente ha utilizzato un sito web per prenotare il soggiorno, è necessario anche verificare l'anno in cui è stata effettuata la prenotazione; se la prenotazione è stata effettuata durante il 2017, allora questa non verrà cancellata, mentre se è stata effettuata durante l'anno 2018 sarà verificato anche il mese di prenotazione. Se la prenotazione è stata effettuata oltre il mese di gennaio allora la prenotazione verrà cancellata, altrimenti no.

Tutti gli attributi che vengono presi in considerazione dall'albero rispecchiano i risultati che sono stati ottenuti nel paragrafo dell'analisi dei dati.

Random Forest

Un secondo algoritmo di apprendimento che abbiamo voluto testare sul nostro dataset è il *Random Forest*.

Essi sono un algoritmo di apprendimento automatico basati sulla costruzione di molteplici alberi di decisione. Il concetto fondamentale è che si ottiene un modello più preciso rispetto a quello che potrebbe essere ottenuto utilizzando un singolo albero di decisione. Inoltre, gli alberi Random Forest sono noti per essere robusti ai rumori presenti nei dati, per avere una bassa varianza rispetto a un singolo albero di decisione e per essere in grado di gestire problemi di overfitting.

Per ottenere dei risultati migliori, abbiamo deciso di normalizzare le variabili, ad esclusione di quelle che rappresentano le date di arrivo, dato che sono rappresentate su una scala temporale unica. Anche gli attributi relativi al numero di prenotazioni cancellate e non cancellate non vengono normalizzate perché hanno valori troppo distanti tra di loro.

Una volta normalizzate le variabili, abbiamo suddiviso il dataset in trainset e testset e abbiamo installato la libreria RandomForest.

Successivamente è stato costruito il modello Random Forest passando come parametri tutte le variabili del trainset, specificando che il modello deve essere costruito utilizzando 500 alberi e che ogni albero deve utilizzare 3 variabili scelte a caso per suddividere i nodi.

Costruito il modello, abbiamo calcolato la matrice di confusione:

```
                Reference
Prediction     Canceled Not_Canceled
  Canceled         2824          418
  Not_Canceled      741         6899
```

Da un primo confronto con l'albero decisionale precedentemente realizzato, è possibile notare che il numero di falsi positivi e di falsi negativi è sicuramente minore in questo caso.

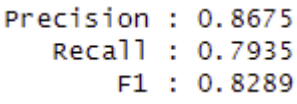
```
               Accuracy : 0.8935
                 95% CI : (0.8875, 0.8992)
No Information Rate : 0.6724
P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7525
```

L'ipotesi è sostenuta anche dal calcolo dell'accuratezza che è pari all' 89,35%, un valore nettamente maggiore rispetto a quanto ottenuto con l'albero decisionale costruito prima. Anche il valore Kappa ottenuto con il modello è nettamente superiore rispetto all'albero decisionale: questo indica un buon livello di precisione.

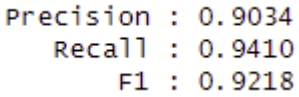
Successivamente abbiamo calcolato i valori di Precision, Recall e F1-Measue per ogni possibile etichetta del target:

- *Precision* indica la percentuale di veri positivi tra tutti i casi classificati come positivi dal modello. In altre parole, la precision indica la capacità del modello di non classificare come positivi casi che in realtà sono negativi;
- *Recall* indica la percentuale di veri positivi che sono stati correttamente identificati dal modello. In altre parole, il recall indica la capacità del modello di identificare tutti i casi positivi presenti nei dati;
- *F1-Measure* è una metrica che combina precision e recall in un unico punteggio. F1 è una media ponderata tra precision e recall e fornisce una misura equilibrata della performance del modello.

Riportiamo di seguito i risultati ottenuti:



Valori ottenuti considerando la classe positiva "Canceled"



Valori ottenuti considerando la classe positiva "Not_Canceled"

I risultati ottenuti mostrano che il modello presenta buone performance per entrambe le etichette del target.

Per la classe positiva "Canceled", la precision è di 0,8675, il che significa che circa l'87% dei casi classificati come "Canceled" sono veramente "Canceled". Il recall è di 0,7935, dove circa l'80% dei casi "Canceled" presenti nei dati sono stati identificati correttamente dal modello. L'F1 score di 0,8289 indica un equilibrio tra precision e recall.

Considerando come classe positiva "Not_canceled", la precision è di 0,9034, il che significa che circa il 90% dei casi classificati come "Not_canceled" sono veramente "Not_canceled". Il recall è di 0,9410, il che significa che circa il 94% dei casi "Not_canceled" presenti nei dati sono stati identificati correttamente dal modello. L'F1 score di 0,9218 indica un equilibrio tra precision e recall.

In generale, i risultati mostrano che il modello ha una buona capacità di classificare correttamente sia i casi "Canceled" che i casi "Not_canceled". Un'ulteriore conferma è ottenuta calcolando i valori di *Macro Average* di ogni misura di performance come la *Precision Macro Average* che ha valore pari a 88,70%, *Recall Macro Average* pari a 86,75% e *F1 Macro Average* uguale a 87,61%.

Confrontando questi valori con quanto ottenuto analizzando il modello *Decision Tree* deduciamo che si ha un notevole miglioramento.

Rete neurale

Un ulteriore modello che abbiamo applicato al nostro dataset sono le reti neurali: ispirate al sistema nervoso umano, questi modelli consistono in una serie di nodi interconnessi che elaborano le informazioni in una serie di calcoli trasmettendo dei segnali.
L'applicazione di questo modello al dataset preso in considerazione è risultata complessa, ed è stato necessario applicare il processo di binarizzazione per gli attributi categorici. L'operazione si è resa necessaria in quanto le reti neurali accettano solo valori numerici, ma nel dataset erano presenti diversi attributi di tipo categorico.

Inoltre, abbiamo convertito tutti i valori di tipo intero in tipo numerico.

Anche la colonna relativa al target, che può assumere due valori, ha subito una modifica: il valore "canceled" è stato sostituito con il valore numerico 1 e 0 per "not_canceled".

Tramite la libreria mltools e data.table si va ad effettuare il processo di binarizzazione trasformando, come precedentemente illustrato, i dati di tipo categorico in tipo numerico.

Successivamente alla trasformazione, suddividiamo in modo casuale il dataset modificato in train-set (70%) e test-set (30%). A questo punto, la rete neurale può essere addestrata tramite la libreria neuralnet, dove la funzione associata specifica la variabile d'uscita booking_status, una funzione di attivazione di tipo logistico, il numero massimo di iterazioni del processo di addestramento (stepmax) e il set dei dati utilizzato per addestrare la rete.

Per calcolare la seguente funzione abbiamo utilizzato duemila istanze del dataset originale; questa scelta è data dal fatto che all'aumentare del numero delle istanze nella rete, essa richiedeva diversi giorni a restituire il risultato.

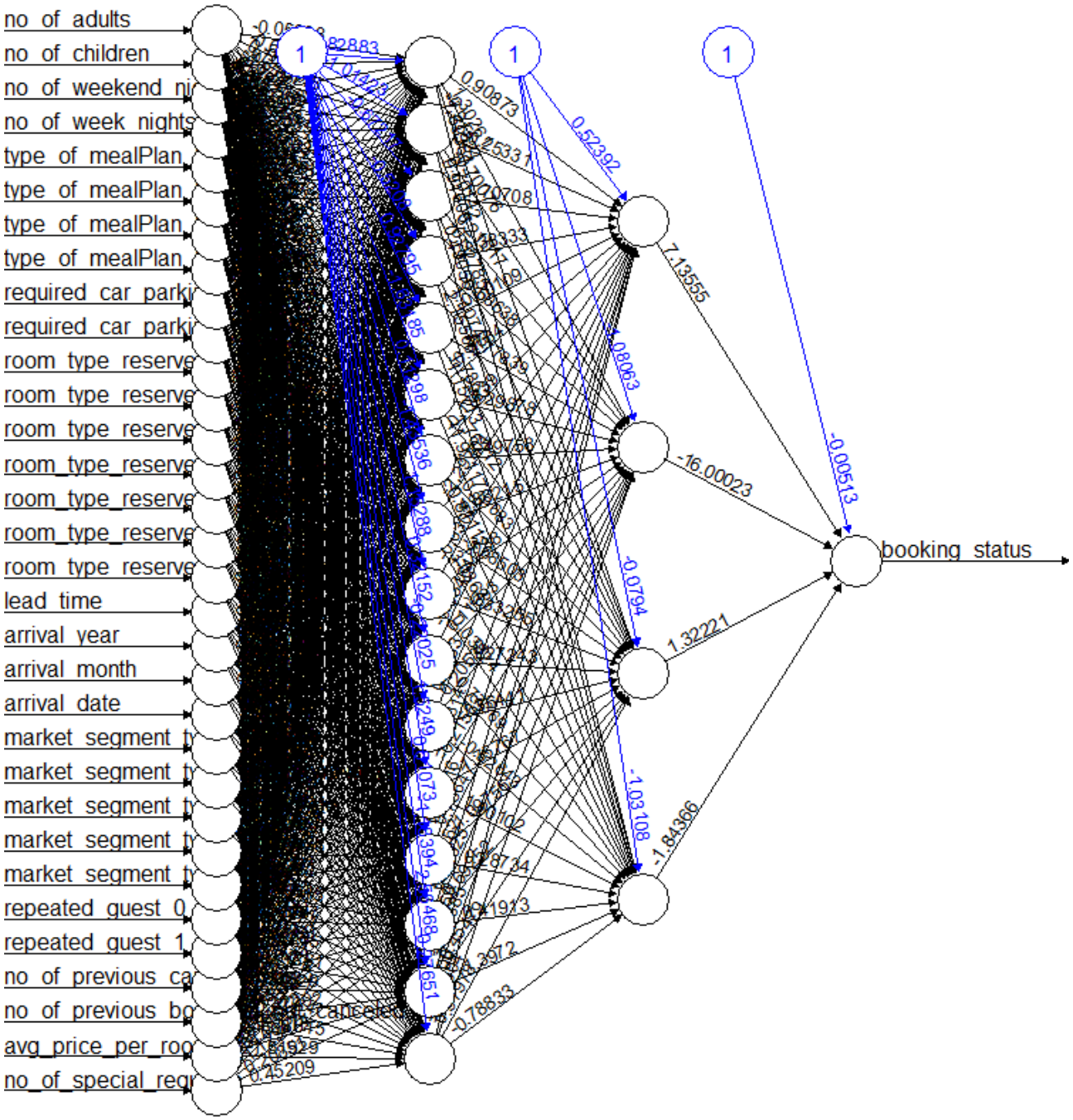
É stato impostato un valore di "stepmax" pari a 1e7: questo è un parametro utilizzato nei modelli di rete neurale per specificare il numero massimo di iterazioni da eseguire durante il processo di addestramento.

Tale parametro è importante perché l'addestramento di un modello di rete neurale può essere un processo costoso a livello computazionale e può richiedere molto tempo; quindi, è utile limitare il numero di iterazioni per evitare tempi di elaborazione eccessivi.

Il modello implementato non convergeva prima che fosse raggiunto il numero massimo di iterazioni specificato da "stepmax" ed infatti il processo di addestramento è stato interrotto più volte con il seguente codice senza calcolare i pesi della rete:

```
Warning message:
Algorithm did not converge in 1 of 1 repetition(s) within the stepmax.
```

Con un numero di istanze ridotto abbiamo ottenuto la seguente rete:



Questa rete neurale utilizza 16 nodi nel primo strato e 4 nel secondo strato. Con queste caratteristiche è stata ottenuta la seguente matrice di confusione:

Prediction	Reference 0	1
0	139	41
1	53	367

e le seguenti misure di performance:

```
               Accuracy : 0.8433
                 95% CI : (0.8117, 0.8715)
    No Information Rate : 0.68
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.634
```

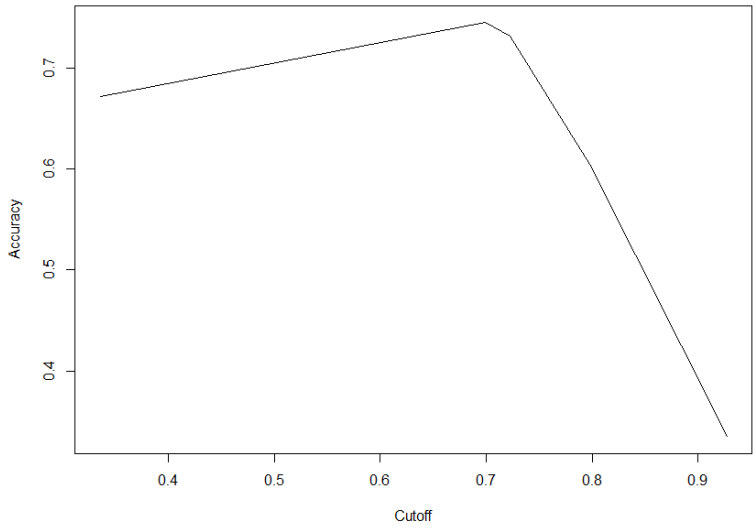
Il valore di Kappa è abbastanza soddisfacente e anche il valore di accuratezza è alto. Per avere una visione completa, calcoliamo anche i valori di performance tramite *macro average* ed otteniamo che il valore di *precision* è 0,8230 il che significa che una grande percentuale di osservazioni classificate come positive sono effettivamente positive. Il valore di *recall* è pari a 0,8117 ed anche questo è un valore che indica la bontà del modello così come l'*F1-Measure* che è pari a 0,8168.

In conclusione, nonostante la rete sia stata applicata ad un numero limitato di istanze del dataset originale, possiamo dedurre che è un buon modello di classificazione per le istanze prese in considerazione.

Misura di performance

Per analizzare le misure di performance dei vari modelli studiati osserviamo il grafico che permette di trovare il punto di *cutoff* che ottimizza la metrica di *accuracy*.

Per il modello degli *alberi decisionali* il grafico ottenuto è il seguente:

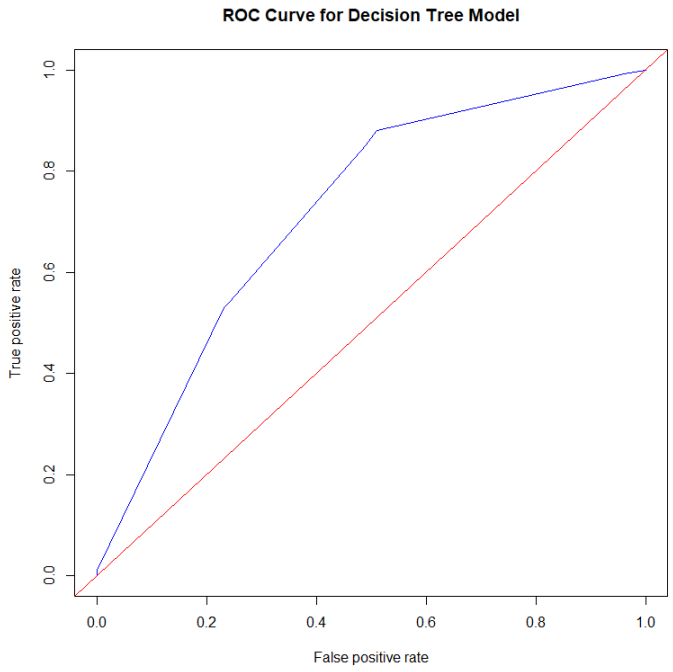


Questo grafico indica che si ottiene un valore di accuratezza migliore quando il *cutoff* è pari a circa 0,7.

Per misurare la bontà del modello, è possibile calcolare anche la curva ROC che rappresenta la relazione tra la vera positività e la falsa positività per tutti i possibili valori di soglia di cutoff. Con "vera positività" si intende il rapporto tra il numero di positivi veri identificati dal modello e il numero totale di positivi nel testset. La "falsa positività" è il rapporto tra il numero di falsi positivi identificati dal modello e il numero totale di negativi nel testset.

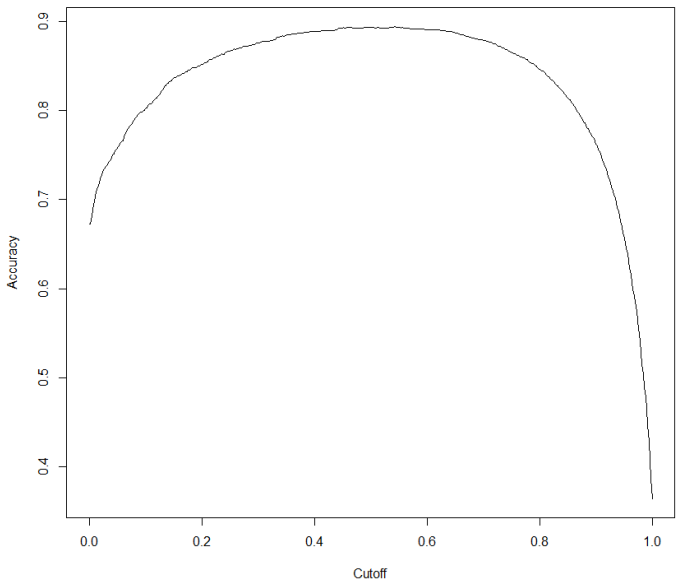
La curva ROC mostra come varia la vera positività a seconda della falsa positività per tutti i possibili valori di cutoff. Un modello ideale avrà una curva ROC che si avvicina il più possibile al punto in alto a sinistra della figura, ovvero una falsa positività di zero e una vera positività di uno. Una curva ROC che segue la diagonale rappresenta un modello che non è in grado di distinguere tra positivi e negativi.

La curva ROC del modello di classificazione *decision tree* è la seguente:



Come già confermato dai dati precedente ottenuti, il modello di predizione degli alberi decisionali non è molto preciso nella scelta del target corretto.

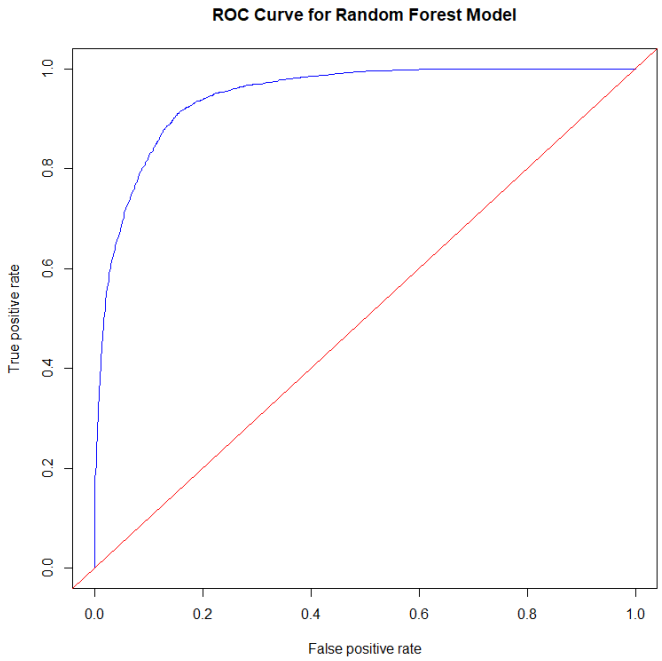
Verifichiamo quindi gli stessi valori anche per il secondo modello di predizione utilizzato, ossia i *random forest*. Il grafico che misura il valore di *cutoff* in relazione alla metrica di accuratezza è il seguente:



Osservando il grafico, il valore di *cutoff* ottimale è attorno a 0,5. Con questo valore, l'accuratezza assume il valore ottimale per il modello pari a 0,9.

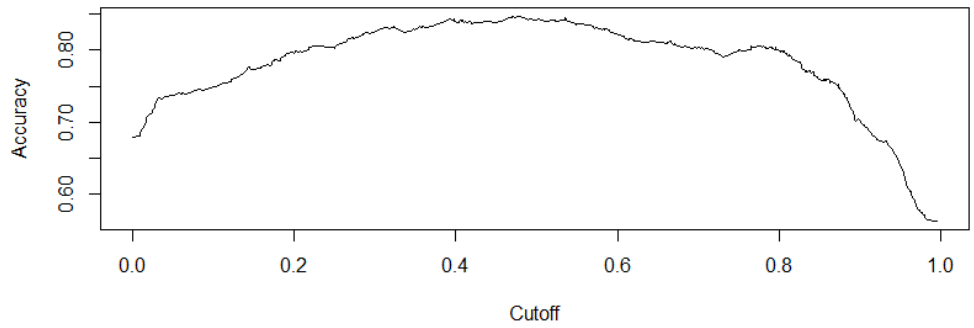
Solo osservando questo grafico possiamo dedurre che il modello decisionale *random forest* è nettamente migliore rispetto agli alberi decisionali.

La stessa conclusione è deducibile anche osservando la curva ROC relativa al modello:



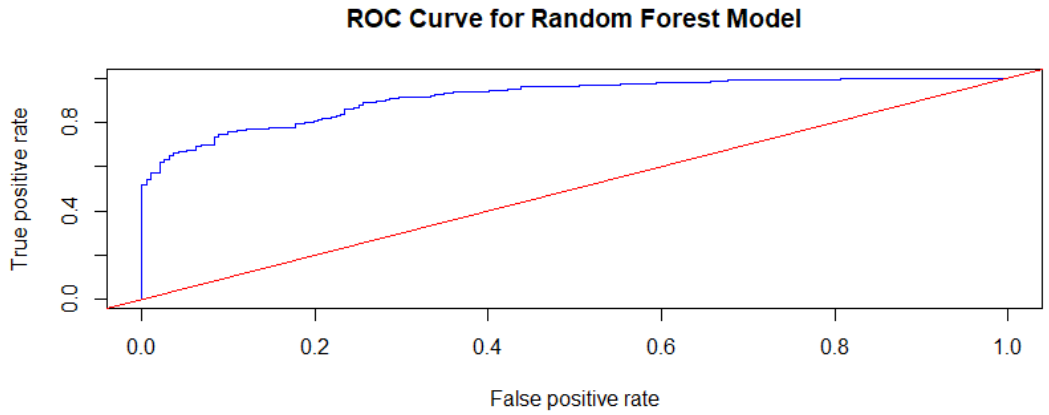
Dato che la curva si avvicina molto a quella ottimale, possiamo dedurre che il modello è ottimo nel classificare correttamente le istanze deducendo che il modello *random forest* ha alta precisione e un'alta capacità di discriminazione.

Anche per il terzo modello di classificazione utilizzato, ossia le *reti neurali*, abbiamo calcolato il grafico che misura il valore di *cutoff* in relazione alla metrica di accuratezza:



Osservando il grafico, è possibile notare che il valore di *cutoff* ottimale si aggira intorno a 0,5 e corrisponde ad un’accuratezza pari a 0,84.

La curva ROC ottenuta è la seguente:



Come osservato nel grafico del modello *Random forest*, la curva si avvicina molto a quella ottimale.

Per capire quale dei due modelli ha curva ROC migliore, cioè quale modello è in grado di fare migliore distinzione tra le due classi del target, è utile calcolare l'area sottesa alla curva (AUC): per i *random forest* il valore è pari a 0,9455 mentre per il modello di classificazione delle *reti neurali* il valore è pari 0,9133.

Possiamo quindi concludere che il modello *random forest* e il modello *neural network* sono dei buoni classificatori per il nostro dataset.