UNIVERSITA DEGLI STUDI DI PADOVA

Facoltà di Scienze Statistiche

Corso di Laurea triennale in Statistica, Tecnologie e Informatiche

Relazione finale:

Stima della durata dei “ticket” tramite metodi di analisi della sopravvivenza

Relatore: Dott. Bruno Scarpa

Laureando: Masin Marco

Matricola:599882-STI

Anno accademico 2010/2011

Un ringraziamento speciale

Ai miei genitori, Francesca, Shao, Fef, Batta, Cava, Cus, Ciano, Federico Longo e a tutti i miei più cari amici che mi sono stati vicini durante la mia carriera universitaria.

Sommario

PREMESSA

La prima parte del elaborato riguarderà la presentazione dell'azienda Wintech S.p.a di Padova e del programma QlikView con cui ho costruito il progetto di tesi durante il mio stage.
La seconda parte riguarderà l'analisi dei dati aziendali.
Lo scopo della relazione finale sarà quello di analizzare e capire quali sono i fattori che influenzano la durata dei ticket aziendali applicando tecniche di analisi di dati di sopravvivenza, adottando dei modelli parametri e semiparametrici. Inoltre ho voluto proporre una soluzione non parametrica attraverso l'uso delle Random Survival Forest.

CAPITOLO 1
PRESENTAZIONE DELL' AZIENDA E QLIKVIEW

Nel mio percorso di studi ho avuto l'opportunità di effettuare uno stage di formazione presso la Wintech S.p.a., un'azienda informatica che da venticinque anni appoggia le persone nella gestione tecnologica.
La società Wintech S.p.a. è composta da tre gruppi di aziende che sono Wintech stessa, Format e Albasoft.
L'attività aziendale consiste nel fornire consulenza personalizzata, soluzioni applicative e tecnologiche in ambito IT per ottimizzare i processi aziendali dei clienti.
Questa azienda è organizzata principalmente in quattro Business Unit :

1. **B.U. Sviluppo** : si occupa dell'analisi e sviluppo di applicazioni software in ambito web su piattaforme IBM e Microsoft.
2. **B.U. Assistenza** : si occupa di supporto sistemistico.
3. **B.U. Commerciale** : si occupa della vendita e stipulazioni di contratti per prodotti hardware e software
4. **B.U. Sistemi:** si occupa dello sviluppo e del supporto alle applicazioni gestionali sviluppate dall'azienda Sistemi Spa.

La mia esperienza lavorativa si è svolta all'interno della B.U. sviluppo e il mio obiettivo era quello di creare un nuovo sistema di analisi delle statistiche di vendita, acquisto e flussi di cassa applicando tecniche di Business Intelligence che utilizzino software QlikView.
Con il termine Business Intelligence ci si riferisce ad un insieme di processi aziendali per raccogliere e analizzare dati con lo scopo di supportare le decisioni aziendali.
Lo scopo finale è quello di trasformare i dati, provenienti dai vari sistemi di contabilità, produzione e CRM (*Customer Relationship Management*), in informazione e conoscenza.
Il concetto di CRM o Gestione delle relazioni con i clienti, in ambito economico, definisce il fine delle operazioni di *marketing* per il mantenimento dei clienti e per garantire loro il più alto grado di soddisfazione del prodotto.
Le applicazioni informatiche CRM hanno l'obiettivo di tenere a stretto contatto con i clienti, inserire e analizzare le loro informazioni nel database.
Le persone che lavorano nell'ambito del Business Intelligence utilizzano delle opportune applicazioni per raccogliere, analizzare, modellare e distribuire le informazioni provenienti dai sistemi citati in precedenza: QlikView fa parte di questi programmi.
Questa applicazione ha una particolare tecnologia "*in-memory*": il programma lavora completamente su ram e questo permette che i dati provenienti da diverse sorgenti si combinino tra loro rapidamente.
Tramite l'utilizzo di vari connettori è possibile integrare informazioni provenienti da diverse fonti o sistemi in modo da permettere l'acquisizione di una unificata e coerente visione generale dei dati su diversi DBMS centralizzati o distribuiti.

La lettura dei dati nei sistemi classici è spesso un compito complesso che richiede la conoscenza estesa della struttura del database e della sintassi del linguaggio di query.
QlikView rimuove questi limiti consentendo di effettuare selezioni libere sui dati visualizzati a video semplicemente con un click del mouse.
Le applicazioni QlikView sono solitamente suddivise in più fogli di lavoro dove i dati sono rappresentati a discrezione dell'utente.
I valori sono selezionabili anche quando rappresentati in maniera grafica, ed ogni singola vista sui dati può essere il punto di partenza di un'analisi.
La rappresentazione grafica delle informazioni può avvenire attraverso:

- Tabelle semplici/pivot;
- Liste;
- Istogrammi;
- Cruscotti;
- Diagramma a torta /dispersione/a scatola;

In sintesi la presentazione dei dati può avvenire :

1. in modo sintetico con l'uso di un grafico.
2. in modo dettagliato lasciando la possibilità all'utente di procedere con esplorazione dei dati in *drill-down*.

QlikView garantisce il massimo grado di esportabilità dei dati verso più comuni formati come documenti Xml, Fogli Excel, documenti Html e molti altri; inoltre risponde alle esigenze di sicurezza per ogni tipo di utente attraverso un controllo degli accessi basato sull'immissione di un username e password.
E' cosi possibile definire profili differenti per amministratori e utenti.
Definendo e gestendo opportune tabelle di sicurezza è possibile quindi creare un collegamento logico tra un utente e uno specifico sottoinsieme di dati: in questo modo utenti diversi potranno visualizzare schermate di dati diverse e personalizzate.
Un opportuno esempio per capire meglio il concetto di fondo potrebbe essere il seguente: prendiamo in considerazione una grande multinazionale suddivisa in più filiali in diversi paesi.
L'utente amministratore sarà l'amministratore di rete e potrà vedere le informazioni di tutte le sedi sparse per il mondo, mentre ogni capo-filiale (utente *User*) vedrà esclusivamente i dati riguardanti la propria sede.
Il vantaggio più grande di questa applicazione, è che si possiede il completo controllo di tutte le informazioni riguardanti clienti, fornitori, vendite, prodotti e bilanci in un'unica applicazione facilmente accessibile, per un'analisi avanzata della gestione aziendale.
Uno dei svantaggi è che le analisi effettuate dei dati presi in esame, sono comunque di natura descrittiva e non è possibile applicare dei modelli inferenziali per fare test e ricavare delle previsioni su i dati.
Siamo comunque riusciti a risolvere il problema, agganciando QlikView ad R tramite comandi *DOS*.
Il concetto è molto semplice:
QlikView manda un file contenete i dati ad R, il quale li legge, ci applica un opportuno modello/test statistico definito nella sintassi dello script R. Fatto questo R salva i risultati ottenuti su un nuovo file e li invia a QlikView, che li legge e li rappresenterà a schermo a discrezione dell'utente.
In questo modo si riesce a visualizzare i risultati inferenziali con una grafica 2D / 3D in alta definizione, ottenendo cosi un'applicazione di Business Intelligence completa che tiene conto sia degli aspetti inferenziali sia di quelli descrittivi.

L'applicazione ottenuta tramite l'interfacciamento tra i due programmi non potrà essere automatica al 100%, perché comunque c'è sempre bisogno di una figura professionale come lo statistico che controlli che il modello adottato continui a rappresentare bene il fenomeno preso in considerazione con l'andare del tempo.

Come ribadito in precedenza l'obiettivo dello stage era quello di realizzare un'applicazione di Business Intelligence in QlikView che fosse in grado descrivere al meglio l'andamento e la gestione dell'attività interne ed esterne svolte dall'azienda.

I dati, aggiornati quotidianamente oggetto di analisi, provenivano principalmente da due sorgenti:

1. Geco
2. Kayako

Sono due software costruiti su due database relazionali con compiti diversi, ma collegati tra loro.

GECO

Geco (Gestione Consuntivazione) è un software costruito internamente all'azienda che si appoggia ad un database relazionale e il suo compito è quello di gestire e consuntivare le attività in corso legate a commesse.
Per comprendere al meglio il compito principale di Geco bisogna introdurre alcuni concetti di base partendo direttamente dallo schema del database, spiegando passo per passo tutte le entità che lo compongo e sottolineando i principi che stanno alla base dell'applicazione.

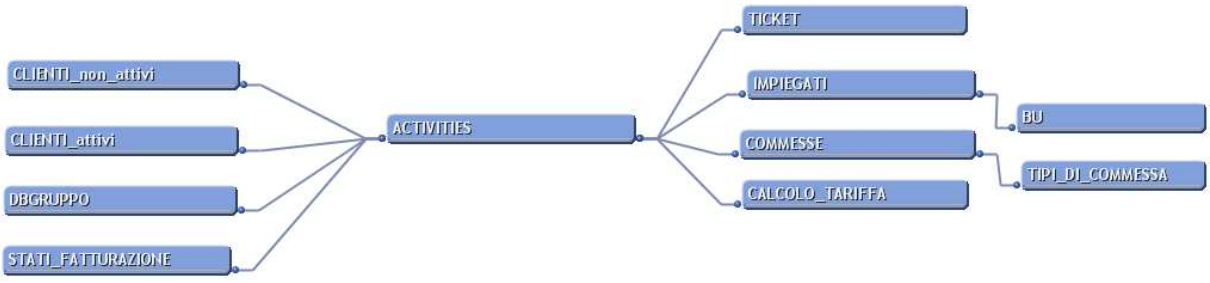


Grafico 1:
Lo schema sopra rappresentato descrive alcune principali tabelle del DBMS di Geco dove si è sviluppata l'applicazione Qlikview

ACTIVITIES: tabella che contiene le attività quotidiane svolte da un impiegato per zero o più clienti.

Le attività possono essere :

1)**interne** se sono svolte da un dipendente all'interno dell'azienda e non producono sicuramente del fatturato; ad esempio tutte le attività di contabilità oppure le attività legate all'assistenza ai server interni.

2)**esterne** se sono svolte da un dipendente per un cliente e possono produrre del fatturato aziendale; ad esempio lo sviluppo di un'applicazione oppure un contratto di assistenza con una società. In questo caso l'attività ha sempre una tariffa oraria.

Possono essere legate ad una o più commesse e possono essere collegate a zero o più ticket. Inoltre hanno sempre uno stato di fatturazione e un DBGruppo (Format, Abasoft ,Wintech).

IMPIEGATI: tabella che contiene tutte le informazioni anagrafiche relative ad un impiegato che lavora in azienda. Un impiegato è sempre legato ad una o più *Business Unit* e può svolgere una o più attività.

B.U: tabella che contiene le informazioni relative alle *Business Unit* di Wintech ed una B.U. può avere uno o più impiegati al suo interno.

COMMESSA: tabelle che contiene tutte le informazioni relative ad un ordine richiesto all'azienda, è composto da una o più attività legate ad uno o più clienti. Inoltre una commessa è sempre legata ad un solo 'tipo di commessa' .

TIPO DI COMMESSA: tabella che contiene le informazioni relative ai tipi di commessa, ad esempio commesse interne, commesse contratti, commesse consultivi, commesse spot ecc.

TARIFFA: tabella che contiene tutte le informazioni riguardanti le tariffe di un attività.

CLIENTI_ATTIVI : tabella che contiene tutte le informazioni anagrafiche riguardo al pacchetto clienti di Wintech.

CLIENTI_non_ATTIVI: tabella che contiene tutte le informazioni riguardanti agli ex clienti.

TICKET: tabella che contiene tutte le informazioni riguardanti i ticket aziendali: contiene sia informazioni proveniente da Geco sia quelle provenienti da Kayako e rappresenta il legame tra le due applicazioni.

Obiettivi di Geco

Grazie a questa struttura Geco è in grado di monitorare la situazione aziendale nell'arco del mese, definendo così un trend mensile e riuscendo a constatare quello che sta accadendo e quello che è accaduto in modo da delineare uno stato di avanzamento dei vari progetti. Questo non è possibile con una comune applicazione gestionale, perché le commesse vengono fatturate solo alla fine del mese e non possono avere nessuna indicazione sull'andamento delle attività in corso.

KAYAKO

Kayako è un sistema per la gestione delle richieste di assistenza ed è stato sviluppato dall'azienda indiana ***Kayako Infotech Ltd*.**

Viene utilizzato in Wintech per il controllo e la risoluzione dei ticket aziendali che attraverso algoritmi interni gestisce le date di aperture e di chiusura dei ticket in modo automatico. Tutte le volte che un cliente riscontra un problema hardware o software e quindi ha bisogno di assistenza/teleassistenza, l'azienda apre un ticket e lo chiude successivamente alla risoluzione del problema. In questa applicazione sono salvate tutte le informazioni riguardanti gli stati dei ticket (aperti/chiusi), le priorità di risoluzione, lo stato di avanzamento e chi ha partecipato a risolvere il problema.

Geco e Kayako sono strettamente collegati tra di loro e in questo modo si riesce a risalire a chi ha svolto le attività legate al ticket, quanto sono durate e chi è il cliente che ha aperto un ticket ottenendo cosi un informazione più dettagliata.

Cenni sulla struttura del progetto QlikView

La struttura del progetto che ho realizzato è divisa sostanzialmente in tre parti:

1. Gestione delle commesse e dell'attività ad esse collegate (fonte primaria Geco)
2. Gestione dei ticket relativi alla B.U. "assistenza tecnica" (fonti Geco e Kayako)
3. Gestione dei ticket relativi alle altre B.U. (fonte Kayako).

Nell'analisi dei dati abbiamo riscontrato un problema nella gestione dei ticket aziendali: ci sono diverse modalità di risoluzione dei ticket da parte delle *Business Unit*.

Per la B.U. "assistenza tecnica" la risoluzione dei ticket avviene con il supporto dei due programmi, registrando di fatto le attività. Per le altre B.U. la risoluzione dei ticket avviene esclusivamente con l'uso di Kayako, quindi abbiamo deciso di dividere il problema in due macroaree per evitare di dover analizzare dei dati anomali che potevano variare completamente l'analisi dei dati.

Per la gestione dei ticket le due applicazioni risultano comunque simili nella struttura dei fogli di lavoro, anche se per le altre B.U. non è presente una figura di utente risolutore non essendoci il collegamento con Geco.

La pagina "Principale" da una visione generale dell'andamento della durata dei ticket, dando anche delle indicazioni sul mese corrente di quanti ticket sono stati aperti/chiusi e la durata media del mese. Nel foglio "Cliente " vengono mostrati dei grafici che riassumo le informazioni di durata dei ticket, del numero di ticket aperti e chiusi per ogni cliente. Inoltre è presente un foglio "Andamento Storico" che mi fornisce le informazioni di apertura/chiusura e durata dei ticket nel tempo. In fine solo nell'applicazione per la B.U. assistenza tecnica è presente un foglio"

Informazioni Impiegati e Attività" dove vengono fornite in dettaglio le informazioni riguardanti le attività svolte in un ticket da un impiegato.

Per quanto riguarda l'applicazione riguardante la gestione delle attività e delle commesse abbiamo sempre una pagina 'Principale' dove sono contenuti alcuni dati di sintesi sulle percentuali di commesse per B.U. e per tipo di commessa. In aggiunta dato che da un comune gestionale possiamo solo avere informazioni relative al fatturato solo alla fine del mese, con questa applicazione QlikView riusciamo a ad avere un quadro generale delle ore svolte da un dipendente che devono essere ancora fatturate.

E' stata dedicata una pagina ai contratti perché si abbia un indicazione aggiornata sul numero ore svolte e sul monte ore prefissato. Infine è presente un foglio per i clienti di Wintech dove sono state inserite le informazioni in dettaglio riguardo alle Province, Paesi e Regioni fornendo anche un grafico che si collega a *Google Maps* per la geo-localizzazione del cliente e in aggiunta un mappa dove sono visualizzate le province italiane in diversi colori a seconda del numero di clienti presenti.

CAPITOLO 2
ANALISI DEI DATI

Obiettivi

Si vuole studiare il tempo di sopravvivenza dei ticket aperti e chiusi dalla B.U. assistenza tecnica. In particolare si vuole capire quali variabili possano influire sul tempo di risoluzione di un ticket dando cosi un supporto alle decisioni aziendali.

I Dati

Descrizione dei dati e delle variabili

Il data set analizzato proviene dall' azienda Wintech S.p.a. di Padova dalla B.U. assistenza. I dati consistono in misurazioni effettuate su ticket aziendali. Un ticket viene aperto dall'assistenza tecnica ogni volta che un cliente ha riscontrato un problema hardware. Il campione preso in considerazione fa riferimento ad un periodo chiuso che va dal 1 aprile 2011 fino al 30 giugno 2011.

status è una variabile categoriale che assume 2 modalità:

- 0 indica che un ticket non è ancora stato risolto, quindi è ancora aperto;
- 1 indica che un ticket è stato risolto, quindi è chiuso;

time indica il tempo (in giorni lavorativi) di durata del ticket dalla data di apertura del ticket fino dalla sua data di chiusura. Se un ticket non è ancora stato risolto e quindi non si conosce la reale durata, verrà considerata come data di chiusura il 30 giugno 2011. Insieme a status costituisce la nostra variabile di risposta.

IsPhoneCall è una variabile dicotomica che indica se un ticket è stato aperto via e-mail o via telefono (0 e-mail, 1 telefono).

Giorno è una variabile qualitativa che indica il giorno lavorativo in cui è stato aperto il Il ticket. Può assumere cinque modalità: lun, mar, mer, gio, ven.

Mese è una variabile qualitativa che indica il mese in cui è stato aperto il ticket (aprile, maggio, giugno).

Cat.clienti è una variabile qualitativa che indica il cliente che ha chiesto di risolvere il ticket e assume tre modalità:
-A cliente Wintech
-B cliente Wintech
-C altri clienti Wintech.

A e B rappresentano i due principali clienti di Wintech (per ragione di privacy non posso riportare il nome reale) e per questo nella mia analisi ho deciso di considerarle in due categorie distinte separate da tutti gli altri clienti.

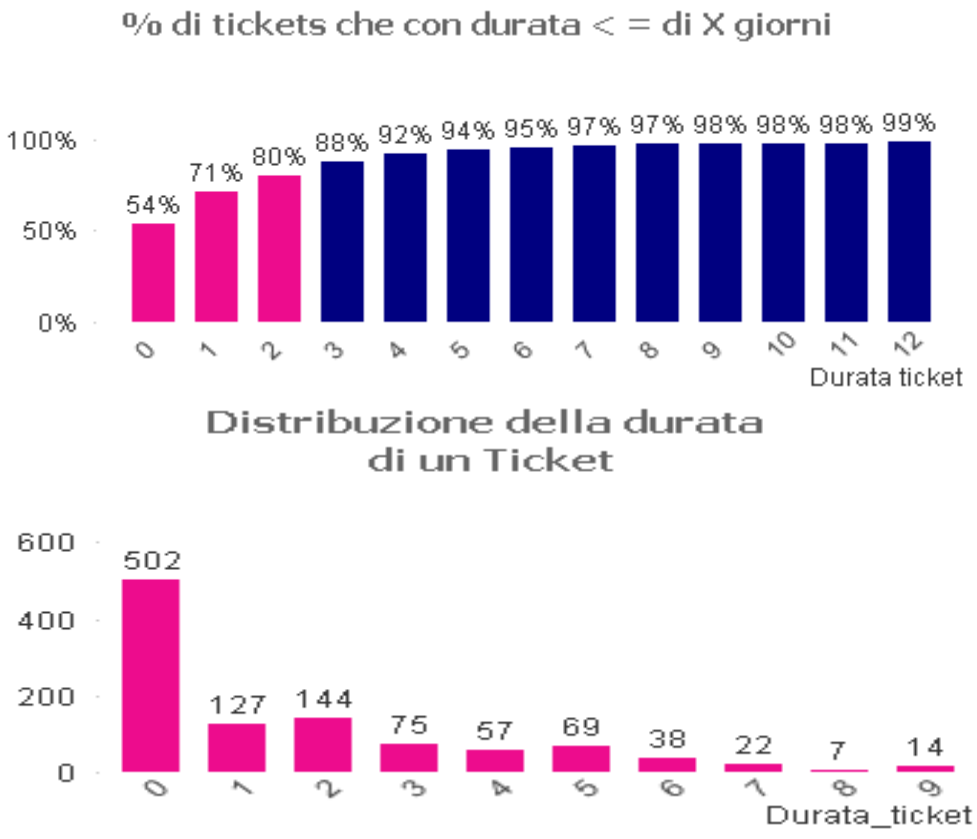
Tr è una variabile dicotomica che indica se è stata fatta almeno una trasferta per risolvere il ticket.

Complessivamente il 6% dei dati sono censurati (*status* uguale a 0) ed il rimanente 94% sono dati completi.
Il numero dei dati censurati è minore rispetto al numero dei dati completi. Questo non dovrebbe causare problemi nel costruire i modelli per fare inferenza sui dati presi in esame.

Analisi Esplorativa

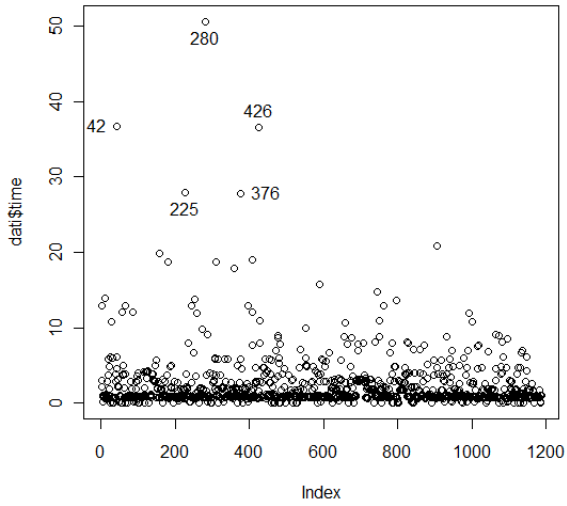
-Time

Grafico A



I due grafici riportati sopra rappresentano la durata dei ticket chiusi.

Il 1° grafico sottolinea il fatto che al massimo 80% dei ticket aziendali vengono chiusi in due giorni. La distribuzione della durata dei ticket è fortemente asimmetrica, quasi tutte le osservazioni sono concentrate nell'intervallo [0,2].

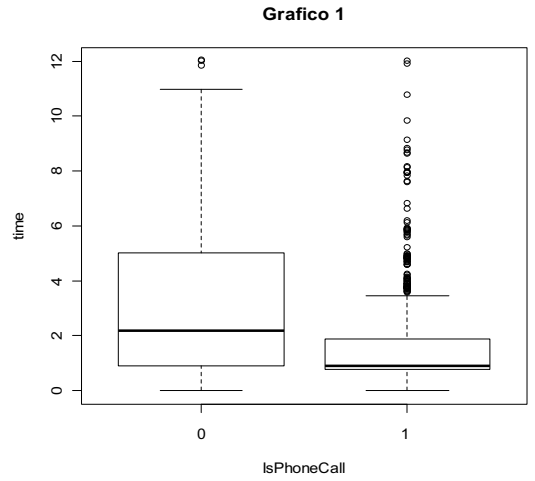


Prima di effettuare l'analisi voglio precisare che ho escluso cinque valori che nel grafico corrispondono alle osservazioni 42,225,376,426,280. Per prendere questa decisione sono andato a chiedere spiegazioni ai tecnici della B.U. assistenza tecnica: il ritardo di questi ticket è dovuto a problemi legati alla gestione di pezzi di ricambio, quindi, essendo casi anomali, ho deciso di toglierli dall'analisi.

Vediamo come è distribuito il tempo di durata dei ticket rispetto alle variabili elencate in precedenza.

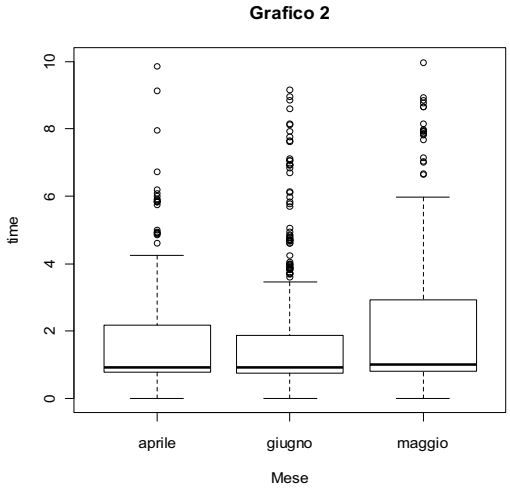
- IsPhoneCall

Il data set presenta il 20% circa di ticket aperti via e-mail ed il restante 80% via telefono.
Si tratta dunque in una situazione non bilanciata



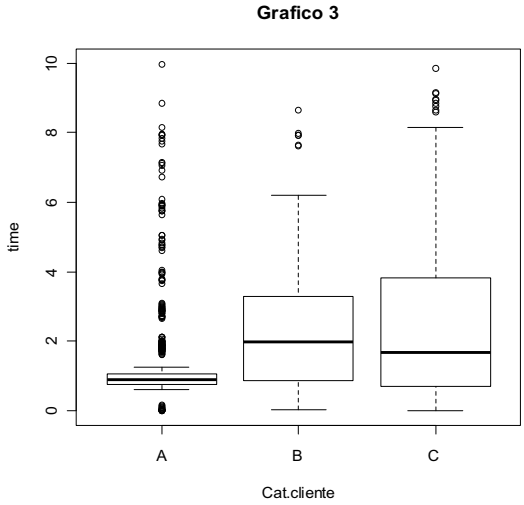
Dal grafico 1 si può notare che i ticket che sono stati aperti via telefono(1) hanno una durata inferiore rispetto ai ticket aperti via e-mail (0)

-Mese



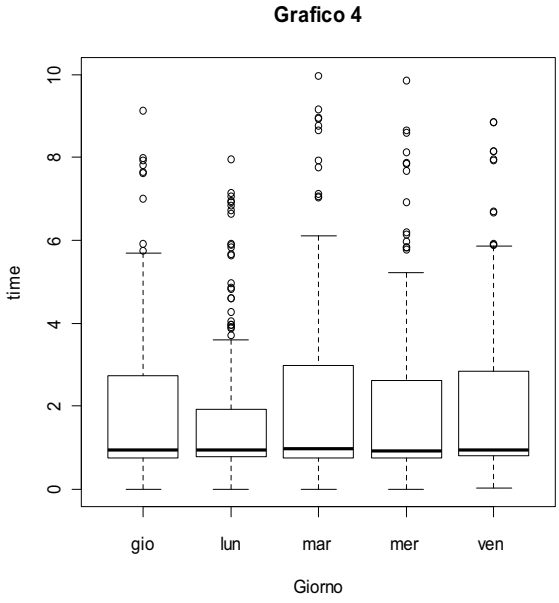
Dal seguente grafico possiamo ipotizzare che il mese di apertura non influisce sulla durata di un ticket, non ci sono differenze evidenti per i tre mesi in mediana, mentre c'è una differenza in varianza tra i tre mesi.

-Categorie Cliente



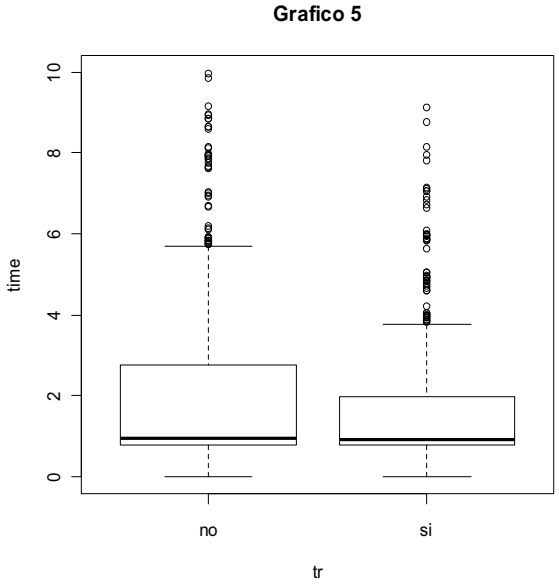
Dal grafico 3 si può evincere una differenza in mediana e varianza per le tre categorie dei clienti. In particolare cat A ha una durata dei ticket molto più bassa rispetto a cat B e altri clienti (cat C). Inoltre le categorie B e C sembrano essere praticamente uguali.

-Giorni



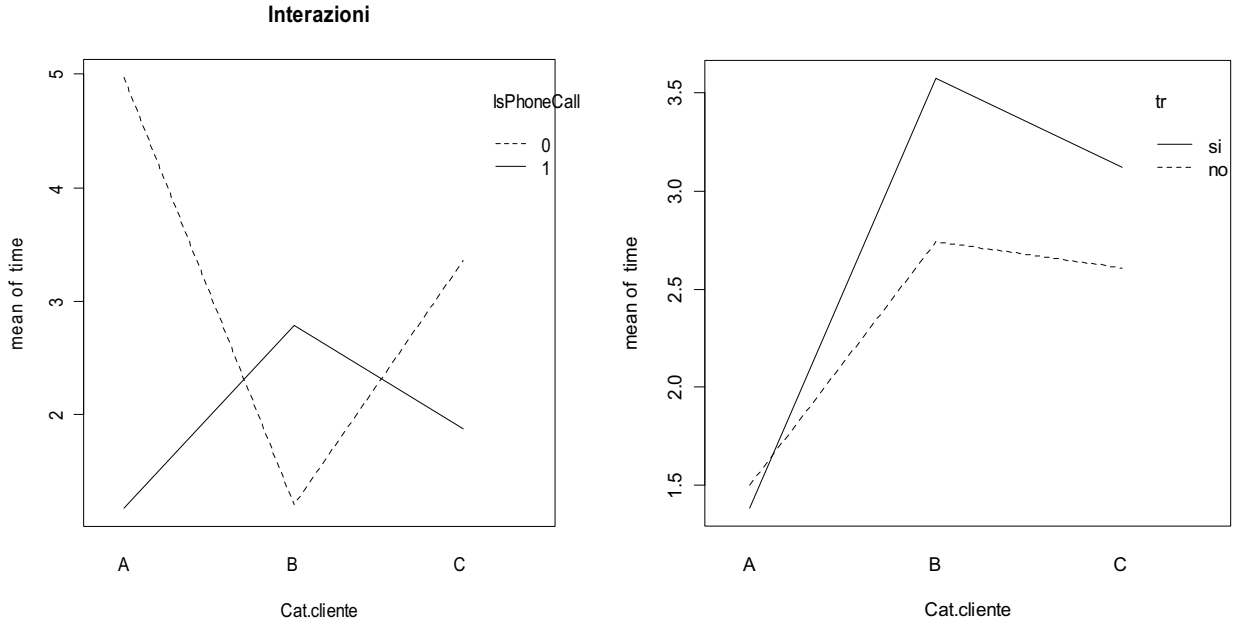
Dal grafico 4 non si riscontrano differenze in mediana, mentre in varianza c'è una leggera differenza.

-Trasferta



Dal grafico 5 si può notare che non c'è una differenza in mediana tra le due categorie,mentre c'è una leggera differenza in varianza.

Valutazione di eventuali interazioni



I due grafici riportati sopra rappresentano degli *interaction plot*: il primo tra la variabile IsPhoneCall e Categorie Cliente, mentre il secondo tra trasferta e Cat.Cliente.
Da questi due grafici posso supporre che ci sia un' interazione tra Cat.Cliente e IsPhoneCall e tra Cat.Cliente e trasferta,quindi potrei considerare l'interazione nei modelli che andrò a costruire.

Da questa prima analisi esplorativa da quanto visto nei *box-plot* in precedenza posso supporre che le variabili Priorità e Cat.cliente siano le più influenti sul tempo di sopravvivenza dei ticket.

Metodologie utilizzate per l'analisi

Per analizzare la durata di risoluzione dei ticket utilizzeremo metodi e stime sia non parametriche, sia semi-parametriche che parametriche. Per maggiori approfondimenti si rimanda Thereau T.M. e Grambsh P.M. (2000) Modeling Survival Data: Extending the Cox model, Usa, Spring-Verlag.

Kaplan-Meier

Una stima non parametrica della funzione di sopravvivenza $S(t)$ calcolata nell'istante di tempo t è $\hat{S}_{KM}(t)$, stima di Kaplan-Meier, che è fortemente consistente e asintoticamente normale (sotto condizioni molto deboli).
Cioè:

$$\hat{S}_{KM}(t) \xrightarrow{q.c.} S(t)$$

$$\sqrt{n}\left(\hat{S}_{KM}(t) - S(t)\right) \xrightarrow{d} N(0, \sigma^2)$$

Dove σ^2 può essere stimato mediante la formula di Greenwood (si rimanda ai riferimenti bibliografici per la formula esatta).
Tramite essa, per ogni covariata, analizzeremo marginalmente le $\hat{S}_{KM}(t)$ e ricaveremo anche le stime delle funzioni di rischio cumulate $\widehat{H}(t)$ tramite la relazione $\mathrm{H(t)} = -\log[\mathrm{S(t)}]$ (dove al posto di $\mathrm{S(t)}$ utilizzeremo la sua stima)
Queste ultime stime ci saranno utili soprattutto per vedere se i rischi sono proporzionali tra loro che è un assunto del modello semi-parametrico di Cox .

Test Log-Rank

Definiamo con "morte" la realizzazione del evento di interesse dello studio che nel nostro caso è la chiusura di un ticket aziendale.
Il test Log-Rank, proposto da Mantel viene utilizzato per la verifica dell'ipotesi
$\mathrm{H_0}: \mathrm{S_1(t)} = \mathrm{S_2(t)} = \cdots = \mathrm{S_n(t)}\ \forall \mathrm{t}$, contro l'ipotesi alternativa $\mathrm{H_1}: \mathrm{almeno\ una\ diversa}$.
(n è il numero di popolazioni), La statistica test nel caso si vogliano confrontare *n+1* popolazioni è la seguente:

$$Q = \sum_{i=0}^{n} \frac{(O_i - A_i)^{\ 2}}{A_i} \sim X^2_{n-1}$$

Dove O_i sono la somma del numero di morti osservate di tutti gli istanti di morti distinti, mentre le A_i rappresentano il numero di morti attese sotto Ho di tutti gli istanti di morti distinti.
Il test Log-Rank raggiunge la massima potenza quando i rischi sono proporzionali tra loro.
Consideriamo il caso n=2 popolazioni e prendiamo due possibili casi.

Grafico A

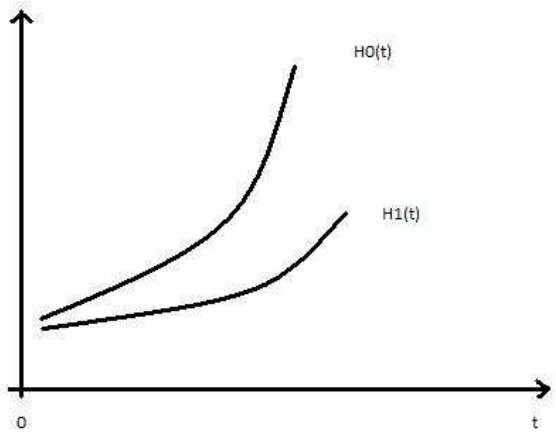
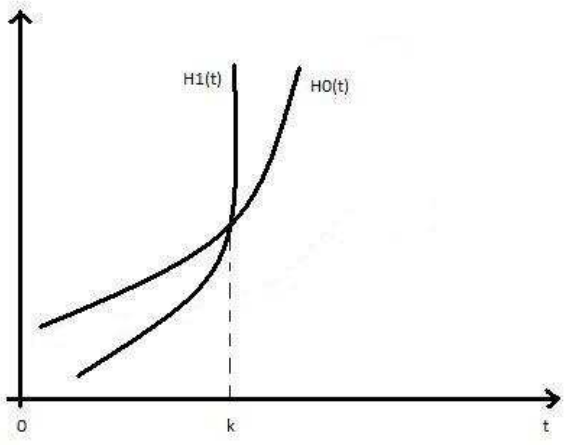


Grafico B



Nel grafico A si può capire che in ogni istante t di morte osservata per ogni gruppo il numero osservato di morti è maggiore rispetto al numero di morti attese sotto Ho. Quindi per un generico gruppo *i* la somma $O_i - A_i$ sarà composta da addendi tutti positivi e quindi Q assumerà un valore grande portando cosi al rifiuto dell'ipotesi Ho.

Nel grafico B si può capire che per un generico gruppo i

- Il numero osservato di morti sarà maggiori del numero atteso di morti sotto Ho per ogni istante di morte minore di *k*.
- Il numero osservato di morti sarà minore del numero atteso di morti sotto Ho per ogni istante di morte minore di *k*.

Quindi per un generico gruppo *i* la somma $O_i - A_i$ sarà composta da addendi positivi e negativi che possono compensarsi a vicenda facendo cosi assumere a Q un valore piccolo portando cosi ad accettare l'ipotesi Ho.

Modelli a tempi accelerati

Un tipo di modelli che utilizzerò nell'analisi, saranno i modelli parametrici a tempi accelerati, dove si assume che le covariate abbiano l'effetto di accelerare (ridurre) o prolungare i tempi.
Si assume pertanto che:

$$T|\boldsymbol{z} = \frac{T_0}{\mu(\mathbf{z};\boldsymbol{\beta})}$$

Dove $\mu(\mathbf{z};\boldsymbol{\beta})$ è una funzione positiva e tale che $\mu(\boldsymbol{z}_0;\boldsymbol{\beta}) = 1$.
Quando $\mu(\mathbf{z};\boldsymbol{\beta}) > 1$ le variabili esplicative riducono i tempi di sopravvivenza e quando $\mu(\mathbf{z};\boldsymbol{\beta}) < 1$ li prolunga (assumiamo $\mu(.) = \exp(.)$).
L'effetto delle covariate, quindi, è solo quello di cambiare la scala dei tempi.
$T_0 = \mathrm{T}|\boldsymbol{z}_0$, $\boldsymbol{z}_0 = \mathbf{0}$, è la variabile casuale durata in una categoria di riferimento e nell'analisi dei dati sceglierò per T una distribuzione *Weibull*, *Log-Logistica* e *Log-Normale* per vedere se i dati si adattano ad uno di questi modelli.

Per verificare l'adeguatezza dei modelli a tempi accelerati si stima la funzione di sopravvivenza partendo dall'esponenziale dei residui standardizzati e si cerca di ricondursi a una funzione lineare per il $\log(\mathrm{t})$. Se la distribuzione scelta per il modello si adatta bene ai dati allora il grafico di $\log_{\mathrm{S(t)}}(\mathrm{t})$ dovrebbe essere lineare.

Modello di Cox (Modello a rischi proporzionali)

Il modello semi-parametrico di Cox viene utilizzato per modellare la funzione di rischio h(t) e non si assumono ipotesi distributive di T, variabile casuale durata.
Il modello è costruito così:

$$h(t|\mathbf{z}) = h_0(t)\mu(\mathbf{z};\boldsymbol{\beta})$$

$\mathbf{z}$ è il vettore delle covariate del soggetto
$\boldsymbol{\beta}$ è il vettore dei coefficienti di regressione

$h_0(t) = h(t|\boldsymbol{z}_0)$ è la funzione di rischio di base calcolata in una determinata categoria di repressori.

$\mu(.)$ è la funzione predittore (anche per questi modelli in genere si utilizza $\mu(.) = \exp(.)$)

Può succedere però che i dati non supportino gli assunti di rischi proporzionali a causa di qualche fattore che ne determina lo scostamento.

In quel caso si può provare ad adattare il modello di Cox stratificato:

$$h_b(t|\mathbf{z}) = h_{0b}(t)\mu(\mathbf{z}; \boldsymbol{\beta})$$

con b l'indice del b-esimo strato

In sostanza, il modello assume che individui appartenenti a sottogruppi diversi possano avere rischi non proporzionali mentre individui appartenente allo stesso sottogruppo, con vettore dei regressori diversi, abbiano rischi proporzionali.

Notiamo inoltre che il modello di Cox assume che l'effetto delle variabili esplicative sia lo stesso in ogni strato (il vettore dei parametri $\boldsymbol{\beta}$ è lo stesso in ogni strato).

Per verificare l'adeguatezza del modello di Cox si guardano i residui di Cox - Snell che possiamo ottenere dai residui di martingala.

Se i rischi hanno una struttura di proporzionalità, mi aspetto che i residui di Cox – Snell si comportino come un campione censurato da una v.c. $Esp(1)$. Quindi, sapendo che la funzione di rischio cumulato di un Esp(1) è la bisettrice del primo quadrante, la funzione di rischio cumulato dei residui di Cox - Snell dovrebbe oscillare attorno a tale retta. Per maggiori approfondimenti si rimanda Thereau T.M. e Grambsh P.M. (2000) Modeling Survival Data: Extending the Cox model, Usa, Spring-Verlag.

Procedura adottata per la selezione del modello

Sia nel modello di Cox che nei modelli a tempi accelerati adotterò la strategia *backward* per scegliere quali variabili includere nel modello finale e quali escludere.

La strategia consiste nel partire da un modello completo che includa tutte le variabili, poi su ogni coefficiente stimato del vettore $\boldsymbol{\beta}$ si fanno dei test di nullità singolarmente.

Si eliminano dal modello le variabili una per volta, iniziando da quella cui corrisponde il più grande livello di significatività osservato purché sia più grande di una soglia prefissata (nel nostro studio scegliamo 0.05).

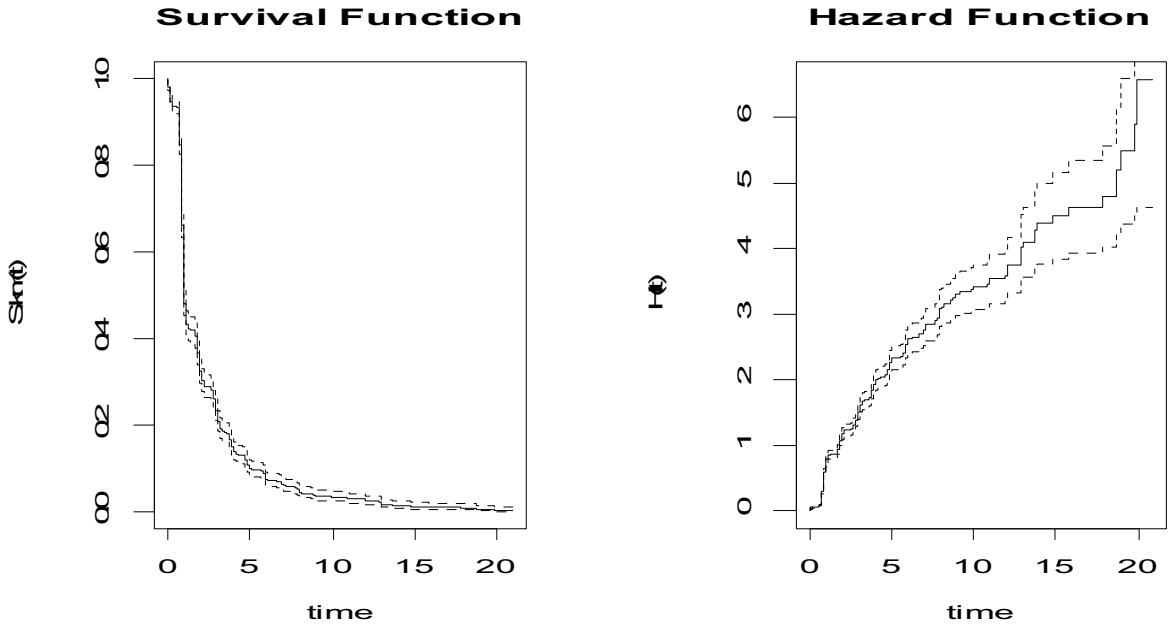
Non è detto però che eliminerò le variabili basandomi esclusivamente su questa strategia; se dall'analisi esplorativa una variabile risulta non discriminante potrò anche valutarne l'eliminazione dal modello. Ovviamente si valuterà anche la natura della variabile.

Analisi dei residui

Una volta scelto il modello, per vedere se è correttamente stimato, controllerò i residui di devianza (distribuiti asintoticamente come v.c. $N(0,1)$). Grazie ad essi potrò capire se ci sono errori commessi e se ci sono relazioni non colte tra la variabile di risposta e le variabili esplicative. Verranno controllati anche i residui beta per vedere se ci sono eventuali punti leva.
Per maggiori approfondimenti si rimanda Modeling Survival Data di Thereau T.M. e Grambsch P.M. (2000)

Analisi

Funzione di sopravvivenza e rischio cumulato

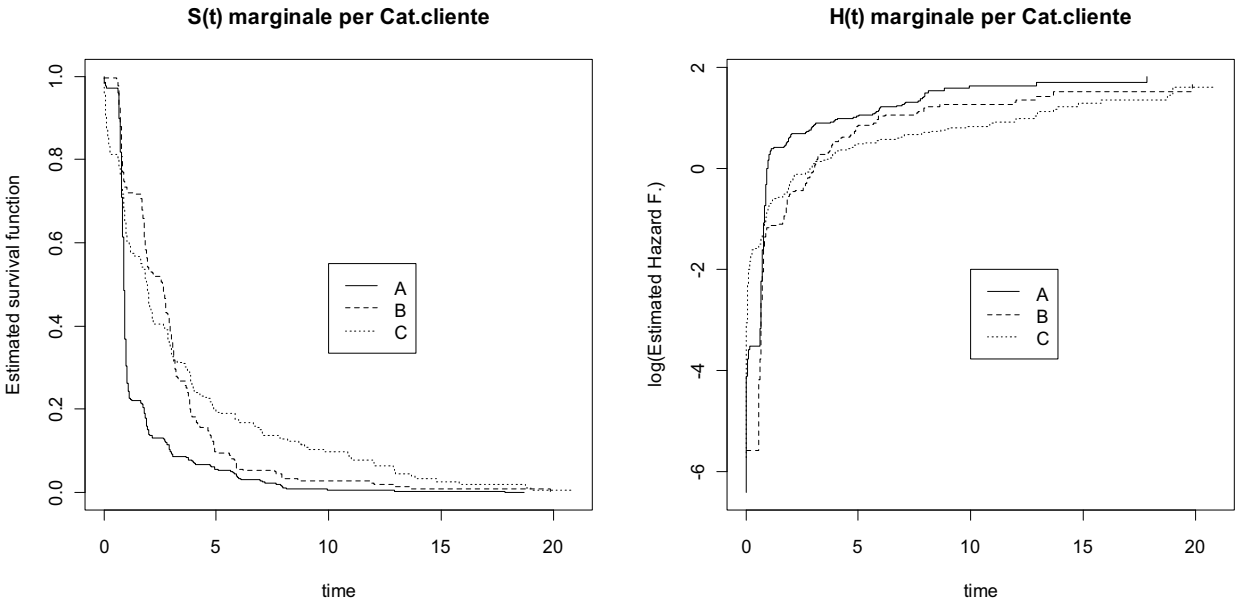


Dal grafico riportato a sinistra si osserva la funzione di sopravvivenza di un soggetto medio stimata attraverso lo stimatore di Kanplan-Meier con le relative bande di confidenza al 95% rappresentate dalle linee tratteggiate. La funzione di sopravvivenza è sempre decrescente esclusa l'ultima parte dove rimane costante. Dal grafico a destra si osserva la funzione di rischio cumulato: è una funzione crescente, escluso negli ultimi punti dove circa che è costante. Ricordo che le due funzioni sono legate tra di loro dal seguente legame : $H(t) = -\log[(S(t))]$. Lo stimatore di Kanplan-Meier è una stima non distorta di $S(t)$ e considera i ticket aperti e chiusi a differenza dei grafici prodotti a pagina 17 .

Alcune analisi marginali

-Categorie Cliente

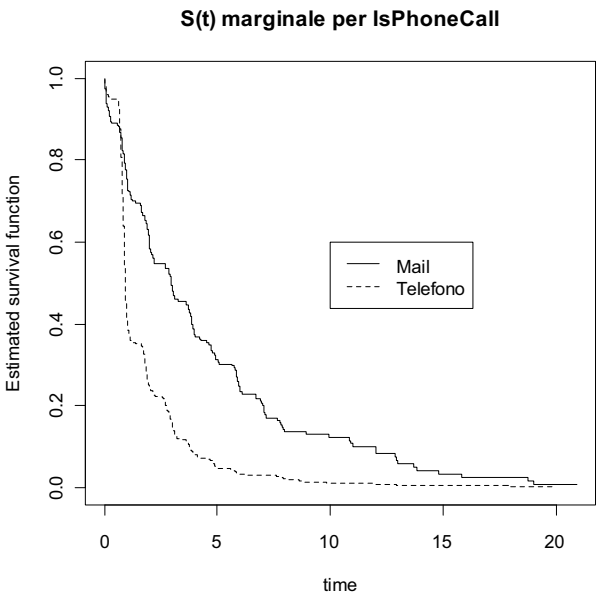
Confrontiamo i tempi di durata per i ticket presi in esame secondo la categorie del cliente e vediamo come si comporta la funzione di sopravvivenza stratificata.

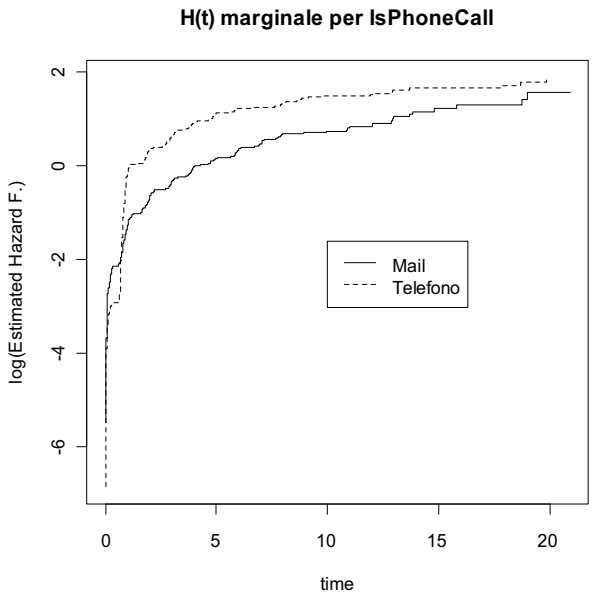


Le tre funzioni di sopravvivenza stimate (grafico a sinistra) sembrano essere diverse tra di loro e si intersecano in corrispondenza del tempo 1 e del tempo 4. Dal grafico si può vedere che i ticket dell'azienda A hanno una probabilità di sopravvivenza minore rispetto alle altre due categorie. La sopravvivenza negli ultimi dati rimane costante e questo è dovuto alla presenza di alcuni dati censurati. Possiamo effettuare il test Log-Rank per verificare l'ipotesi nulla che le tre funzioni di sopravvivenza siano uguali tra loro, ma prima guardiamo se i rischi sono proporzionali per avere massima potenza nel test. I rischi (grafico a destra) non sono proporzionali: le curve di B e C si intersecano tra di loro. La "non proporzionalità" dei rischi è comunque limitata, dunque, applicando il test Log-Rank, otteniamo che per qualsiasi valore di alfa prefissato rifiutiamo l'ipotesi di uguaglianza delle funzioni di sopravvivenza marginali.

-IsPhoneCall

Confrontiamo marginalmente i tempi di vita per i ticket presi discriminando per IsPhoneCall. Vediamo come si comporta la funzione si sopravvivenza stratificata per la variabile presa in esame.





Le due funzioni di sopravvivenza stimate sono diverse tra loro. Siamo portati a supporre che la sopravvivenza dei ticket aperti via telefono, è minore rispetto ai ticket aperti via e-mail. Anche qui applichiamo il test di Log-Rank per verificare se i ticket aperti via e-mail hanno la stessa funzione di sopravvivenza dei ticket aperti via telefono. Possiamo assumere la proporzionalità dei rischi. Applicando il test di Log-Rank otteniamo che per qualsiasi valore di alfa prefissato rifiutiamo l'ipotesi Ho.

Adattamento dei modelli parametrici

Nell'analisi indicheremo con:

z_{i1}= valore della variabile *IsPhoneCall* dell'i-esimo ticket;
z_{i2}= valore della variabile *trs* dell'i-esimo ticket;
z_{i3}= valore della varibile Cat.cliente dell'i-esimo ticket che se è uguale a uno corrisponde al cliente B, zero altrimenti;
z_{i4}= valore della variabile Cat.cliente dell'i-esimo ticket che se è uguale a uno corrisponde al cliente C, zero altrimenti;
z_{i5}= valore della variabile *gmv* dell'i-esimo ticket che se è uguale a uno se il giorno di apertura corrisponde a mercoledì o venerdì altrimenti se vale zero corrisponde ai restanti giorni della settimana;
z_{i6}= valore della variabile Parte.Settimana dell'i-esimo ticket che se è uguale a uno il giorno di apertura corrisponde a venerdì altrimenti se vale zero corrisponde ai restanti giorni della settimana ;

Cominciamo ad adottare ai nostri dati alcuni modelli parametrici.

Gli output che mostrerò faranno riferimento ai modelli finali che ho trovato, riferirsi all'Appendice per maggiori dettagli.

Distribuzione di Weibull

Il modello ridotto fornisce il seguente output:

```
                          Value Std. Error       z        p
(Intercept)               1.768     0.1516  11.662 1.99e-31
IsPhoneCall1             -1.531     0.1494 -10.253 1.14e-24
trsi                     -0.167     0.0730  -2.293 2.18e-02
Cat.clienteB             -1.579     0.6391  -2.471 1.35e-02
Cat.clienteC             -0.450     0.1698  -2.648 8.10e-03
gmv2                      0.182     0.0566   3.207 1.34e-03
IsPhoneCall1:Cat.clienteB 2.410     0.6411   3.759 1.71e-04
IsPhoneCall1:Cat.clienteC 0.838     0.1888   4.438 9.09e-06
trsi:Cat.clienteB         0.282     0.2957   0.952 3.41e-01
trsi:Cat.clienteC         0.326     0.1346   2.419 1.56e-02
Log(scale)               -0.130     0.0214  -6.088 1.14e-09
Scale= 0.878
```

Nel modello ridotto restano le variabili Cat.cliente, IsPhoneCall, trasferta, l'interazione tra loro. Inoltre viene creata la nuova variabile gmv che può assumere due valori:

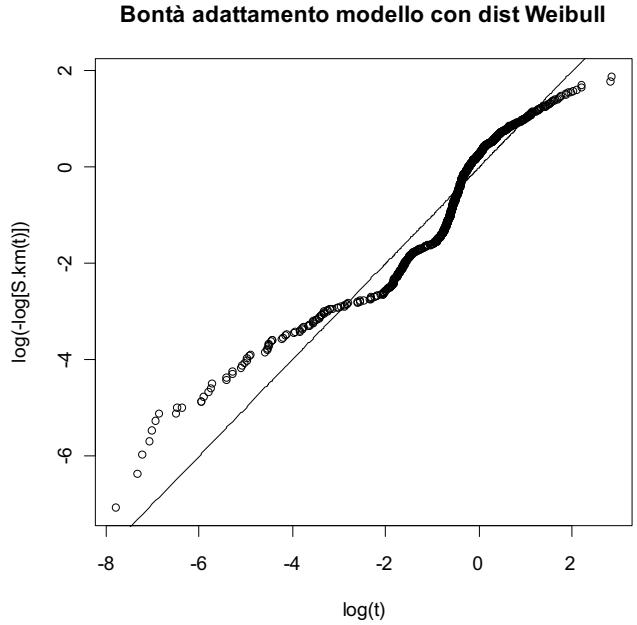
- "1" se il giorno di apertura è lun,mar,gio,
- "2" se il giorno di apertura è mer,ven.

Ottengo cioè

$$T|\boldsymbol{z} = \frac{T_0}{exp(\boldsymbol{z};\boldsymbol{\beta})}$$

Dove:
$T_0 \sim Weibull(exp(-(\text{intercept}), 1/scale\ )$
$exp(\boldsymbol{z};\boldsymbol{\beta}) =$ exp $(1.768 - 1.531 * z_{i1} - 0.167 * z_{i2} - 1.579 * z_{i3} - 0.450 * z_{i4} + 0.182 * z_{i5}$
$+ 2.140 * z_{i1} * z_{i3} + 0.838 * z_{i1} * z_{i4} + 0.282 * z_{i2} * z_{i3} + 0.326 * z_{i2} * z_{i4})$

Verifica grafica dell'adattamento :



Nel grafico vedo che $\log\left(-\log\left(\hat{S}_{KM}(t)\right)\right)$, non è una funzione lineare per il $log(t)$.
La distribuzione Weibull non sembra adatta ai tempi di sopravvivenza del nostro data set.

Distribuzione Log-Logistica

Adottando sempre la strategia backward per arrivare ad un modello ridotto, assumendo che T si distribuisca come una v.c. log-logistica, ottengo il seguente modello:

	Value	Std. Error	z	p
(Intercept)	1.682	0.1270	13.253	4.36e-40
IsPhoneCall1	-1.713	0.1270	-13.486	1.89e-41
trsi	-0.040	0.0630	-0.635	5.25e-01
Cat.clienteB	-1.503	0.5279	-2.848	4.40e-03
Cat.clienteC	-1.134	0.1580	-7.173	7.35e-13
IsPhoneCall1:Cat.clienteB	2.310	0.5308	4.353	1.34e-05
IsPhoneCall1:Cat.clienteC	1.031	0.1806	5.705	1.16e-08
trsi:Cat.clienteB	0.111	0.3168	0.350	7.26e-01
trsi:Cat.clienteC	0.477	0.1378	3.461	5.38e-04
Log(scale)	-0.677	0.0264	-25.664	2.98e-145

Scale= 0.508

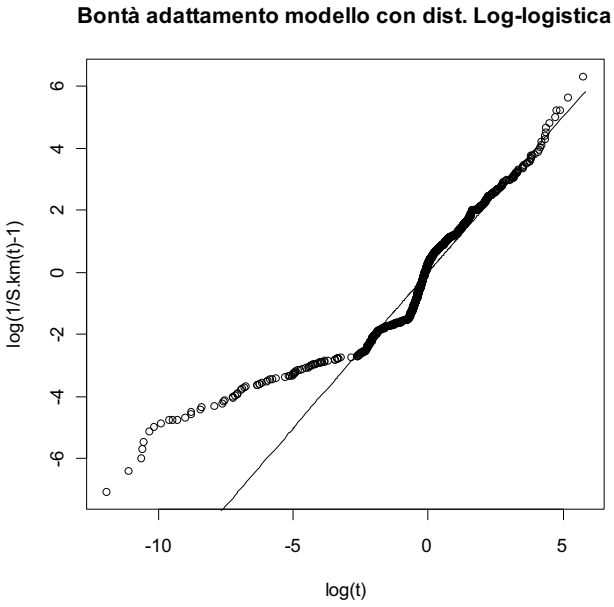
Ottengo quindi

$$T|\boldsymbol{z} = \frac{T_0}{\exp(\boldsymbol{z};\boldsymbol{\beta})}$$

Dove:
$T_0 \sim LogLogistica\ (\exp(-intercept), 1/scale)$

$$\exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp(1.682 - 1.713 * z_{i1} - 0.040 * z_{i2} - 1.503 * z_{i3} - 1.134 * z_{i4} + 2.310 * z_{i1} * z_{i3} + 1.031 * z_{i1} * z_{i4} + 0.111 * z_{i2} * z_{i3} + 0.477 * z_{i2} * z_{i4})$$

Verifica grafica dell'adattamento :



Anche il modello a tempi accelerati con distribuzione log-logistica non sembra essere adatta per il tempo di sopravvivenza dei ticket.

Distribuzione Log-Normale

Si costruisce il modello con l'assunto distributivo Log-Normale in modo analogo a quanto fatto per il modello precedente.

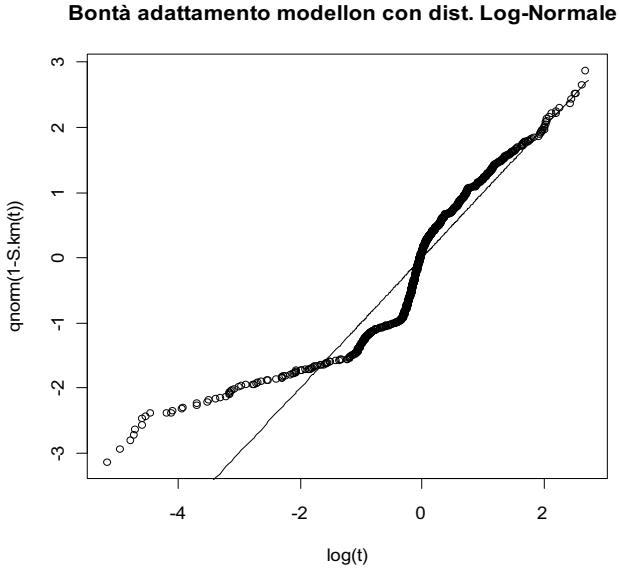
	Value	Std. Error	z	p
(Intercept)	1.7014	0.1790	9.505	2.00e-21
IsPhoneCall1	-1.7445	0.1798	-9.704	2.89e-22
trsi	-0.0537	0.0889	-0.604	5.46e-01
Cat.clienteB	-1.5222	0.7921	-1.922	5.47e-02
Cat.clienteC	-1.4037	0.2031	-6.913	4.74e-12
IsPhoneCall1:Cat.clienteB	2.3419	0.7954	2.944	3.24e-03
IsPhoneCall1:Cat.clienteC	0.8724	0.2241	3.893	9.91e-05
trsi:Cat.clienteB	-0.1752	0.3631	-0.482	6.30e-01
trsi:Cat.clienteC	0.7208	0.1584	4.550	5.37e-06
Log(scale)	0.0873	0.0214	4.082	4.46e-05

Scale= 1.09

Dove:

$T_0 \sim LogN(\exp(-intercept), 1/scale)$

$$\exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp(1.7014 - 1.7445 * z_{i1} - 0.0537 * z_{i2} - 1.5222 * z_{i3} - 1.4037 * z_{i4} + 2.3419 * z_{i1} * z_{i3} + 0.8724 * z_{i1} * z_{i4} - 0.1752 * z_{i2} * z_{i3} + 0.7208 * z_{i2} * z_{i4})$$



Nemmeno il modello a tempi accelerati con distribuzione log-Normale riesce ad adattarsi bene.

I modelli parametrici adottati , non riescono a spiegare bene il fenomeno e questo possiamo vederlo dalle funzioni non lineari sul tempo delle $\hat{S}_{KM}(t)$ stimate sui residui standardizzati dei vari modelli. Guardando i tre grafici dei residui, si può affermare che il modello con distribuzione di Weibull è quello che si adatta meglio.
Proviamo a vedere se adottando un modello semi parametrico di Cox la situazione migliora.

Adattamento del modello di Cox

Vediamo come si adatta il modello a rischi proporzionali di Cox. Il modello ridotto che otteniamo è il seguente:

```
                              coef exp(coef) se(coef)      z Pr(>|z|)
IsPhoneCall1                1.69704   5.45775  0.17439  9.731  < 2e-16 ***
trsi                        0.15734   1.17040  0.08278  1.901  0.05734 .
Cat.clienteB                1.45570   4.28747  0.72921  1.996  0.04590 *
Cat.clienteC                0.59094   1.80568  0.19414  3.044  0.00233 **
IsPhoneCall1:Cat.clienteB  -2.34502   0.09585  0.73372 -3.196  0.00139 **
IsPhoneCall1:Cat.clienteC  -1.04695   0.35101  0.21765 -4.810 1.51e-06 ***
trsi:Cat.clienteB          -0.32458   0.72283  0.33393 -0.972  0.33104
trsi:Cat.clienteC          -0.33333   0.71653  0.15269 -2.183  0.02903 *
```

La funzione di rischio quindi è:

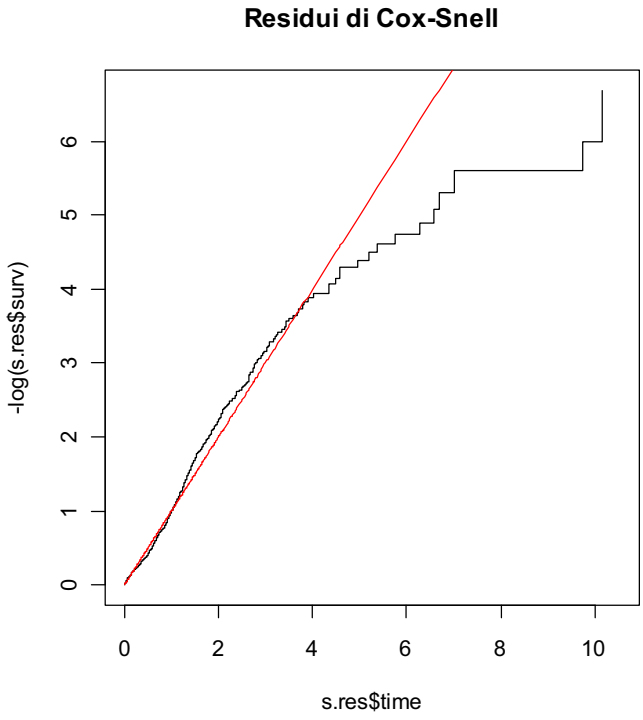
$$h(t|\mathbf{z}) = h_0(t)exp(\mathbf{z};\boldsymbol{\beta})$$

dove

$$\exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp\,(1.69704 * z_{i1} + 0.15734 * z_{i2} + 1.45570 * z_{i3} + 0.59094 * z_{i4} - 2.34502 * z_{i1} * z_{i3} - 1.04695 * z_{i1} * z_{i4} - 0.32458 * z_{i2} * z_{i3} - 0.3333 * z_{i2} * z_{i4})$$

Ora che abbiamo stimato adattato il modello semi-parametrico ai dati non ci resta che verificarne la bontà dell'adattamento.

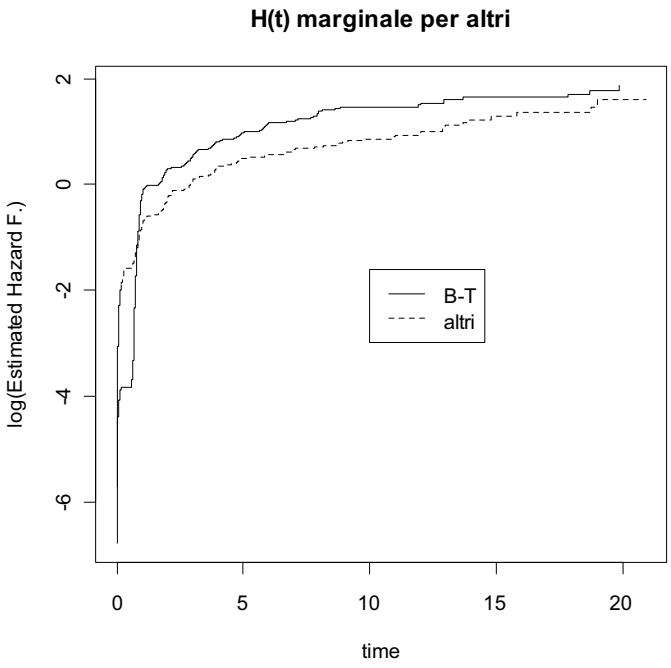
Cominciamo dai residui di Cox – Snell che dovrebbero comportarsi come un campione censurato da una v.c. $Esp(1)$ se il modello è ben adattato.
Riportiamo nel grafico sottostante la funzione di rischio cumulato dei residui e la confrontiamo con la retta bisettrice.



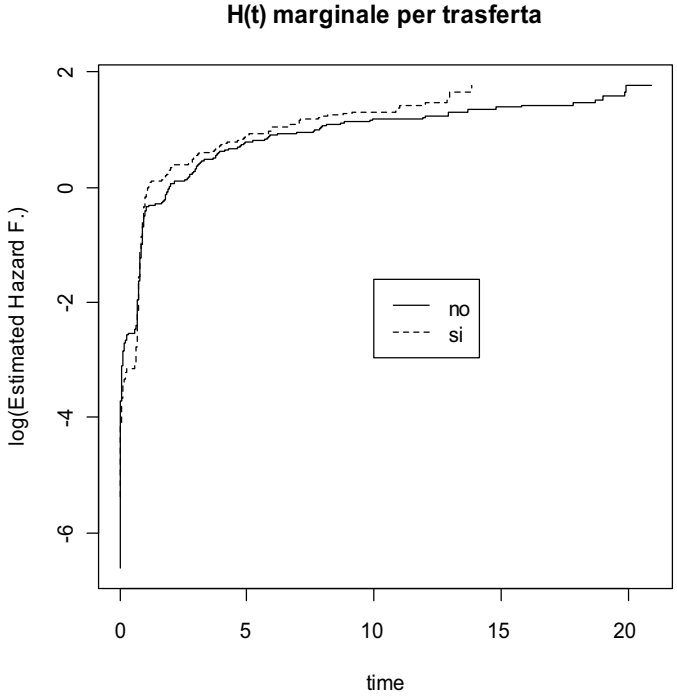
$H(t)$ non oscilla molto attorno alla bisettrice in particolare dal tempo 4 in poi si discosta di molto dalla retta. Il modello di Cox non sembra adattarsi bene, ma lo si potrebbe migliorare sfruttando il grafico dei residui di devianza e beta eliminando le osservazioni influenti per vedere se l'adattamento migliora.
Inoltre come abbiamo potuto osservare prima dalle analisi marginali l'assunto di proporzionalità non è rispettato per la variabile Cat.cliente.

L'output R , riportato sopra,rivela che per il modello di Cox non sembra esserci una differenza molto significativa tra i due clienti A e B. Potremo costruire quindi una nuova variabile cliente chiamandola "altri" che ha due modalità:

1. 0 se Cat.cliente=A o Cat.cliente=B
2. 1 se Cat.cliente=C



Nel grafico delle funzioni di rischio cumulato per la variabile altri, osservo che nei due livelli la proporzionalità delle H(t) può essere accettata. Nella prima parte le due funzioni si incrociano però questo è dovuto alla censura dei dati.



Dal grafico a fianco si può evincere che anche per la trasferta può essere assunta la proporzionalità dei rischi.

Indicheremo con z_{i7} nel nuovo modello di Cox il valore della variabile "*Altri"* dell'i-esimo ticket

Verificati gli assunti di proporzionalità non ci resta di applicare il nuovo modello al nostro data-set: l'output R del modello ridotto è il seguente:

```
                          coef exp(coef) se(coef)      z Pr(>|z|)
IsPhoneCall1          1.30586   3.69085  0.16786  7.779 7.33e-15 ***
trsi                  0.51846   1.67944  0.07349  7.055 1.73e-12 ***
altri1                0.73327   2.08188  0.19100  3.839 0.000124 ***
IsPhoneCall1:altri1 -0.67140   0.51099  0.21251 -3.159 0.001581 **
trsi:altri1         -0.68893   0.50211  0.14778 -4.662 3.13e-06 ***
```

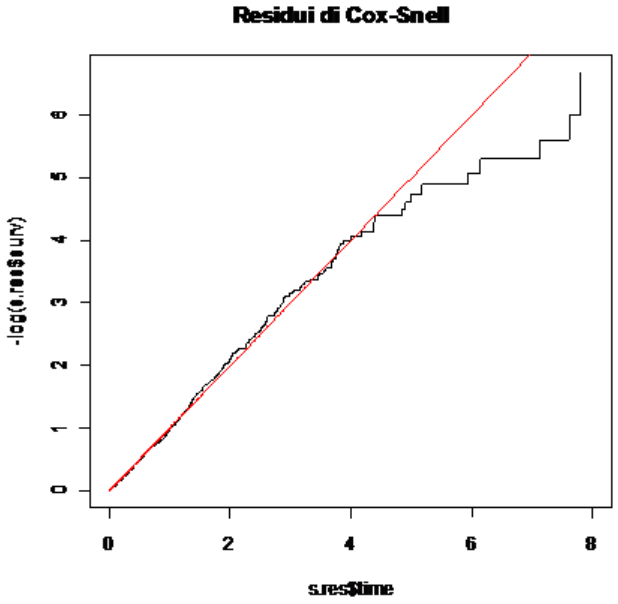
La funzione di rischio stimata quindi è:

$$h(t|\mathbf{z}) = h_0(t)exp(\mathbf{z};\boldsymbol{\beta})$$

dove

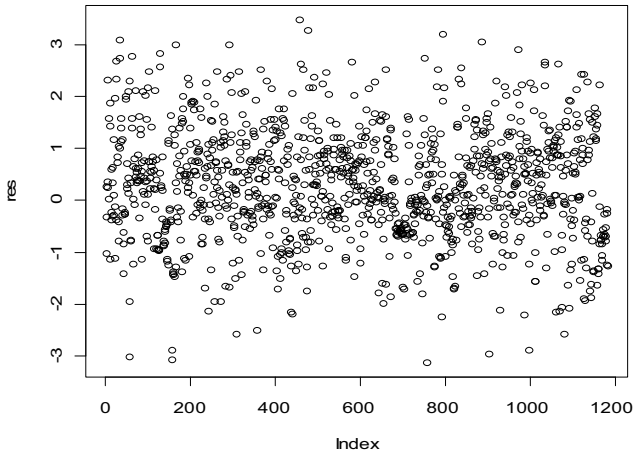
$$\exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp\left(1.306 * z_{i1} + 0.518 * z_{i2} + 0.733 * z_{i7} - 0.671 * z_{i1} * z_{i7} - 0.689 * z_{i2} * z_{i7}\right)$$

Guardiamo come si adatta il modello stimato guardando i residui i Cox-Snell.

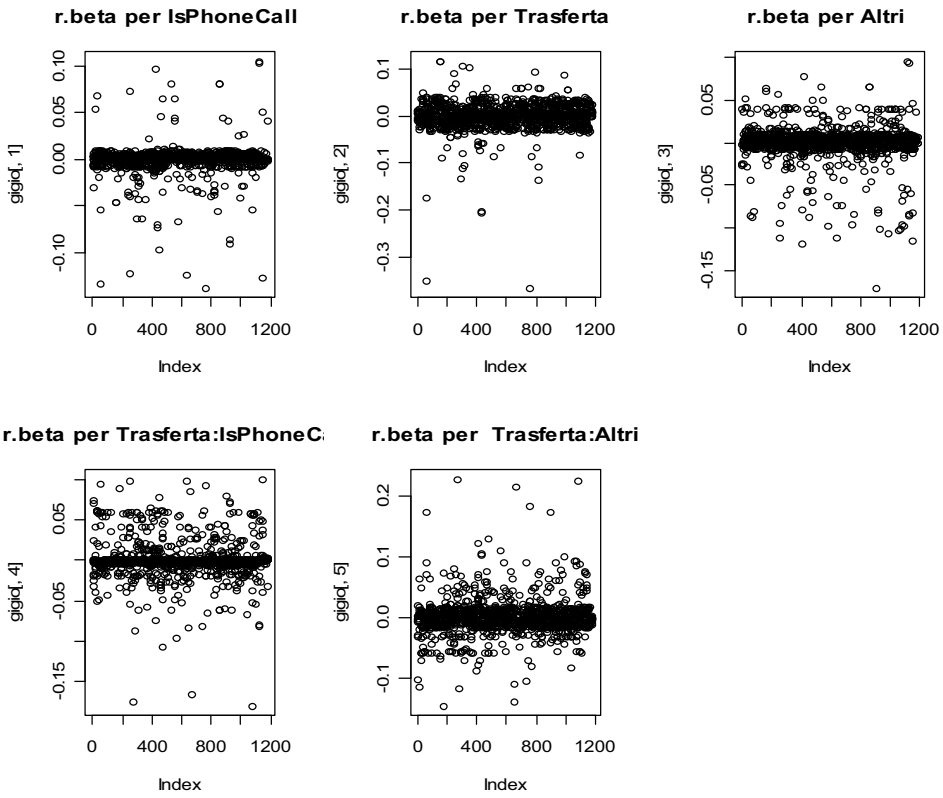


Il modello adottato sembra andare meglio di prima, la funzione rischio cumulato si adatta bene alla retta, esclusa l'ultima parte che si discosta non di molto. Ora che abbiamo verificato la bontà del modello di Cox e l'assunto di rischi proporzionali andiamo a controllare i residui di devianza ed i residui beta.

Residui di devianza



Residui Beta



I residui di devianza non presentano comportamenti sistematici, sono disposti a nuvola di punti,si distribuiscono nell' intervallo da (-3,3) e non sembrano esserci osservazioni anomale.

Nei residui beta sono presenti diversi punti influenti che si ripetono in tutti i cinque grafici e provando a toglierli le stime dei coefficienti cambiano radicalmente e molte variabili non risultando più significative per il modello.

Risultati dei modelli adottati

Per i dati di sopravvivenza riguardanti i ticket ho visto che il modello che si adatta meglio è il modello a rischi proporzionali di Cox con la variabile " Altri" in sostituzione alla variabile Cat.cliente.
Si poteva ipotizzare una funzione di rischio crescente nel tempo e si poteva provare con un modello parametrico a tempi accelerati con distribuzione di Weibull, ma si è visto che non andava bene.
Il modello con distribuzione Log-Normale e Log-Logistica erano anche essi inadeguati e questo possiamo vederlo dalle funzioni non lineari sul tempo delle $\hat{S}_{KM}(t)$ stimate sui residui standardizzati dei vari modelli. Non viene preso in considerazione il modello con distribuzione esponenziale in quanto sembra irragionevole pensare la funzione di rischio dei ticket costante nel tempo.
Dalle analisi marginali avevo visto che la funzione di sopravvivenza era diversa a seconda del trasferta, Categoria del cliente e anche a seconda della tipologia di apertura (via e-mail o via telefono).
Dall'output del modello di Cox trovato in precedenza, si nota che i ticket aperti telefonicamente vengono risolti prima anche se non c'è una trasferta.
Se il ticket viene aperto da un altro cliente (categoria C), il tempo impiegato per risolverlo è minore rispetto a quello impiegato per i clienti A o B.
Però, se questo altro cliente apre il ticket telefonicamente il tempo per chiudere il ticket cresce rispetto ad una richiesta per mail così come il tempo cresce se viene fatta una trasferta per il cliente in questione rispetto a quando non viene fatta.

Nel prossimo capitolo proverò ad adottare un metodo non-parametrico ai dati utilizzando le Random Survival Forest, confrontando infine i risultati ottenuti dalle soluzioni precedenti.

CAPITOLO 3
Random Survival Forest

Le Random Survival Forest (RSF) sono state introdotte nell'articolo da Ishwaran and Kogalur (2007) e sono un' implementazione delle Random Forest (Breiman, 2001) per l'analisi dei dati di sopravvivenza. Prima di introdurre l'argomento richiamo un concetto di metodo Bootstrap e Random Forest.

Metodo Bootstrap

Il bootstrap (Zieffler A. Harring R. (2011) Randomization and Bootstrap Methods Using R) è una tecnica statistica di ricampionamento per approssimare la distribuzione campionaria di una statistica. Permette cioè, di approssimare media e varianza di uno stimatore, costruire intervalli di confidenza e calcolare p-value di test statistici in particolare quando non si conosce la distribuzione di uno stimatore.

	Mondo Reale	**Mondo Parallelo**
fenomeno d'interesse :	Y ~ P	Y^* ~ $\hat{P}$
dati :	y = (determinazione di Y)	y^*= (determinazione di Y^*)
Statistica d'interesse :	$T(y)$	$T(y^*)$
Problemi che ci poniamo :	Distribuzione di $T(Y)$? Media e varianza di $T(Y)$?	Se $\hat{P}$ è completamente nota la distribuzione $T(Y^*)$ in questo mondo "parallelo" è nota.

L'idea del bootstrap è di affiancare al "mondo reale" un "mondo parallelo". Nel mondo "parallelo" esiste una variabile casuale Y^* la cui distribuzione $\hat{P}$ è una stima di P costruita sulla base dei dati del "mondo reale": se facessimo un esperimento nel "mondo parallelo", osserveremmo dei dati y^* e la statistica di interesse avrebbe valore $T(y^*)$. Poiché la distribuzione dei dati nel mondo parallelo è nota lo è anche la distribuzione della statistica d'interesse: riusciamo a rispondere nel mondo parallelo ai problemi che ci eravamo posti nel mondo reale. L'idea di bootstrap è di usare come approssimazione della distribuzione di $T(Y)$ nel "mondo reale" la vera distribuzione di $T(Y^*)$ nel "mondo parallelo". Si osservi che la distribuzione di $T(Y^*)$, cioè la distribuzione della statistica d'interesse di $T(.)$, quando i dati sono generati da Y^*, è spesso difficile da calcolare analiticamente.

Si procede quindi via simulazione con il metodo Monte Carlo: nel mondo parallelo simuliamo B volte (B molto grande) la ripetizione dell'esperimento , cioè B replicazioni dei dati originali

$$y_1^* \sim \widehat{P}, \dots\dots\dots\dots\dots, y_B^* \sim \widehat{P}$$

Successivamente calcoliamo la statistica d'interesse per ciascuna delle replicazioni dei dati ottenute al passo precedente

$$t_1^* = T(y_1^*), \dots\dots\dots\dots\dots, t_B^* = T(y_B^*)$$

e utilizziamo questi valori per ottenere delle informazioni sulla distribuzione di $T(Y^*)$.

Random Forest

Le foreste casuali sono una tecnica di classificazione (Breiman , Cutler, 2001) utilizzate per migliorare le prestazioni di un processo di analisi. Si fa riferimento alle RF indicando le combinazioni di alberi ottenute utilizzando la selezione casuale delle variabili esplicative. La procedura consiste nel selezionare in modo casuale ,ad ogni nodo di un albero, un sottoinsieme di variabili esplicative di grandezza K costante in tutti i nodi che verranno poi analizzate per trovare lo "splitting" ottimale. L'albero viene fatto crescere fino alla grandezza massima e non viene potato come un normale albero classificatore;sarà infatti l'operazione di combinazione dei diversi alberi che permetterà di evitare problemi di sovra adattamento. Questa procedura viene applicata per tutti i B alberi della foresta. Ciascun albero viene fatto crescere su un campione bootstrap diverso utilizzando per ciascun nodo un numero K di variabili selezionate casualmente. I vantaggi delle Random Forest sono:

- Producono un albero classificatore molto accurato con un insieme di dati molto grande, riuscendo a dare una stima di importanza alle variabili esplicative.
- Permettono di gestire un gran numero di variabili esplicative e trovare le interazioni tra loro.
- Comprendono un buon metodo per la stima dei dati mancati.
- Sono Utili per l'individuazione di *out-liers* e per la visualizzazione dei dati

Per maggiori approfondimenti si rimanda Azzalini A. e Scarpa B. (2004) Analisi dei dati e Data mining, Milano, Springer – Verlag Italia.

RSF (Random Survival Forest)

Sono state create successivamente alle RF per effettuare delle analisi statistiche nell'ambito dell'analisi di sopravvivenza. Le RSF sono un metodo estremamente adattivo ai dati, non vincolato a particolari assunzioni restrittive: questo è un enorme vantaggio nell'analisi dei dati di durata, perché i metodi spesso usati sono legati sempre ad ipotesi, come ad esempio la proporzionalità dei rischi nel modello di Cox. Essendo strettamente collegate alle RF ne hanno ereditato tutti i vantaggi.

Le caratteristiche principali da evidenziare delle RSF sono due:

- Facili da usare, sono dei metodi robusti,e prima devono essere impostati i seguenti parametri:
    1) Numero di variabili esplicative da selezionare casualmente per ogni nodo dell'albero
    2) Numero di alberi che compongono la foresta
    3) Scegliere le Splitting Rules per la creazione dei vari nodi.
- Metodo che si adatta in maniera semplice ai dati e virtualmente non vincolato a particolari assunzioni come detto in precedenza.

ALGORITMO DELLE RSF

L'algoritmo delle RandomSurvivalForest proposto da Ishwarab e Kogalur è descritti in questi cinque passi:

1) Si creano n campioni bootstrap dai dati originali.
2) Ogni albero bootstrap viene fatto crescere nel seguente modo: vengono scelte casualmente per ogni nodo M variabili esplicative a cui verrà applicato un opportuno criterio di Splitting. La migliore suddivisione di un nodo avviene quando la differenza tra le $S(t)$ da nodo padre a nodo figlia è massima.
3) La crescita dell'albero avviene con la massima ampiezza con il vincolo che deve essere almeno una "morte" per nodo.
4) Viene calcolato il rischio cumulato $H(t)$ combinando l'informazioni proveniente dagli n alberi. Viene cosi calcolata una stima per ogni individuo presente nei dati originali.
5) Per calcolare il tasso di errore (OOB) vengono utilizzate quelle osservazioni che non vengono prese in considerazione per costruire l'albero.

Solitamente si usa il metodo di splitting che produce un minore tasso di errore. Infatti nella libreria sviluppata in R sono proposti quattro differenti metodi di splitting: Log-Rank splitting, *Conservation of events* splitting, Log-rank score splitting e *Random splitting*.

Log-Rank splitting

Prima di introdurre il metodo di *splitting* bisogna inserire alcune notazioni. Siano $t_1 < t_2 <\ldots\ldots< t_n$ i tempi di morte distinti per un nodo padre h e sia $d_{i,j}$ e $Y_{i,j}$ rispettivamente il numero di morti e di individui a rischio al tempo di morte t_i nei nodi figli $j = 1,2$. Definiamo quindi $Y_{i,1} = \#\{ T_l \geq t_i, x_l \leq c\}$, $Y_{i,2} = \#\{ T_l \geq t_i, x_l > c\}$ dove x_l è il valore di un predittore x per un individuo l con $l = 1$, …,n e $Y_i = Y_{i,1} + Y_{i,2}$ e $d_i = d_{i,1} + d_{i,2}$. Definiamo infine con n_j il numero totale di osservazioni in un nodo figlio *j* e quindi n = $n_1 + n_2$. Notiamo che n_1=#$\{l: x_l \leq c\}$ e n_2=#$\{l: x_l > c\}$.

Il log-rank test per un suddivisione a un valore c per un predittore x è il seguente :

$$L(x,c) = \frac{\sum_{i=1}^{N} (d_{i,1} - Y_{i,1}\frac{d_i}{Y_i})}{\sqrt{[\sum_{i=1}^{N} \frac{Y_{i,1}}{Y_i}\left(1 - \frac{Y_{i,1}}{Y_i}\right)\left(\frac{Y_i - d_i}{Y_i - 1}\right) d_i]}}$$

(dove $L(x,c)$ è la radice quadrata del test log-rank introdotto nel capitolo 2). Il valore $|L(x,c)|$ è la misura della separazione tra i nodi e più grande è questo valore, maggiore sarà la differenza tra due gruppi e maggiore sarà la loro divisione. Il migliore *splitting* si ha quando per un nodo *h* si trovano x^* e c^* tale che per ogni *x* e c si ottiene $|L(x,c)| \leq |L(x^*,c^*)|$.

Insieme di stima

Le RSF producono una stima per la funzione di rischio cumulato $H(t)$ dei dati, che poi verrà successivamente usata per calcolare il tasso di errore. Per ogni albero cresciuto tramite *bootstrap* viene stimato il rischio cumulato e questo viene realizzato raggruppando le stime del rischio al nodo terminale. Consideriamo uno specifico nodo h e definiamo $\{t_{l,h}\}$ l'insieme dei tempi distinti di morte per il nodo, mentre $\{d_{l,h}\}$ e $\{Y_{l,h}\}$ sono rispettivamente l'insieme delle morti e degli individui a rischio nel insieme $\{t_{l,h}\}$.
La funzione del rischio cumulato per il nodo h è definita come:

$$\widehat{H_h(t)} = \sum_{t_{l,h} \leq t} \frac{d_{l,h}}{Y_{l,h}} \qquad (2)$$

Ogni albero verrà fornita una sequenza di stime $\widehat{H_h(t)}$: se ci sono *m* nodi terminali in un albero, ci saranno quindi *M* stime.
Per calcolare $\widehat{H(t|x_i)}$ per un individuo *i* con predittore x_i, basta scendere x_i dall'albero. I nodi foglia forniscono lo stimatore desiderato per l'individuo i :
$\widehat{H(t|x_i)} = \widehat{H_h(t)}$ se x_i appartiene al nodo *h*.
Questo valore verrà calcolato per tutti gli individui presenti nel nostro insieme di dati.

La stima (2) è basata solo su un albero e per produrre la stima complessiva dobbiamo calcolarla per tutti gli alberi: sia $\widehat{H_b(t|x)}$ la funzione di rischio cumulato per gli alberi b=1,.....,ntree.
Definiamo una variabile indicatrice $I_{i,b}$ che assume due valori:

$$\begin{cases} 1\ se\ \ i\ \in\ b\ . \\ 0\ altrimenti. \end{cases}$$

Lo stimatore della funzione di rischio cumulato per *i* è

$$\widehat{H_e^*}(t|x_i) = \frac{\sum_{b=1}^{ntree} I_{i,b} \widehat{H_b(t|x_\iota)}}{\sum_{b=1}^{ntree} I_{i,b}}$$

Tasso di Errore

Lo stimatore $\widehat{H_e^*}(t|x_i)$ è l'elemento per calcolare il tasso di errore che viene misurato tramite l'indice di concordanza di Harrell (*Harrell 1982*). A differenza di altre misure per la bontà della sopravvivenza, questo indice non dipende dalla scelta di un tempo fisso per la valutazione di un modello e più precisamente tiene conto delle censure degli individui ed ha lo scopo di valutare la bontà delle previsioni nell'analisi dei dati di durata.

Per calcolare l'indice devo definire quale è il peggiore risultato previsionale: sia t_1^* ..... t_N^* i tempi distinti di morte dei nostri dati. L'individuo *i* ha il peggiore risultato rispetto all'individuo *j* se

$$\sum_{K=1}^{N} \widehat{H_e^*}(t_k^*|x_i) > \sum_{K=1}^{N} \widehat{H_e^*}\left(t_k^*\middle|x_j\right)$$

Il tasso di errore è calcolato nel seguente modo:

1. Si forniscono tutte le possibili coppie del data set.
2. Si tolgono le coppie dove il tempo più piccolo è censurato. Inoltre, omettere le coppie i e j se $T_i = T_j$, ma se $\vartheta_i = 1$ e $\vartheta_j = 0$ oppure se $\vartheta_i = 0$ e $\vartheta_j = 1$ non vanno tolte. Definiamo come *Permissible* il numero totale di coppie ammissibili.
3. Conta 1 per ogni coppia in cui un tempo più breve ha il peggiore risultato. Conta 0.5 se i risultati di previsione sono accoppiati. Definiamo *Concordance* la somma totale di tutte le coppie.
4. Definiamo l'indice di concordanza C come

$$C = \frac{Concordance}{Permissible}$$

5. Il tasso di errore è definito come $Errore = 1 - C$. Notiamo che l'errore è compreso $0 \leq Errore \leq 1$ e se *Errore*=0.5 è come lanciare a caso una moneta, mentre Errore=0 indica un accuratezza perfetta.

Adattamento delle RSF

Lo scopo di questa parte dell'elaborato è quella di ricondurre un'analisi dei dati sulla durata dei ticket di Wintech utilizzando un approccio non parametrico basato sulle Random Survival Forest usando la libreria in R.

Una volta scelto lo splitting *Log-rank* Il modello finale ottenuto è sintetizzato nel seguente output R:

```
                         Importance   Relative Imp
Cat.cliente:IsPhoneCall      0.0175         1.0000
Mese                         0.0151         0.8619
IsPhoneCall                  0.0114         0.6494
Cat.cliente                  0.0040         0.2253
tr                           0.0038         0.2148
Giorno                       0.0027         0.1526


Sample size: 1182                    Average no. of terminal nodes: 84.03
Number of deaths: 1099               Total no. of variables: 6
Number of trees: 1000                Minimum terminal node size: 3
Splitting rule: logrank              Estimate of error rate: 36.35%
```

Grafico A

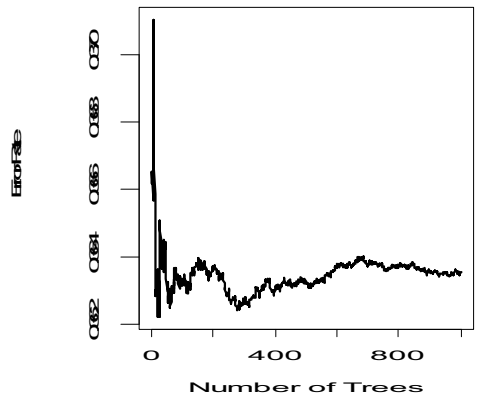
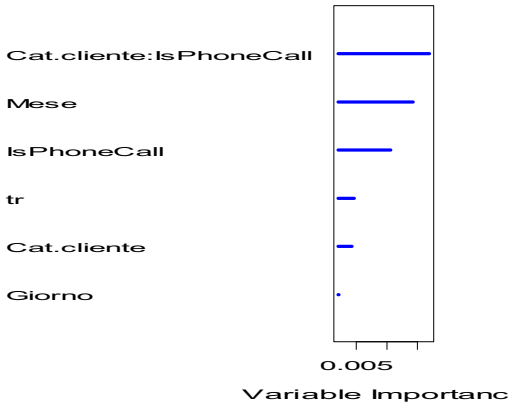


Grafico B



Il grafico A rappresenta la frazione di *error rate* per il modello RSF come funzione del numero di alberi, mentre il grafico B rappresenta *out-of-bag* valore di importanza dei predittori. Secondo

l'output le variabili che hanno maggiore importanza nello spiegare la durata dei ticket sono l'interazione Cat.cliente e IsPhoneCall, Mese, IsPhoneCall.

Confronto tra i modelli

Risulta difficile dopo queste analisi confrontare le RSF con i modelli trovati in precedenza essendo di diversa natura e avendo delle variabili di costruzione diverse. Non ci resta che scegliere il modello che produce il minor errore di previsione. Per affrontare questo problema mi sono appoggiato ad un package R pec (Mongensen, Ishwaran, Gerds) .

L'errore di previsione al tempo t_i è definito come *the Brier Score*

$$BS\big(t,\hat{S}\big) = E(Y_i\,(t) - \widehat{S}\,(t|X_i))^2$$

dove i è un soggetto che non fa parte del *traning data ,* $\mathrm{Y_i\,(t) = P(T_i > }t)$ è lo stato reale del soggetto *i* e $\widehat{S}\,(t|X_i)$ è la stima della probabilità di sopravvivenza al tempo t per il soggetto i con variabili predittrici definite come X_i. Un valore utile per valutare *The Brier score* sono: 33% come un numero casuale generato da $U\sim(0{,}1)$, 25% come una buona stima e 0% stima perfetta.

La funzione *pec* stima e confronta gli errori di previsioni di diversi modelli di analisi di sopravvivenza che possono essere di diversa natura. Nella libreria ci sono diversi metodi per affrontare il problema dell' *over-fitting* tra cui la cross-validation. Per ulteriori approfondimenti/riferimenti si segnala l'articolo scritto da Gerds, Mogensen, Ishwaran Evaluating random forest for survival analysis using prediction error curves (2000) .

Nell'analisi successiva si confrontano tre modelli di analisi di sopravvivenza:

1. Modello parametrico di Weibull.
2. Modello semiparetrico di Cox.
3. Modello non parametrico Random Survival Forest.

Cominciamo ad adottare ai nostri dati i tre metodi di analisi di sopravvivenza usando un insieme stima per la costruzione del modello e un insieme di verifica per verificare la bontà delle previsioni. L'insieme di stima e di verifica sono stati creati estraendo casualmente dal campione originale un sottoinsieme di dati di grandezza rispettivamente 800 e 382.

Nell'analisi indicheremo con:

z_{i1}= valore della variabile *IsPhoneCall* dell'i-esimo ticket;
z_{i2}= valore della variabile *trs* dell'i-esimo ticket;
z_{i3}= valore della varibile Cat.cliente dell'i-esimo ticket che se è uguale a uno corrisponde al cliente B, zero altrimenti;
z_{i4}= valore della variabile Cat.cliente dell'i-esimo ticket che se è uguale a uno corrisponde al cliente C, zero altrimenti;
z_{i5}= valore della variabile *gmv* dell'i-esimo ticket che se è uguale a uno se il giorno di apertura corrisponde a mercoledì o venerdì altrimenti se vale zero corrisponde ai restanti giorni della settimana;
z_{i6}= valore della variabile mameve dell'i-esimo ticket che se è uguale a uno il giorno di apertura corrisponde a martedì,mercoledì,venerdì altrimenti se vale zero corrisponde ai restanti giorni della settimana
z_{i7}= valore della variabile altri dell'i-esimo ticket che se è uguale a uno il cliente corrisponde alla categoria A o B altrimenti se vale zero corrisponde alla categoria C

Modello parametrico di Weibull

Il modello ridotto ottenuto attraverso l'insieme di stima fornisce il seguente output R:

```
(Intercept)           1.868     0.1926  9.70 2.98e-22

IsPhoneCall1         -1.248     0.1876 -6.65 2.92e-11

trsi                 -0.559     0.0837 -6.67 2.55e-11

altri1               -0.647     0.2157 -3.00 2.69e-03

mameve1               0.224     0.0693  3.23 1.23e-03

IsPhoneCall1:altri1   0.584     0.2369  2.46 1.37e-02

trsi:altri1           0.681     0.1657  4.11 3.93e-05

Log(scale)           -0.075     0.0264 -2.84 4.44e-03

Scale= 0.928

n= 800
```

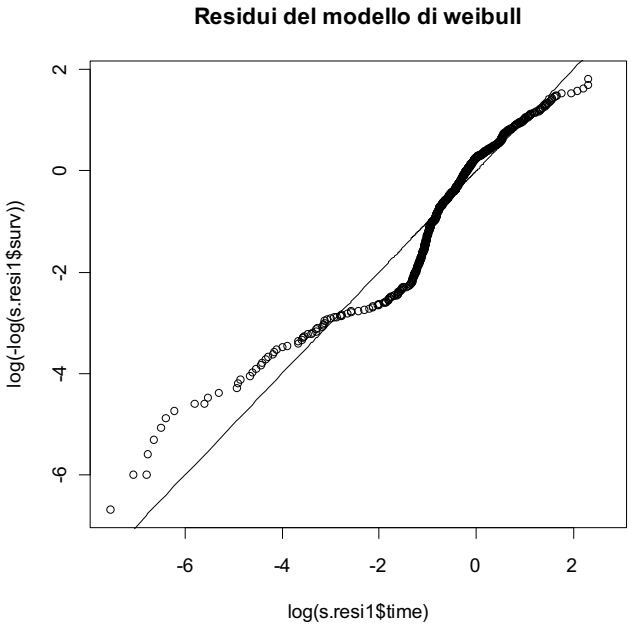
Ottengo cioè

$$T|\boldsymbol{z} = \frac{T_0}{exp(\boldsymbol{z};\boldsymbol{\beta})}$$

Dove:

$T_0 \sim Weibull(exp(-(\text{intercept}), 1/scale\ )$

$$exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp\left(1.868 - 1.248 * z_{i1} - 0.559 * z_{i2} - 0.647 * z_{i7} + 0.224 * z_{i6} + 0.584 * z_{i1} * z_{i7} + 0.681 * z_{i2} * z_{i7}\right)$$



il modello a tempi accelerati con distribuzione log-Logistica non riesce ad adattarsi bene.

Modello semi-parametrico di Cox.

li modello ridotto ottenuto attraverso l'insieme di stima fornisce il seguente output R:

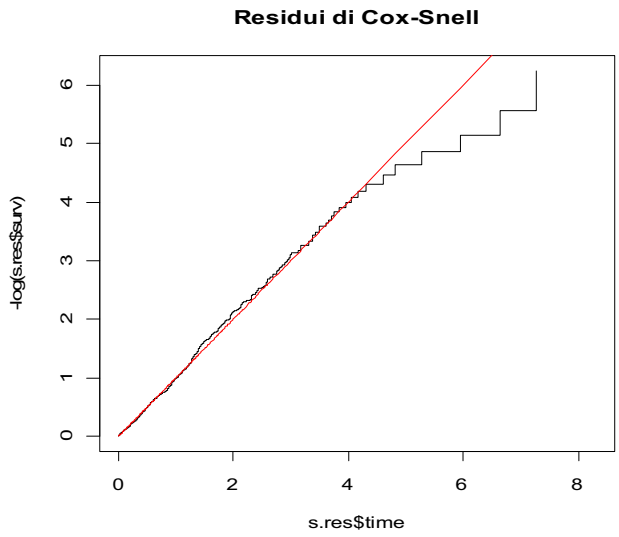
```
                      coef exp(coef) se(coef)     z       p
IsPhoneCall1         1.314     3.722   0.2051  6.41 1.5e-10
trsi                 0.569     1.767   0.0919  6.20 5.7e-10
altri1               0.804     2.234   0.2337  3.44 5.8e-04
IsPhoneCall1:altri1 -0.768     0.464   0.2579 -2.98 2.9e-03
trsi:altri1         -0.743     0.476   0.1791 -4.15 3.4e-05


Likelihood ratio test=118  on 5 df, p=0  n= 800
```

La funzione di rischio stimata quindi è:

$$h(t|\mathbf{z}) = h_0(t)exp(\mathbf{z};\boldsymbol{\beta})$$

dove

$$\exp(\boldsymbol{z};\boldsymbol{\beta}) = \exp\left(1.314 * z_{i1} + 0.569 * z_{i2} + 0.804 * z_{i7} - 0.768 * z_{i1} * z_{i7} - 0.743 * z_{i2} * z_{i7}\right)$$



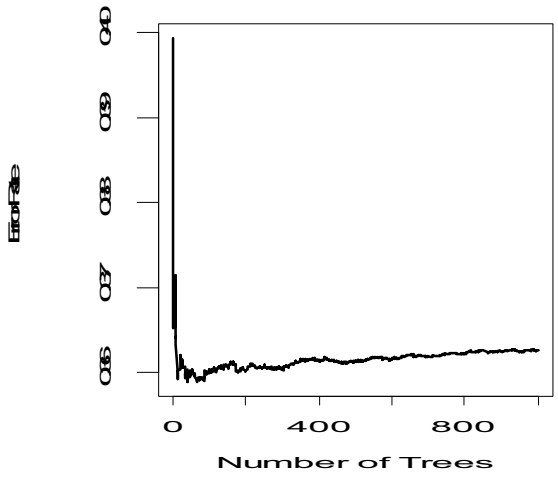
Il modello di Cox sembra adattarsi bene al campione. La funzione di Rischio cumulato si adatta bene alla retta e si discosta leggermente solo nell'ultima parte.

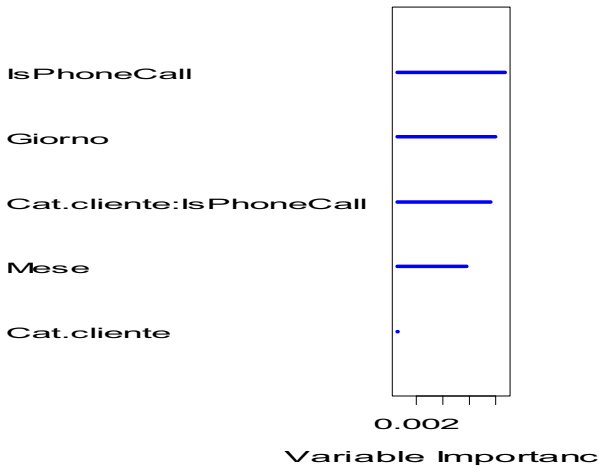
Modello non parametrico Random Survival Forest

Il modello ridotto ottenuto attraverso l'insieme di stima fornisce il seguente output R:

	Importance	Relative Imp
IsPhoneCall	0.0089	1.0000
Giorno	0.0081	0.9097
Cat.cliente:IsPhoneCall	0.0077	0.8661
Mese	0.0059	0.6614
Cat.cliente	0.0004	0.0493

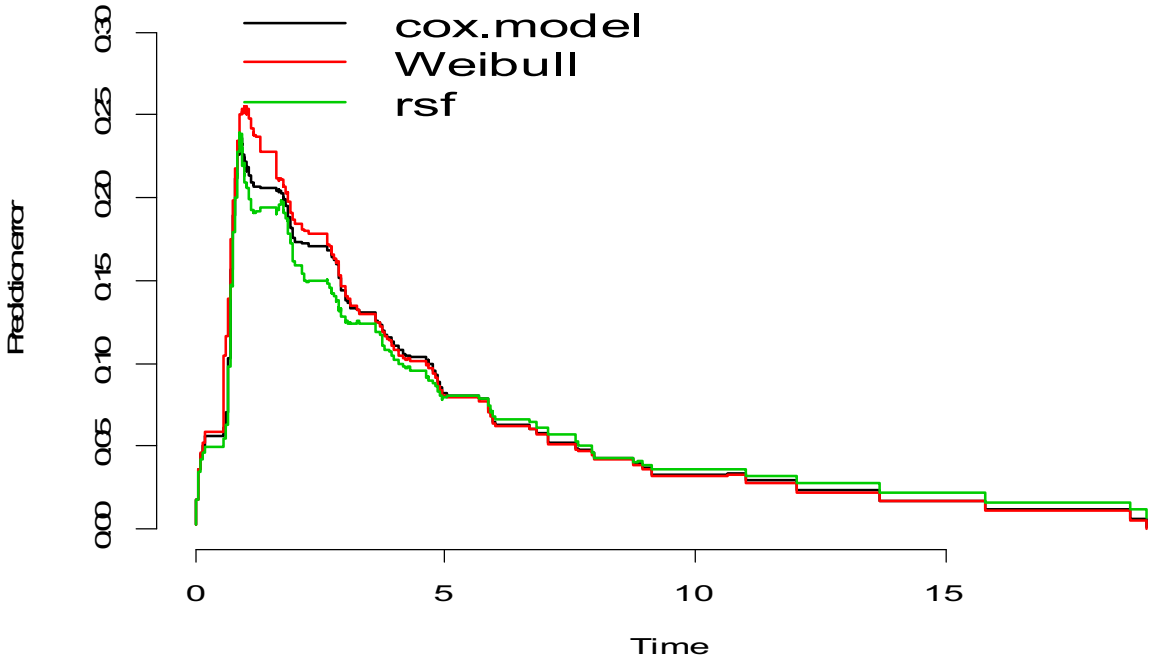
```
Sample size: 800                                    No. of variables tried at each split: 2
Number of deaths: 747                               Total no. of variables: 5
Number of trees: 1000                               Splitting rule: logrank
Minimum terminal node size: 3                       Estimate of error rate: 36.26%
Average no. of terminal nodes: 52.637
```





Le RSF stimate attraverso il campione di stima non ritengono più significativa la trasferta come variabile che può incidere nella durata dei ticket. Le variabili che hanno una maggiore importanza per il modello sono: IsPhoneCall, Giorno, l'interazione tra Cat.cliente e IsPhoneCall e Mese. Attraverso il comando *pec* calcolo le curve di errore di previsione dei tre modelli usando come strategie di calcolo la bootstrap *cross-validation* simulando il processo di calcolo per un B=100000 cercando di avvicinarmi cosi al risultato reale.

Quello che ottengo è il seguente risultato:



Le RSF sono quelle che sembrano avere un errore di previsione minore rispetto agli altri modelli, esclusa l'ultima parte in cui il modello di Weibull e di Cox sembrano prevedere meglio. Il massimo Brier Score raggiunto dal modello parametrico è circa 0.25 , mentre il modello semiparametrico e non parametrico circa del 0.23. Quindi la bontà delle previsioni dei tre modelli secondo Brier sono comunque stime accettabili.

CODICE REALIZZATO CON IL SOFTWARE R VERSIONE 2.9.0

```
LETTURA DEI DATI
rm(list=ls())
library(survival)
dati=read.table(file.choose(),sep=";",header=T)
dati
dati$Giorno=as.factor(dati$Giorno)
dati$Mese=as.factor(dati$Mese)
dati$Cat.cliente=as.factor(dati$Cat.cliente)
dati$IsPhoneCall=as.factor(dati$IsPhoneCall)
dati$Parte.Settimana=as.factor(dati$Parte.Settimana)
dati$tr=as.factor(dati$tr)
dati$cliente=as.factor(dati$cliente)
IDENTIFICAZIONE DEI VALORI ANOMALI
plot(dati$time)
boxplot(dati$time)
identify(dati$time)
plot(time~Cat.cliente)
a=dati[dati$time<21,]
attach(a)
ANALISI ESPLORATIVA

plot(time~IsPhoneCall,type="n",main='Grafico 1')
plot(time~Mese,main="Grafico 2")
plot(time~Cat.cliente,main="Grafico 3")
plot(time~Giorno,main="Grafico 4")
plot(time~tr,main="Grafico 5")
library(ellipse)
interaction.plot(Cat.cliente,IsPhoneCall,time,main='Interazioni')
interaction.plot(Cat.cliente,tr,time,main='Interazioni')

ANALISI MARGINALI DEI RISCHI e SOPRAVVIVENZE
FUNZIONE Di SOPRAVVIVENZA
fit<-survfit(Surv(time,status)~1)
par(mfrow=c(1,2))
plot(fit$time,fit$surv,type="s",xlab="time",ylab="S.km(t)",main="Survival
Function")
lines(fit$time,fit$upper,type="s",col=1,lty=2)
lines(fit$time,fit$lower,type="s",col=1,lty=2)


FUNZIONE DI RISCHIO
plot(fit$time,-log(fit$surv),type="n",xlab="time",ylab="H(t)",main="Hazard
Function")
lines(fit$time,-log(fit$upper),type="s",col=1,lty=2)
lines(fit$time,-log(fit$lower),type="s",col=1,lty=2)
lines(fit$time,-log(fit$surv),type="s",col="black")

Funzione di Rischio E Di SOPRAVVIVENZA marginali

CATEGORIA CLIENTI

fit1=survfit(Surv(time,status)~Cat.cliente)
strato=c(rep(0,429),rep(1,248),rep(2,280))
par(mfrow=c(1,2))
plot(fit1$time,fit1$surv,ylim=c(0,1),xlab="time",ylab="Estimated survival
function",type="n",main="S(t) marginale per Cat.cliente")
lines(fit1$time[strato==0],fit1$surv[strato==0],type="s",lty=1)#cat A
lines(fit1$time[strato==1],fit1$surv[strato==1],type="s",lty=2)  #cat B
```

```
lines(fit1$time[strato==2],fit1$surv[strato==2],type="s",lty=3)# cat C
legend(10,0.55,legend=c("A","B","C"),lty=1:3)


plot(fit1$time,log(-log(fit1$surv)),xlab="time",ylab="log(Estimated Hazard
F.)",type='n',main="H(t) marginale per Cat.cliente")
lines(fit1$time[strato==0],log(-log(fit1$surv[strato==0])),type='s',lty=1)
lines(fit1$time[strato==1],log(-log(fit1$surv[strato==1])),type='s',lty=2)
lines(fit1$time[strato==2],log(-log(fit1$surv[strato==2])),type="s",lty=3)
legend(10,-2,legend=c("A","B","C"),lty=1:3)

-applichiamo il test log-rank
survdiff(Surv(time,status)~Cat.cliente)

ISPHONECALL
fit2=survfit(Surv(time,status)~IsPhoneCall)
fit2$strata
strato=c(rep(0,224),rep(1,661))
par(mfrow=c(1,2))
plot(fit2$time,fit2$surv,ylim=c(0,1),xlab="time",ylab="Estimated survival
function",type="n",main="S(t) marginale per IsPhoneCall")
lines(fit2$time[strato==0],fit2$surv[strato==0],type="s",lty=1)#pririt? 1 Bassa
lines(fit2$time[strato==1],fit2$surv[strato==1],type="s",lty=2)  #priorità 2
media
legend(10,0.6,legend=c("Mail","Telefono"),lty=1:2)

plot(fit2$time,log(-log(fit2$surv)),xlab="time",ylab="log(Estimated Hazard
F.)",type='n',main="H(t) marginale per IsPhoneCall")
lines(fit2$time[strato==0],log(-log(fit2$surv[strato==0])),type='s',lty=1)
lines(fit2$time[strato==1],log(-log(fit2$surv[strato==1])),type='s',lty=2)
legend(10,-1.6,legend=c("Mail","Telefono"),lty=1:2)

-applichiamo il test log-rank
survdiff(Surv(time,status)~IsPhoneCall)

#creo nuova variabile
#altri=rep(0,1182)
#se vale 0 sono A e B
#se vale 1 sono C

for(i in 1:1182){
if(a$Cat.cliente[i]=='C')
altri[i]=1
else
altri[i]=0
}
altri=as.factor(altri)
ALTRI
fit2=survfit(Surv(time,status)~altri)
fit2$strata
strato=c(rep(0,662),rep(1,280))
par(mfrow=c(1,2))
plot(fit2$time,fit2$surv,ylim=c(0,1),xlab="time",ylab="Estimated survival
function",type="n",main="S(t) marginale per altri")
lines(fit2$time[strato==0],fit2$surv[strato==0],type="s",lty=1)#BT
lines(fit2$time[strato==1],fit2$surv[strato==1],type="s",lty=2)  #altri

legend(10,0.6,legend=c("B-T","altri"),lty=1:2)
```

```
plot(fit2$time,log(-log(fit2$surv)),xlab="time",ylab="log(Estimated Hazard
F.)",type='n',main="H(t) marginale per altri")
lines(fit2$time[strato==0],log(-log(fit2$surv[strato==0])),type='s',lty=1)
lines(fit2$time[strato==1],log(-log(fit2$surv[strato==1])),type='s',lty=2)
legend(10,-1.6,legend=c("B-T","altri"),lty=1:2)

TRAFERTA

fit2=survfit(Surv(time,status)~tr)
fit2$strata
strato=c(rep(0,603),rep(1,354))
par(mfrow=c(1,2))
plot(fit2$time,fit2$surv,ylim=c(0,1),xlab="time",ylab="Estimated survival
function",type="n",main="S(t) marginale per IsPhoneCall")
lines(fit2$time[strato==0],fit2$surv[strato==0],type="s",lty=1)
lines(fit2$time[strato==1],fit2$surv[strato==1],type="s",lty=2)
legend(10,0.6,legend=c("no","si"),lty=1:2)

plot(fit2$time,log(-log(fit2$surv)),xlab="time",ylab="log(Estimated Hazard
F.)",type='n',main="H(t) marginale per trasferta")
lines(fit2$time[strato==0],log(-log(fit2$surv[strato==0])),type='s',lty=1)
lines(fit2$time[strato==1],log(-log(fit2$surv[strato==1])),type='s',lty=2)
legend(10,-1.6,legend=c("no","si"),lty=1:2)


MODELLI PARAMETRICI

MODELLO CON DISTRIBUZIONE WEIBULL
fit1<-survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Mese+Giorno)
summary(fit1)
#scarto mese
fit1<-survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno)
summary(fit1)
#unisco le due categorie merc e ven e vediamo cosa succede
dim(a)
gmv=rep(0,1182)
for(i in 1:1182){
if(Giorno[i]=='mer'|| Giorno[i]=='ven')
gmv[i]=2
else
gmv[i]=1
}
gmv=as.factor(gmv)

fit1<-survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+gmv)
summary(fit1)


RESIDUI MODELLO CON DISTRIBUZIONE WEIBULL
resi1.s<-(fit1$y[,1]-fit1$linear)/fit1$scale
s.resi1<-survfit(Surv(exp(resi1.s),status)~1)
plot(log(s.resi1$time),log(-log(s.resi1$surv)),xlab="log(t)",ylab="log(-
log[S.km(t)])",main="Bontà adattamento modello con dist Weibull")
lines(log(s.resi1$time),log(s.resi1$time))

MODELLO CON DISTRIBUZIONE LOG-NORMALE

fit3=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno+Mese,dist='lo
gnorm')
summary(fit3)
#scarto mese
```

```
fit3=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno,dist='lognorm
')
summary(fit3)
#scarto  Giorno
fit3=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente,dist='lognorm')
summary(fit3)

RESIDUI MODELLO CON DISTRIBUZIONE LOG-NORMALE

resi3.s<-(fit3$y[,1]-fit3$linear)/fit3$scale
s.resi3<-survfit(Surv(exp(resi3.s),status)~1)
plot(log(s.resi3$time),qnorm(1-s.resi3$surv),ylab="qnorm(1-
S.km(t))",xlab="log(t)",main="Bontà adattamento modellon con dist. Log-Normale
")
lines(log(s.resi3$time),log(s.resi3$time))

MODELLO CON DISTRIBUZIONE LOG-LOGISTICA
fit2=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno+Mese,dist='lo
glogi')
summary(fit2)
#tolgo Mese
fit2=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno,dist='loglogi
')
summary(fit2)
tolgo Giorno
fit2=survreg(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente,dist='loglogi')
summary(fit2)


RESIDUI MODELLO CON DISTRIBUZIONE LOG-LOGISTICA

resi2.s<-(fit2$y[,1]-fit2$linear)/fit2$scale
s.resi2<-survfit(Surv(exp(resi2.s),status)~1)
plot(log(s.resi2$time),log((1/s.resi2$surv)-
1),xlab="log(t)",ylab="log(1/S.km(t)-1)",main="Bontà adattamento modello con
dist. Log-logistica")
lines(log(s.resi2$time),log(s.resi2$time))

MODELLO DI COX

fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno+Mese,method="bre
slow")
summary(fit)
#tolgo mese
fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Giorno,method="breslow"
)
summary(fit)
#visto che è significativo   ven uso una variabile dicotomica #Parte.Settimana
fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente+Parte.Settimana,method=
"breslow")
summary(fit)
#tolgo parte settimana
fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*Cat.cliente,method="breslow")
summary(fit)

RESIDUI DI COX- SNELL
resi.m<-residuals(fit,type='mart')
resi.cs<-status-resi.m
s.res<-survfit(Surv(resi.cs,status)~1)
par(pty="s")
plot(s.res$time,-log(s.res$surv),type='s',main="Residui di Cox-Snell")
```

```
lines(s.res$time,s.res$time,col=2)

Modello di Cox con altri al posto di Cat.cliente

Facciamo questo modello perchè abbiamo visto che marginalmente i rischi sono
proporzionali
fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno+Mese,method="breslow")
summary(fit)
tolgo mese

fit=coxph(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno,method="breslow")
summary(fit)
tolgo giorno
fit<-coxph(Surv(time,status)~(IsPhoneCall+tr)*altri,method="breslow")
summary(fit)


RESIDUI  COX-SNELL
resi.m<-residuals(fit,type='mart')
resi.cs<-status-resi.m
s.res<-survfit(Surv(resi.cs,status)~1)
par(pty="s")
plot(s.res$time,-log(s.res$surv),type='s',main="Residui di Cox-Snell con due
cat")
lines(s.res$time,s.res$time,col=2)
RESIDUI DI DEVIANZA
res=resid(fit,'dev')
plot(res)

library(randomSurvivalForest)

modello=rsf(Survrsf(time,status)~Mese+Giorno+tr+Cat.cliente+IsPhoneCall,data=a,n
tree=1000,forest=T)
modello
#AGGIUNGIAMO L'INTERAZIONE
modello=rsf(Survrsf(time,status)~Mese+Giorno+tr+Cat.cliente+IsPhoneCall+
IsPhoneCall:Cat.cliente,data=a,ntree=1000,forest=T)
modello

# CALCOLO DELL'ERRORE DI PREVISIONE ATTRAVERSO L'INSIEME DI STIMA E DI VERIFICA

library(pec)
library(rms)
library(party)
library(survival)
library(MASS)
#index=sample(1:1182)
#index
#write(index,file="index.txt")
index=scan(file.choose())
#allora i primi 800 insieme di stima.
#I restanti 382  mi serviranno per fare le previsioni.

indicitraning=index[1:800]
indicetest=index[801:1182]

#mi estraggo i dati dal mio campione estraendo dal data set le osservazioni
#corrispondenti.

datitraning=a[indicitraning,]
datitest=a[indicetest,]
```

```
mameve=rep(0,1182)

#vale 0 se giorno è lun o giovedi
#vale 1 se giorno è martedi mercoledi venerdi
for(i in 1:1182){
if(a$Giorno[i]=='lun' |  a$Giorno[i]=='gio')
mameve[i]=0
else
mameve[i]=1
}
mameve=as.factor(mameve)
a=cbind(a,altri,mameve)



Costruzione dei modelli attraverso l'insieme di stima
MODELLO DI WEIBULL

fit1_=psm(survreg(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno+Mese,data=dati
traning),data=datitraning,dist="weibull")
fit1_
# tolgo mese
fit1_=psm(survreg(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno,data=datitrani
ng),data=datitraning,dist="weibull")
fit1_
# sostituisco la variabile Giorno con mameve visto che mar,mer,ven #risultano
significativi.
fit1_=psm(survreg(Surv(time,status)~(IsPhoneCall+tr)*altri+mameve,data=datitrani
ng),data=datitraning,dist="weibull")
fit1_



MODELLO DI COX
fit_<coxph(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno+Mese,method="breslow"
,data=datitraning)
fit_
#tolgo Mese
fit_<coxph(Surv(time,status)~(IsPhoneCall+tr)*altri+Giorno,method="breslow",data
=datitraning)
fit_
#tolgo Giorno
fit_<coxph(Surv(time,status)~(IsPhoneCall+tr)*altri,method="breslow",data=datitr
aning)
fit_

RSF
modello_=rsf(Survrsf(time,status)~Mese+Giorno+tr+Cat.cliente+IsPhoneCall,data=da
titraning,ntree=1000,forest=T)
plot(modello_)
#TOLGO TR

modello_=rsf(Survrsf(time,status)~Mese+Giorno+Cat.cliente+IsPhoneCall+
Cat.cliente: IsPhoneCall,data=datitraning,ntree=1000,forest=T)
plot(modello_)
#AGGIUNGO L'INTERAZIONE

#CODICE PER CALCOLARE LE PREVISIONI CON UNA RSF
predictSurvProb.rsf <- function(object,newdata,times,train.data=list(status)){
```

```
require(randomSurvivalForest)
H <- predict.rsf(object=object,test=newdata)$ensemble
S <- exp(-H)
Time <- object$timeInterest
p <- cbind(1,S)[,1+sindex(jump.times=Time,eval.times=times),drop=FALSE]
p
}
# calcolo dell'errore di previsione dei tre modelli
PredError <- pec(list("cox-model"=fit_,"Weibull"=fit1_,"rsf"=modello_),
                 formula=Surv(time,status)~1,
                 data=datitest,
                 exact=TRUE,
                 cens.model="marginal",
                 replan="cvK",
                 B=100000,
                 verbose=TRUE)

plot(PredError,ylim=c(0,0.010))
```

Bibliografia

1. Ishwaran H. e Kogalur U.B. “Random Survival Forest for R”, R News, Vol.7/2, pg 25-31
2. Ishwaran H. e Kogular U.B. (2007) randomSurvivalForest Package.
3. Marubini E. e Valsecchi M.G., Analysing survival data from clinical trials and observational studies, John Wiley & Sons, 1995.
4. Gerds, Mogensen, Ishwaran (2010) Evaluating random forest for survival analysis using prediction error curves .
5. Gerds, Mogensen, Ishwaran pec Package.
6. Azzalini A. e Scarpa B. (2004) Analisi dei dati e Data mining, Milano, Springer – Verlag Italia.
7. +Thereau T.M. e Grambsh P.M. (2000) Modeling Survival Data: Extending the Cox model, Usa, Spring-Verlag.
8. Zieffler A. Harring R. Randomization and Bootstrap Methods Using R (2011)