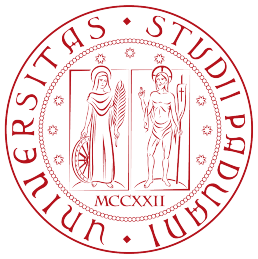

UNIVERSITÀ DEGLI STUDI DI PADOVA
Dipartimento di Ingegneria dell'Informazione
Corso di Laurea in Bioingegneria

TESI DI LAUREA MAGISTRALE

Random Survival Forests per la stratificazione del rischio in pazienti affetti da Sclerosi Laterale Amiotrofica

RELATORE: Prof.ssa Barbara Di Camillo

CORRELATORE: Dott. Alessandro Zandonà

LAUREANDA: Erica Tavazzi

Padova, 11 settembre 2017

A.A. 2016-2017

Alla mia famiglia
a Roberto
e ai miei amici,
solidi Alberi.

Abstract

Amyotrophic Lateral Sclerosis (ALS) is a fatal neurodegenerative disease, which selectively affects the motor neurons that control voluntary muscles. This condition, which probably results from the complex interplay from genetic and environmental factors, causes the progressive loss of vital functions.

A large variability between patients is observed either in the site of onset, in the symptoms progression and in the survival expectation, making it challenging to predict the disease course at the level of the individual patient. Currently, there is no available clinical tool to differentiate between these patients starting from their symptoms, nor to predict the clinical progression or the death risk.

The disease hetereogenity is at the origin of the failure of several clinical trials, conducted to detect the potential effect of treatments and medications on ALS patients. As a matter of fact, the more heterogeneous the disease, the more difficult to predict how a given patient's condition will progress and thereby to demonstrate the effect of a potential therapy.

In this work, differences between the clinical disease manifestations are investigated and a stratification study is performed, in order to partition the ALS patients' population into meaningful subgroups.

Using demographic and clinical data from the Amyotrophic Lateral Sclerosis Clinical Trial database PRO-ACT, a feature-based model of risk prediction exploiting the Random Survival Forest algorithm is developed. From the patients' features, the estimates of the individual death risk and of the survival probability are obtained. Furthermore, features are ranked in terms of their contribution to predict the patient prognosis.

Exploiting the survival estimates, stratification of the patients into homogeneous subgroups through Hierarchial and K-means Clustering is performed. Relying on the similarity of the survival curves, 2 and 3 potential subgroups have been respectively identified.

Abstract

La Sclerosi Laterale Amiotrofica (SLA) è una patologia neurodegenerativa fatale, che colpisce i neuroni che controllano la muscolatura volontaria. Tale condizione, che risulta probabilmente dalla complessa interazione tra fattori genetici ed ambientali, causa la perdita progressiva delle funzioni vitali.

Si osserva etereogeneità nei pazienti sia per quanto riguarda il sito di esordio della malattia, sia nella progressione dei sintomi e nella prospettiva di sopravvivenza dei pazienti: tale variabilità tra i soggetti rende difficoltoso delineare la prognosi della malattia a livello di singolo paziente.

Attualmente, non esistono strumenti clinici in grado di differenziare tra le diverse tipologie di manifestazione della patologia a partire dalla sintomatologia, né si è in grado di predire la progressione clinica o il rischio di morte.

L'eterogeneità della SLA ha contribuito, negli ultimi anni, al fallimento di numerosi studi clinici condotti al fine di individuare potenziali trattamenti farmacologici efficaci. Infatti, le difficoltà nella previsione del decorso naturale della patologia rendono difficilmente individuabili eventuali effetti, positivi o negativi, dei trattamenti.

In questo lavoro di tesi, ci si è posti l'obiettivo di approfondire le differenze tra le diverse manifestazioni cliniche della malattia, realizzando uno studio di stratificazione, che permetta di suddividere i pazienti SLA in sottogruppi significativi.

A partire da dati clinici e demografici estratti dal database PRO-ACT (Amyotrophic Lateral Sclerosis Clinical Trial), si è implementato l'algoritmo di classificazione Random Survival Forests.

A partire dalle variabili registrate in ciascun paziente, si sono ottenute le predizioni delle curve di rischio di morte e della probabilità di sopravvivenza. Inoltre, si è ricavata una stima della significatività di ciascuna variabile nella predizione, sulla base del suo contributo nel delineare la prognosi del paziente.

Alle predizioni survival ottenute sono state applicate tecniche di stratificazione, sfruttando gli algoritmi di Clustering Gerarchico e K-means, con lo scopo di raggruppare i pazienti in sottogruppi omogenei. Basandosi sulla similarità delle curve di sopravvivenza, tali approcci hanno permesso di individuare rispettivamente 2 e 3 possibili suddivisioni del pool di pazienti.

Indice

Elenco delle figure

Elenco delle tabelle

Introduzione

La Sclerosi Laterale Amiotrofica (SLA) è una malattia neurodegenerativa che colpisce i motoneuroni, causando progressivamente la perdita delle funzioni vitali. Le cause di questa patologia sono attualmente sconosciute, ma, verosimilmente, la sua insorgenza è dovuta alla presenza simultanea di più fattori, tra cui una predisposizione genetica.

La SLA si presenta con eterogeneità di sintomi e decorso clinico, rendendo difficoltosa la predizione della prognosi a livello di singolo paziente. Attualmente, non esiste alcuno strumento che permetta di differenziare i pazienti a partire dalle manifestazioni cliniche della malattia, né che sia in grado di predirne la progressione clinica o il rischio di morte. Queste difficoltà di previsione rappresentano purtroppo un notevole carico psicologico per i pazienti e le famiglie, che affrontano la diagnosi senza ricevere indicazioni sul futuro con la malattia.

Negli ultimi anni, sono stati condotti diversi trial clinici per individuare gli effetti di cure farmacologiche su pazienti SLA, senza però riuscire ad individuare una terapia in grado di arrestarne la progressione.

Il fallimento dei trial clinici è dovuto principalmente proprio all'intrinseca eterogeneità della manifestazione della malattia: la variabilità nella progressione, infatti, non permette di stabilirne il decorso naturale e di verificare potenziali effetti migliorativi dovuti alle terapie.

Nel tentativo di superare queste difficoltà di analisi della patologia, si può progettare uno studio di stratificazione della popolazione di pazienti, che permetta di suddividere gli stessi in sottogruppi dalle caratteristiche simili (età di esordio, prevalenza del motoneurone coinvolto, sede di esordio, tempo di evoluzione, ecc.). Tale ricerca può portare ad approfondire la conoscenza della malattia e, potenzialmente, fungere da supporto sia nella pianificazione che nell'interpretazione di futuri trial clinici.

In questo lavoro di tesi, ci si è proposti di stratificare i pazienti a partire dai loro dati clinici e demografici raccolti nelle visite successive alla diagnosi. Si è fatto uso di dati relativi a un pool di 1822 pazienti, subset del database open source PRO-ACT (*Pooled Resource Open-Access ALS Clinical Trials Database*). Si tratta di informazioni demografiche, dati di laboratorio e valutazioni clinico/funzionali sul decorso della malattia relativi a pazienti affetti da SLA, che, nel tempo, hanno preso parte a diversi trial clinici.

Per ciascun paziente, si sono prese in considerazione le variabili misurate nei primi tre mesi dall'ingresso nel trial clinico. Si è utilizzato un metodo di Machine Learning denominato *Random Survival Forests*, che ha permesso di ricavare, a partire dai dati, una predizione della prognosi tramite il calcolo del rischio di morte del paziente e la stima della probabilità di sopravvivenza in funzione del tempo.

L'applicazione di tale metodo ha consentito, inoltre, di analizzare in maniera automatica l'importanza delle variabili nella definizione dello stato di salute futuro del paziente. Tali risultati, confrontati con quelli presenti in letteratura, possono essere sfruttati nella ricerca clinica per modellizzare la progressione della malattia ed individuare nuovi possibili marcatori prognostici.

Si sono poi sperimentati alcuni metodi di clustering non supervisionato, come il *Clustering Agglomerativo Gerarchico* ed il *Clustering K-Means*, ottenendo la stratificazione dei pazienti a partire delle loro curve di sopravvivenza, rispettivamente in 2 e 3 sottogruppi.

La tesi si sviluppa come segue:

Nel **Capitolo 1** si presentano il contesto biologico e le scale di valutazione funzionale utilizzate in clinica per stabilire la gravità della malattia nelle sue varie fasi. Si introduce, inoltre, il database PRO-ACT, da cui sono estratti i dati utilizzati per questo lavoro.

Il **Capitolo 2** presenta il metodo di classificazione Random Survival Forests (RSF), utilizzato per ottenere le curve di sopravvivenza dei pazienti. L'algoritmo costituisce l'estensione all'analisi di dati di sopravvivenza del metodo di classificazione Random Forests, ottenuto come aggregazione di più Alberi Decisionali.

Nel **Capitolo 3** si presenta nel dettaglio la struttura dei dati utilizzati e si espongono le procedure di preparazione degli stessi, applicate per generare i dataset per la classificazione e gestirne le variabili.

Nel **Capitolo 4** si illustrano le scelte di parametrizzazione della Foresta Survival per lo specifico problema trattato.

Nel **Capitolo 5** vengono descritti ed analizzati i risultati ottenuti applicando il metodo RSF, soffermandosi sulle variabili più significative per la definizione della prognosi e sulle curve di rischio e sopravvivenza ottenute per i soggetti.

Nel **Capitolo 6** vengono presentati i metodi di Clustering Gerarchico Agglomerativo e K-Means e se ne descrive la loro applicazione alle curve di sopravvivenza dei soggetti per fare stratificazione.

Nel **Capitolo 7**, infine, si traggono le conclusioni sul lavoro svolto e si presentano le possibili direzioni di ricerca futura.

Capitolo 1

Contesto biologico

1.1 La Sclerosi Laterale Amiotrofica

La Sclerosi Laterale Amiotrofica, indicata comunemente con l'acronimo SLA e conosciuta anche come "Morbo di Lou Gehrig", o "malattia del motoneurone", è una patologia neurodegenerativa fatale.

La SLA colpisce selettivamente le cellule nervose cerebrali (primo motoneurone) e del midollo spinale (secondo motoneurone) che controllano i muscoli volontari, provocando la perdita progressiva delle funzioni vitali, attaccando in particolare il movimento, la parola, la masticazione, la deglutizione e la respirazione. Gli organi interni non vengono compromessi e, generalmente, le attività mentali sono preservate. Col progredire della malattia, si osserva un graduale peggioramento della capacità respiratoria, fino alla necessità, nelle ultime fasi, dell'utilizzo della ventilazione artificiale passiva. Solitamente, la morte avviene per polmonite *ab ingestis* o per insufficienza respiratoria grave.

Questa condizione deriva probabilmente dall'interazione tra una predisposizione genetica ed altri fattori, come quello ambientale, che possono contribuire allo sviluppo della malattia.

L'incidenza si colloca intorno ai 3 casi ogni 100.000 abitanti/anno, e la prevalenza è pari a 10 ogni 100.000 abitanti, nei paesi occidentali. Attualmente, sono circa 6.000 i malati in Italia. La malattia colpisce entrambi i sessi, anche se vi è una lieve preponderanza nel sesso maschile [1].

La diagnosi viene formulata per esclusione di altre malattie neuromuscolari con manifestazioni simili, ricostruendo le fasi di comparsa dei sintomi ed effettuando alcuni esami clinici di approfondimento.

Non esiste ancora una cura per la SLA e i trattamenti attualmente disponibili riescono soltanto ad alleviarne i sintomi, rallentando parzialmente, e solo in alcuni casi, la progressione della malattia. L'unico farmaco attualmente approvato per il trattamento della SLA è il Riluzolo (Rilutek), che ha un effetto limitato su alcuni pazienti, estendendone l'aspettativa di vita di due-tre mesi [32].

Mediamente, l'aspettativa di sopravvivenza dopo la manifestazione dei primi sintomi è intorno ai 3-5 anni [27, 40], per quanto circa il 10% dei pazienti sopravviva fino a 10 anni

o più [31]. Una notevole variabilità tra i pazienti è riscontrata non solo nell'aspettativa di sopravvivenza, ma anche nel sito biologico di manifestazione dei primi sintomi e nella loro progressione. Tale variabilità rende difficoltosa la predizione del decorso della malattia a livello di singolo paziente.

Si ipotizza che esistano forme diverse di SLA, con meccanismi completamente differenti, che convergono nella stessa diagnosi. Si è attualmente in grado di classificare la SLA in alcuni sottotipi, determinati in alcuni casi dal sito di esordio della patologia. Nella maggior parte dei casi, la SLA si presenta nella sua forma spinale, aggredendo per primi i motoneuroni del midollo spinale e facendo insorgere inizialmente sintomi motori a livello degli arti. In circa un terzo dei casi, invece, la lesione dei motoneuroni interessa il tronco cerebrale/bulbare. Questa forma, che prende il nome di forma bulbare, si rivela la più aggressiva e presenta tra le prime manifestazioni difficoltà nel masticare, nell'ingoiare e nel parlare. Pur lasciando generalmente inalterate le capacità cognitive, in una bassa percentuale di casi queste possono essere colpite, ed è questo il caso della SLA con demenza fronto-temporale. Questa forma è la più comune tra i malati con una storia familiare di demenza e comporta, tra i sintomi, profondi cambiamenti della personalità. La distinzione tra i sottotipi, comunque, non è sempre netta e identificabile, coesistendo in certi casi sintomi misti, specie nel caso di diagnosi tardiva.

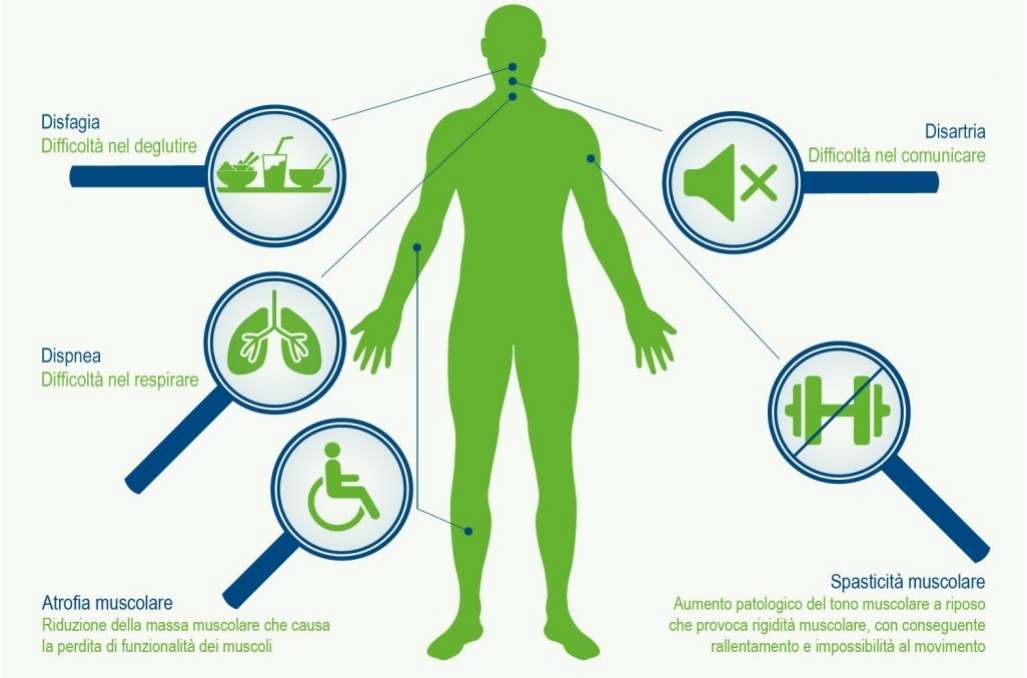

Figura 1.1: Sintomi associati alla Sclerosi Laterale Amiotrofica.

1.2 Le scale di valutazione funzionale

Dopo la formulazione della diagnosi di SLA, il paziente viene sottoposto regolarmente a visite mediche per monitorare l'evoluzione delle manifestazioni della patologia. In Figura 1.1 sono riportati i principali sintomi associati alla malattia [1]. Gli effetti limitanti si riferiscono a quattro domini biologici specifici: movimento/autocura, deglutizione, comunicazione e respirazione.

Per definire in maniera oggettiva lo stato di progressione della malattia (detto *stadiazione*), in clinica si fa uso di scale di valutazione funzionale. Queste stime permettono di monitorare l'avanzamento della patologia, nonché di impostare il percorso terapeutico ed assistenziale del paziente e, nel caso di trial clinici, di esaminare gli effetti dei trattamenti sul paziente.

Attualmente, la scala funzionale *ALSFRS* (ALS Functional Rating Scale) [9] e la sua versione revisionata *ALSFRS-R* [10] rappresentano lo standard in clinica.

La scala ALSFRS raccoglie informazioni relative allo stato clinico del paziente tramite la compilazione di un questionario di 10 domande. Tali domande, che indagano la compromissione delle funzioni vitali e i deficit nello svolgimento di attività quotidiane, sono relative ai domini biologici attaccati dalla malattia. Nello specifico, per ciascun dominio vengono individuati dei task: linguaggio, salivazione, deglutizione, scrittura manuale, manualità strumentale a tavola ed utilizzo di utensili (con distinzione a seconda che il paziente sia alimentato artificialmente o meno), vestizione e cura dell'igiene personale, mobilità a letto e gestione delle coperte, deambulazione, capacità di salire le scale, respirazione. A ciascuna domanda viene assegnato un punteggio di gravità compreso tra 0 e 4, dove 0 indica la totale perdita di autonomia e 4 la completa autosufficienza. Sommando i punteggi, si ottiene lo score totale ALSFRS, che è quindi compreso tra 0 e 40. Si riporta nella Figura 1.2 il questionario di valutazione ALSFRS.

Nella versione revisionata ALSFRS-R, l'area funzionale della respirazione (domanda Q.10 nell'ALSFRS) viene indagata ulteriormente, facendo riferimento a task più specifici: dispnea, ortopnea, insufficienza respiratoria. In questo modo, nella scala revisionata le domande sono in totale 12, con un punteggio finale compreso tra 0 e 48. Il questionario ALSFRS-R è riportato in Figura 1.3.

Valutazioni alternative della stadiazione possono essere ottenute tramite l'uso delle scale *KING* (King's College Staging system) [39] e *ALS-MITOS* (ALS Milano-Torino Staging) [12].

Il sistema KING definisce 5 stadi della malattia: per i primi tre stadi, viene considerato il numero di aree funzionali coinvolte; il quarto viene raggiunto quando si manifesta la necessità di gastrostomia e ventilazione assistita; il quinto stadio indica la morte del paziente. La stadiazione così ottenuta è riportata in Figura 1.4 [17].

Il sistema MITOS sfrutta la suddivisione negli stessi quattro domini biologici cui fa riferimento ALSFRS(-R), fissando per ciascuno una soglia di compromissione della

Q.1 - LINGUAGGIO	
Processo fonatorio normale	4
Alterazione evidenziabile del linguaggio	3
Linguaggio intellegibile con ripetizioni	2
Linguaggio associato a comunicazione non vocale	1
Impossibilità ad un linguaggio utile	0
Q.2 – SALIVAZIONE	
Normale	4
Lieve ma chiaro eccesso di saliva in bocca; vi può essere scolo di saliva nelle ore notturne	3
Moderato eccesso di saliva; vi può essere minimo scolo di saliva nelle ore diurne	2
Marcato eccesso di saliva; con scolo di saliva nelle ore diurne	1
Marcato scolo di saliva; vi è necessità continua di asciugarla	0
Q.3 – DEGLUTIZIONE	
Normale	4
Iniziali problemi alimentari; occasionalmente va di traverso	3
Necessità di modificare la consistenza delle diete	2
Necessità di alimentazione enterale supplementare	1
Non in grado di deglutire (alimentazione esclusivamente enterale o parenterale)	0
Q.4 - SCRIVERE A MANO (con la mano già dominante)	
Normale	4
Rallentato o approssimativo; tutte le parole sono leggibili	3
Non tutte le parole sono leggibili	2
In grado di afferrare la penna ma non di scrivere	1
Incapace di afferrare la penna	0
Q.5a - TAGLIARE IL CIBO ED USARE UTENSILI (paziente senza PEG)	
Normale	4
Un po' rallentato e goffo, ma non necessita di aiuto	3
Può tagliare la maggior parte dei cibi, anche se in modo rallentato o goffo; è necessario un certo aiuto	2
Il cibo deve essere tagliato da altri, ma ancora in grado di portarsi il cibo alla bocca da solo, anche se lentamente	1
Deve essere nutrito	0
Q.5b - PREPARARE IL CIBO E USARE UTENSILI (paziente con PEG)	
Normale	4
Maldestro ma in grado di eseguire tutte le manipolazioni da solo	3
Necessario un certo aiuto con dispositivi di fissaggio e chiusura	2
In grado di fornire un minimo di aiuto a chi lo assiste	1
Incapace di eseguire qualsiasi aspetto di questi compiti	0

Q.6 - ABBIGLIAMENTO ED IGIENE PERSONALE	
Normale	4
Bada a se stesso in modo indipendente e completo con sforzo o ridotta efficienza	3
Necessita di assistenza non continuativa e di metodi sostituitivi	2
Necessita di aiuto consistente per la cura di sé	1
Completamente dipendente	0
Q.7 - GIRARSI NEL LETTO E AGGIUSTARSI LE COPERTE	
Normale	4
Un po' rallentato e goffo, ma non necessita di aiuto	3
Può girarsi da solo o sistemare le coperte, ma con grande difficoltà	2
Può iniziare il movimento, ma non girarsi o sistemare le coperte da solo	1
Completamente dipendente	0
Q.8 - CAMMINARE	
Normale	4
Iniziali difficoltà nella deambulazione	3
Cammina con necessità di assistenza (con qualsiasi tipo di ausilio o ortesi)	2
Solo movimenti funzionali che non permettono la deambulazione	1
Nessun movimento utile o finalizzabile degli arti inferiori	0
Q.9 - SALIRE LE SCALE	
Normale	4
Rallentato	3
Lieve instabilità	2
Necessita di assistenza (compreso l'uso del manocorrente)	1
Non può farlo	0
Q.10 - RESPIRAZIONE	
Normale	4
Dispnea in attività fisiche minimali (es. camminare, parlare)	3
Dispnea a riposo	2
Necessità di assistenza ventilatoria intermittente (es. notturna)	1
Dipendenza assoluta dal ventilatore	0

Figura 1.2: Scala di valutazione funzionale ALSFRS.

Q.1 - LINGUAGGIO	
Processo fonatorio normale	4
Alterazione evidenziabile del linguaggio	3
Linguaggio intellegibile con ripetizioni	2
Linguaggio associato a comunicazione non vocale	1
Impossibilità ad un linguaggio utile	0
Q.2 – SALIVAZIONE	
Normale	4
Lieve ma chiaro eccesso di saliva in bocca; vi può essere scolo di saliva nelle ore notturne	3
Moderato eccesso di saliva; vi può essere minimo scolo di saliva nelle ore diurne	2
Marcato eccesso di saliva; con scolo di saliva nelle ore diurne	1
Marcato scolo di saliva; vi è necessità continua di asciugarla	0
Q.3 – DEGLUTIZIONE	
Normale	4
Iniziali problemi alimentari; occasionalmente va di traverso	3
Necessità di modificare la consistenza delle diete	2
Necessità di alimentazione enterale supplementare	1
Non in grado di deglutire (alimentazione esclusivamente enterale o parenterale)	0
Q.4 - SCRIVERE A MANO (con la mano già dominante)	
Normale	4
Rallentato o approssimativo; tutte le parole sono leggibili	3
Non tutte le parole sono leggibili	2
In grado di afferrare la penna ma non di scrivere	1
Incapace di afferrare la penna	0
Q.5a - TAGLIARE IL CIBO ED USARE UTENSILI (paziente senza PEG)	
Normale	4
Un po' rallentato e goffo, ma non necessita di aiuto	3
Può tagliare la maggior parte dei cibi, anche se in modo rallentato o goffo; è necessario un certo aiuto	2
Il cibo deve essere tagliato da altri, ma ancora in grado di portarsi il cibo alla bocca da solo, anche se lentamente	1
Deve essere nutrito	0
Q.5b - PREPARARE IL CIBO E USARE UTENSILI (paziente con PEG)	
Normale	4
Maldestro ma in grado di eseguire tutte le manipolazioni da solo	3
Necessario un certo aiuto con dispositivi di fissaggio e chiusura	2
In grado di fornire un minimo di aiuto a chi lo assiste	1
Incapace di eseguire qualsiasi aspetto di questi compiti	0
Q.6 - ABBIGLIAMENTO ED IGIENE PERSONALE	
Normale	4
Bada a se stesso in modo indipendente e completo con sforzo o ridotta efficienza	3
Necessita di assistenza non continuativa e di metodi sostituitivi	2
Necessita di aiuto consistente per la cura di sé	1
Completamente dipendente	0

Q.7 - GIRARSI NEL LETTO E AGGIUSTARSI LE COPERTE	
Normale	4
Un po' rallentato e goffo, ma non necessita di aiuto	3
Può girarsi da solo o sistemare le coperte, ma con grande difficoltà	2
Può iniziare il movimento, ma non girarsi o sistemare le coperte da solo	1
Completamente dipendente	0
Q.8 - CAMMINARE	
Normale	4
Iniziali difficoltà nella deambulazione	3
Cammina con necessità di assistenza (con qualsiasi tipo di ausilio o ortesi)	2
Solo movimenti funzionali che non permettono la deambulazione	1
Nessun movimento utile o finalizzabile degli arti inferiori	0
Q.9 - SALIRE LE SCALE	
Normale	4
Rallentato	3
Lieve instabilità	2
Necessita di assistenza (compreso l'uso del manocorrente)	1
Non può farlo	0
Q.10 – DISPNEA	
Assente	4
Dispnea durante la deambulazione	3
Dispnea nel corso di una o più delle seguenti attività: mangiare, farsi il bagno, vestirsi (ADL)	2
Dispnea a riposo, difficoltà di respirazione anche in posizione seduta o sdraiata	1
Dispnea significativa, deve essere presa in considerazione la ventilazione assistita	0
Q.11 - ORTOPNEA	
Assente	4
Alcune difficoltà durante il sonno notturno per la sensazione di "respiro corto", ma non sono di solito necessari più di due cuscini	3
Sono necessari più di due cuscini per poter dormire	2
Può dormire solo se seduto	1
Impossibilità del sonno notturno per le difficoltà respiratorie	0
Q.12 – INSUFFICIENZA RESPIRATORIA	
Assente	4
Uso intermittente di BIPAP	3
Utilizzo continuo della BIPAP durante la notte	2
Utilizzo continuo della BIPAP durante la notte e il giorno	1
Ventilazione assistita invasiva mediante intubazione o tracheostomia	0

Figura 1.3: Scala di valutazione funzionale revisionata ALSFRS-R.

Figura 1.4: Stadiazione ottenuta con la scala di valutazione funzionale KING.

Table 1 Functional domains and stages

ALSFRS domain	Item	Score	Functional score*
Movement (walking/self-care)†	8	4 Normal	0
	Walking	3 Early ambulation difficulties	
		2 Walks with assistance	
		1 Non-ambulatory functional movement only	1
	OR	0 No purposeful leg movement	
	6	4 Normal function	0
	Dressing and hygiene	3 Independent and complete self-care with effort or decreased efficiency	
		2 Intermittent assistance or substitute methods	
		1 Needs attendant for self-care	1
		0 Total dependence	
Swallowing	3	4 Normal eating habits	0
	Swallowing	3 Early eating problems; occasional choking	
		2 Dietary consistency changes	
		1 Needs supplemental tube feeding	1
		0 NPO (exclusively parenteral or enteral feeding)	
Communicating†	1	4 Normal speech processes	0
	Speech	3 Detectable speech with disturbances	
		2 Intelligible with repeating	
		1 Speech combined with non-vocal communication	1
	AND	0 Loss of useful speech	
	4	4 Normal	0
	Handwriting	3 Slow or sloppy; all words are legible	
		2 Not all words are legible	
		1 Able to grip pen but unable to write	1
		0 Unable to grip pen	
Breathing†	10	4 None	0
	Dyspnea	3 Occurs when walking	
		2 Occurs with one or more of: eating, bathing, dressing	
		1 Occurs at rest, difficulty breathing when either sitting or lying	1
	OR	0 Significant difficulty, considering using mechanical respiratory support	
	12	4 None	0
	Respiratory insufficiency	3 Intermittent use of NIPPV	
		2 Continuous use of NIPPV during the night	1
		1 Continuous use of NIPPV during the night and day	
		0 Invasive mechanical ventilation by intubation or tracheostomy	
ALS-MITOS	Stage	Functional domains lost	
	0	None	
	1	1 domain	
	2	2 domains	
	3	3 domains	
	4	4 domains	
	5	Death	

*Staging determined by the sum of functional score of 1 for each domain.
†Where two items were used, scoring was based on either or both item scores as indicated.
ALFSRS, Amyotrophic Lateral Sclerosis Functional Rating Scale; ALS-MITOS, Amyotrophic Lateral Sclerosis Milano-Torino Staging; NIPPV, nasal intermittent positive pressure ventilation; NPO, nothing by mouth.

Figura 1.5: Conversione dei punteggi da ALSFRS-R a MITOS e definizione degli stage MITOS.

funzionalità. Al di sotto di tale soglia, viene attribuito a quel dominio punteggio 0 (funzionalità mantenuta), al di sopra puteggio 1 (funzionalità compromessa).

La somma dei punteggi relativi alle varie aree permette di ottenere lo score MITOS: un punteggio complessivo di 0 punti indica coinvolgimento funzionale senza perdita di autonomia in alcun dominio, tra 1 e 4 rappresenta il numero di funzioni compromesse, 5 indica la morte del paziente. Per la conversione dei punteggi ALSFRS-R negli stage MITOS si rimanda alla Figura 1.5 [12].

Tale scala di valutazione è stata recentemente validata, dimostrandosi uno strumento efficace per identificare stadi rilevanti della malattia nei pazienti, in accordo col numero delle funzionalità compromesse. Inoltre, MITOS si è dimostrato consistente con la progressione sequenziale della malattia [43].

1.3 Il database PRO-ACT

Dal momento che la prevalenza della SLA è sufficientemente bassa, i trial clinici condotti nel corso degli anni hanno tipicamente riguardato gruppi ristretti di pazienti, con pool massimi intorno ai 1000 soggetti negli studi più ampi [14]. Per mettere a disposizione della comunità scientifica una base di dati quanto più ampia e completa possibile, su cui poter intervenire con analisi solide dal punto di vista della significatività statistica, si è progettato di fondere i dati a disposizione in un unico database.

Il database PRO-ACT (Pooled Resource Open-Access ALS Clinical Trials) [6, 46], creato dunque con questo scopo, raccoglie attualmente i dati di oltre 10700 pazienti che hanno partecipato a 23 trial clinici di fase II/III. I dati sono stati resi anonimi e non tracciabili, rimuovendo informazioni anagrafiche di ciascun soggetto e identificative del trial.

Nel database, per ciascun paziente è presente un codice univoco di identificazione (*SubjectID*) e un certo numero di record, che contengono in modo variabile misure demografiche, valutazioni cliniche ed esiti di esami di laboratorio, registrati durante le visite nel corso dei trial. Il tempo intercorso tra l'inizio dello studio clinico e la data della visita è espresso da una variabile *delta*, con $delta = 0$ in riferimento al primo giorno dello trial. Un eventuale *delta* negativo indica che i dati sono stati acquisiti in una visita precedente all'inizio dello studio. È questo il caso, ad esempio, dei valori riferiti al momento di insorgenza dei primi sintomi (*onset delta*), oppure della diagnosi clinica (*diagnosis delta*).

Si osserva come, per la natura stessa del database, i tempi di campionamento dei vari pazienti siano eterogenei, a seconda del trial di partecipazione, così come il tipo di variabili registrate, la loro numerosità (sono presenti per ciascun paziente alcuni valori mancanti, detti *missing values*) e le loro unità di misure. Queste differenze tra i record richiedono alcune accortezze nel design dell'analisi, con opportuni passi di preprocessing per uniformare i valori e gestire i missing values.

Sul sito di PRO-ACT [4] i record sono disponibili in un unico file, oppure suddivisi a seconda del loro tipo. Si riportano di seguito le categorie di dati disponibili, con una breve descrizione per ciascuna tipologia. Per ulteriori dettagli si rimanda a [4].

- **Demographics:** i dati demografici comprendono età, genere, razza ed etnia, età al momento della comparsa della malattia.
- **Family and Medical History:** in circa il 5% dei casi, la SLA sembra essere causata da un difetto genetico che si presenta in più membri della stessa famiglia, con una trasmissione in genere dominante e raramente recessiva. Si parla in questo caso di "SLA familiare"o "SLA genetica". Per alcuni pazienti sono quindi disponibili informazioni riferite a membri della famiglia e, in certi casi, alla storia medica del paziente stesso.
- **Subject ALS History:** questa sezione contiene informazioni riguardo alla sintomatologia sperimentata dal paziente, che coinvolge la muscolatura volontaria in modo sempre più grave all'aumentare del tempo di osservazione. Viene registrato il sito di esordio della malattia (*Limb, Bulbar, Other, Limb and Bulbar, Spine*) ed i momenti sia di esordio che di diagnosi. Sono presenti, inoltre, le parti del corpo aggredite in seguito dall patologia e i sintomi relativi.
- **Treatment Group:** il tipo di farmaco utilizzato nel trial non è specificato, per evitare che il paziente possa essere ricondotto ad uno studio specifico. È invece registrato se al paziente è stato somministrato il trattamento o il placebo, ed il tempo intercorso tra l'inizio del trial e la somministrazione.
- **Riluzole use:** viene riportato se il paziente ha assunto, durante il trial, il Riluzolo.
- **Concomitant Medication Use:** in questa sezione sono registrati eventuali altri farmaci assunti dal paziente durante il trial. Si tratta di terapie già in corso, integrazioni, o interventi per effetti indesiderati del trial. Tale parte del database è carente di molte informazioni, pertanto è impossibile concludere la mancata assunzione di farmaci concomitanti solo perchè non se ne ha riscontro.
- **Adverse Event:** vengono riportati eventuali eventi avversi registrati nella cartella clinica del paziente durante il trial clinico (contusioni, cefalee, infarti, ecc), che possono essere correlati o meno al trattamento sperimentale in corso. Sono disponibili i *delta* relativi all'insorgenza ed alla conclusione dell'evento e la gravità dello stesso.
- **Symptoms and outcome measures (ALSFRS, FVC, SVC):** la gravità dei sintomi è spesso misurata facendo uso delle scale di valutazione funzionale ALSFRS e ALSFRS-R (vedi Sezione 1.2). Per ogni visita, contrassegnata dal suo *delta*, oltre ai punteggi delle singole domande, viene riportato lo score complessivo ALSFRS(-R). Nel caso in cui il punteggio di una singola domanda non fosse disponibile, nella creazione del database il valore mancante è stato sostutito con un

punteggio opportuno ricavato dai record relativi alle visite precedenti e successive dello stesso soggetto, ottenendo talvolta risultati non interi (es. 2.5, 3.5 punti).
Sono riportati, inoltre, i valori della *Capacità Vitale Forzata* (Forced Vital Capacity, o FCV), un'ulteriore misura di stadiazione, comunemente utilizzata in clinica, corrispondente al volume totale di aria espulsa in un'inspirazione forzata partendo da un'inspirazione massimale. Sono anche presenti i valori della *Capacità Vitale Lenta* (Slow Vital Capacity, o SVC), utilizzata anch'essa come metrica per definire la funzionalità polmonare e corrispondente al massimo volume di aria che può essere espirato lentamente dopo una lenta inspirazione massimale.

- **Vital Signs:** i segni vitali raccolti per ciascun paziente, contrassegnati con il *delta* relativo alla visita specifica, comprendono le misurazioni di pressione sanguigna, pulsazioni, peso, altezza, temperatura corporea e tasso respiratorio. Sono inoltre presenti le relative unità di misura, per le quali si possono osservare standard diversi a seconda del trial.

- **Laboratory Data:** questa sezione contiene i risultati di esami di laboratorio condotti sui pazienti. Per ciascuno, sono indicati nome del test, risultato, unità di misura e *delta* dall'inizio dello studio. Nella creazione del database, si è scelto di riportare i valori anche nel caso in cui questi rientrino nel range di normalità, ipotizzando che possano contribuire comunque all'analisi della condizione globale del paziente. Le unità di misura, nomenclatura ed il tipo di dati registrati sono stati inoltre standardizzati per permetterne un più facile uso. In alcuni casi, si osservano valori assolutamente non fisiologici, dovuti probabilmente ad errori nel riportare i dati, che richiedono un'attenzione specifica al momento dell'utilizzo.
  I test di laboratorio svolti sono: esami delle urine ed esami del sangue per valutare ematocrito, funzionalità epatica e renale, glicemia, livelli ormonali, stato del sistema immunitario, ecc.

- **Death Report:** per i pazienti deceduti durante il corso del trial clinico, è registrato il giorno di morte.

A partire dai dati strutturati contenuti nel database PRO-ACT, è possibile quindi intraprendere analisi che coinvolgano un gran numero di soggetti. Numerosi studi hanno fatto uso di questi dati, ad esempio per predire la progressione della malattia basandosi sugli score ALSFRS e indagare l'esistenza di marcatori prognostici [28], progettare modelli di progressione della malattia [19], ricavare informazioni sulla natura delle relazioni che legano le variabili presenti nel database [18].

In questa tesi, ci si è proposto di estrarre informazioni su rischio di morte e probabilità di sopravvivenza di un subset di soggetti, utilizzando il metodo di classificazione Random Survival Forests, che viene presentato nel prossimo capitolo. A partire dalle informazioni estratte, si sono poi applicati alcuni metodi di clustering (Capitolo 6) per ottenere una stratificazione dei pazienti.

Capitolo 2

Random Survival Forests

Per ottenere informazioni in modo automatico a partire da dati strutturati, si sono sviluppati negli ultimi decenni numerosi metodi di Machine Learning, che permettono di analizzare istanze contenute in database, individuando relazioni e possibili raggruppamenti tra esse. Nel caso in cui i dati da analizzare siano in quantità considerevole (Big Data), l'applicazione degli algoritmi di Machine Learning prende il nome di Data Mining.

Uno degli scopi classici del Machine Learning è quello di effettuare predizione.
Si parte da un pool di dati in cui ogni istanza (o record) è caratterizzata da un certo numero di variabili (o feature) e da una label, che ne descrive la caratteristica di appartenenza ad una classe. Tale dataset viene solitamente suddiviso in due subset, detti *training set* e *test set*, che svolgono due ruoli ben distinti nel processo di classificazione.

1. In una prima fase, detta di apprendimento automatico, vegono presentate all'elaboratore le istanze contenute nel training set, ciascuna caratterizzata da valori specifici delle sue variabili e dalla label (si parlerà in questo caso di classificazione supervisionata). L'elaboratore apprende, mimando i processi cognitivi umani, le relazioni ed il peso delle variabili che portano quella istanza ad essere etichettata dalla propria label.

2. In una seconda fase, si testa l'abilità acquisita dall'elaboratore nel riconoscere le caratteristiche correlate al valore della label. Vengono presentate al classificatore delle nuove istanze, nello specifico quelle contenute nel test set, con valori delle features non precedentemente incontrati ed omettendo il valore della label. Il compito del classificatore, a questo punto, sarà quello di assegnare ad ogni istanza la classe di apparteneza, determinandola attraverso gli schemi da esso auto-appresi.

Talvolta si fa uso di un terzo dataset distinto, detto *validation set*: questo viene utilizzato per verificare ulteriorimente la bontà del classificatore, o, nel caso dell'implementazione di più modelli, per definirne il migliore.

2.1 Alberi Decisionali

Tra le tecniche messe a disposizione dal Machine Learning, l'apprendimento tramite Alberi Decisionali [37] prevede di costruire, a partire dai dati a disposizione, un modello predittivo a grafo. Questo è in grado di classificare le istanze nelle classi indicate dalle label che, contenendo l'informazione sulla quale si vuole fare predizione, prendono comunemente anche il nome di *target*.

L'algoritmo opera su un dataset di istanze, ciascuna descritta da un certo numero di feature e dal target, solitamente suddiviso in due parti: di queste, il subset di training viene utilizzato per allenare il classificatore, mentre il subset di test viene utilizzato in seguito per testarne la bontà.

La costruzione dell'Albero a partire dal training set avviene in maniera automatica, suddividendo a cascata le istanze in sottoinsiemi accomunati da valori concordi delle feature, e omogenei rispetto all'attributo target.

L'algoritmo prende in considerazione ad ogni passo un insieme di feature (dette candidate di *split*) e cerca tra esse la più significativa nel discriminare tra le classi, in termini di feature con il miglior guadagno di informazione normalizzato (*normalized information gain*).
Si riporta di seguito l'algoritmo nel dettaglio.

ALBERI DECISIONALI - ALGORITMO

Sia T l'insieme delle istanze del training set.

1. Si seleziona la feature che meglio differenzia, sulla base delle classi indicate dai target, le istanze contenute in T.
   Tale suddivisione consente di racchiudere in uno stesso sottogruppo quante più istanze possibili caratterizzate dalla stessa classe di appartenenza.
   Si crea un nodo dell'Albero (*root*) il cui valore corrisponde alla feature selezionata.

2. A partire dal nodo root, si crea una coppia di collegamenti (rami), ciascuno dei quali rappresenta un valore/intervallo di valori per la variabile scelta.

3. Ciascun ramo cade in un nuovo nodo.
   Se tutte le istanze del sottogruppo che cade nel nodo appartengono alla stessa classe, si interrompe la suddivisione e il nodo è detto foglia (*leaf*).
   Altrimenti, si procede suddividendo iterativamente le istanze del sottogruppo sulla base delle loro feature, creando nuovi nodi interni e rami, finché non si soddisfa una condizione di stop.

Nella creazione dell'Albero, possono essere posti alcuni vincoli, tra cui la massima profondità di crescita. Questa limita l'algoritmo nell'ulteriore suddivisione in sottogruppi e, di conseguenza, nell'aumento della complessità computazionale oltre i limiti necessari per una buona classificazione. Si parla in questo caso di *pruning* (potatura).

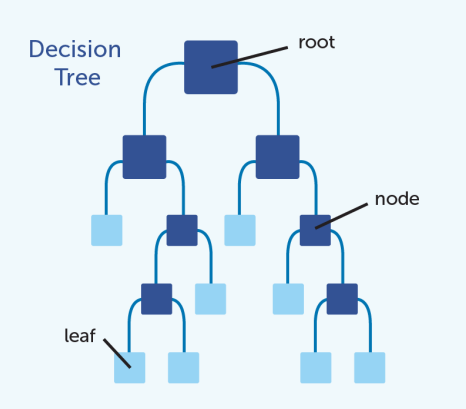

Figura 2.1: Esempio di Albero Decisionale.

Il grafo risultante presenta una forma gerarchica caratteristica (Figura 2.1), che si sviluppa verso il basso scendendo dal nodo superiore (root) verso i nodi estremi (foglie). Ogni nodo interno rappresenta la feature considerata a quel passo per discriminare tra le istanze, gli archi tra i nodi rappresentano il valore/intervallo di valori assunto dalla feature soprastante, i nodi foglia indicano le classi, etichettate coi differenti valori dell'attributo target.

2.1.1 Validazione dell'Albero

Supponendo che i record del training set siano sufficientemente numerosi e descrittivi della popolazione di interesse, tale Albero sarà in grado di classificare in maniera corretta istanze diverse da quelle viste nella fase di allenamento.

Per quantificare la bontà della classificazione, si testa l'Albero su un dataset di cui sono note le label. Si usa a tal fine il test set: ciascuna istanza viene fatta scendere dal nodo root ai nodi foglia dell'Albero, tramite un cammino (path) univocamente determinato dai valori delle sue feature. Tale cammino rappresenta l'insieme di regole che definiscono l'istanza come appartenente ad una determinata classe, rappresentata dal nodo foglia in cui cade.

Dal momento che la vera classe di appartenenza delle istanze del test set è nota, si può determinare la bontà dell'Albero nel classificare i dati, utilizzando opportune metriche tra cui l'*error rate*, definito come la percentuale di classificazioni errate sul totale delle istanze.

Dopo aver testato l'Albero, la sua applicazione viene generalizzata a nuovi dataset in cui la variabile target è mancante, al fine di determinare la classe di appartenenza delle nuove istanze.

Gli Alberi Decisionali presentano come vantaggi il fatto di essere invarianti allo scaling e ad altre trasformazioni dei valori delle feature, la robustezza all'inclusione di eventuali feature irrilevanti e la produzione di modelli ispezionabili. Da contro, forniscono

raramente una classificazione accurata: se un Albero è troppo profondo, infatti, tende a costruire percorsi altamente irregolati, che portano ad un overfitting del training set.

2.2 Random Forests

Nel contesto dei metodi di classificazione supervisionata basati su Alberi, il metodo Random Forests (RF) proposto da Breiman [8] propone di combinare le predizioni ottenute da più Alberi Decisionali (da qui il termine *forests*) tra loro scorrelati, al fine di ottenere una classificazione più robusta.
La Foresta è costruita come segue:

RANDOM FORESTS - ALGORITMO

Per ciascun Albero $b = 1, ..., B$:

1. Siano date T istanze distinte costituenti il training set. Da esse, si selezionano T istanze in maniera casuale, effettuando un campionamento con ripetizione, detto *bagging* (*b*ootstrap *agg*regat*ing*).

2. Il dataset così ottenuto viene quindi utilizzato per costruire l'Albero.
   Siano M le feature di input di ciascuna istanza, ad esclusione di quelle target. Ad ogni nodo, vengono scelte come candidate per lo split $m << M$ variabili in maniera causale tra quelle disponibili. Il valore di m viene mantenuto costante per tutta la crescita della Foresta.

3. Gli Alberi sono cresciuti iterando a ciascun nodo la procedura esposta ai punti precedenti, senza imporre limiti alla loro profondità massima.

Una volta ottenuta la Foresta, vengono fatte scendere lungo ciascun Albero le istanze da classificare, ottenendo per ogni Albero un "voto" di assegnazione ad una classe, specifica per la foglia di arrivo dell'istanza. Al termine della discesa, la Foresta sceglie, per quell'istanza, la classe che ha ottenuto la maggioranza di voti, ovvero la moda delle classi in output agli Alberi Decisionali.

L'utilizzo di una Foresta permette di superare i problemi di overfitting degli Alberi singoli, portando, all'aumentare del numero di Alberi utilizzati, la soluzione a convergenza per la Legge dei Grandi Numeri. Nel caso di feature deboli, ovvero che portano ciascuna poca informazione, inoltre, un singolo Albero porterebbe ad un risultato poco distante dalla scelta casuale di una classe. La combinazione di più Alberi costruiti sfruttando il bagging permette, invece, di aumentare l'accuratezza della classificazione.
Da contro, l'uso di una Foresta di Alberi riduce in parte l'interpretabilità del metodo, che pretende un'approccio più "black box" nel suo utilizzo rispetto agli Alberi Decisionali.

2.2.1 Randomizzazione

La forza dell'algoritmo Random Forests è dovuta ai due passi di randomizzazione introdotti nella costruzione degli Alberi, rispettivamente nel sampling bagging e nella scelta delle variabili di split:

- **SAMPLING BAGGING**: La selezione casuale di T istanze nella costruzione di ciascun Albero, estratte dal training set con ripetizione, permette di produrre una Foresta di Alberi tra loro scorrelati. Tale passo migliora le performance della Foresta in termini di riduzione della varianza del modello, senza aumentarne il bias, rendendo quindi il classificatore meno sensibile al rumore.

  Il bagging produce come secondo effetto la presenza, per ogni Albero, di istanze *out-of-bag* (OOB) non utilizzate per la sua costruzione. Tipicamente, in ogni bagging training set, sono lasciate fuori circa un terzo delle istanze.
  Facendo scendere in ciascun Albero le proprie istanze OOB, si può calcolare l'*errore di predizione* (prediction error) ed avere così un'idea della bontà dello stesso come classificatore. L'uso delle istanze OOB permette di effettuare una stima "interna" dell'accuratezza della predizione, ottenendo risultati comparabili all'uso di un test set di dimensioni analoghe al training set, ma senza la necessità di sfruttare nuovi record indipendenti [7].

- **FEATURE BAGGING**: La casualità nella selezione di m feature come variabili candidate per lo split tra le M disponibili permette di scorrelare ulteriormente gli Alberi. In mancanza di tale passaggio, infatti, una o più feature che risultassero essere forti predittori verrebbero selezionate per lo split in gran parte dei B Alberi, causandone la correlazione.

2.2.2 Parametri utente

I parametri dell'algoritmo su cui l'utente ha arbitrarietà di scelta sono:

- il numero $\boldsymbol{B}$ di Alberi che compongono la Foresta: tipicamente vengono testati valori compresi tra $10^2 - 10^4$, a seconda della natura e della dimensione T del training set. Il numero ottimo di Alberi viene individuato utilizzando meccanismi di Cross-Validation, tenendo conto dell'errore di predizione out-of-bag complessivo della Foresta, calcolato come media degli errori OOB di ciascun Albero.
  Per quanto un numero maggiore di Alberi porti in generale un miglioramento nell'accuratezza della predizione, questo implica anche un'aumento della complessità computazionale. Si è dimostrata l'esistenza di una soglia nel numero di Alberi [34], il cui superamento non risulta conveniente in termini di trade-off tra il miglioramento nella predizione e il carico computazionale;

- il numero $\boldsymbol{m}$ di feature prese in considerazione ad ogni nodo come candidate di split: in un problema di classificazione con M feature, una buona indicazione può essere scegliere $m = \sqrt{M}$ (eventualmente arrotondato per eccesso) variabili candidate in ciascun nodo di ogni Albero.

Anche in questo caso, un'analisi mediante Cross-Validation può individuare il valore di m più adatto per il problema considerato.

È interessante osservare come la profondità dell'Albero, parametro su cui si agisce normalmente nel generare gli Alberi Decisionali, non sia preso in considerazione nell'algoritmo standard delle Random Forests. Infatti, mentre il pruning nel caso di un singolo Albero serve ad evitare problemi di overfitting, nelle Random Forests questo rischio viene intrinsecamente gestito coi passi di randomizzazione.

In letteratura sono comunque presenti alcune variazioni dell'algoritmo originale [16], che prevedono ad esempio l'introduzione di un parametro di pruning e l'uso dell'intero training set per allenare ciascun Albero, o che sostituiscono i bagging training set subset distinti del dataset di training.

2.2.3 Importanza delle variabili

Oltre ad effettuare la classificazione delle istanze, il metodo Random Forests permette di misurare l'importanza delle feature nella classificazione. In determinate applicazioni, come per esempio l'analisi di dati clinici, individuare le feature più significative da misurare nei pazienti per definirne lo stato clinico può essere talvolta proprio il fine principale dell'applicazione dell'algoritmo.

Il contributo delle singole variabili all'accuratezza della predizione può essere ricavato tramite un'operazione di permutazione: dopo aver costruito la Foresta di Alberi, inizialmente si calcola l'errore di predizione in modo standard, classificando le istanze OOB di ogni Albero. In seguito, si effettua una permutazione dei valori della i-sima feature per le stesse istanze OOB. Le istanze con i valori permutati vengono quindi nuovamente proposte alla Foresta, registrando il nuovo errore di predizione.
L'importanza della singola variabile nella classificazione è tanto maggiore, quanto più la sua permutazione provoca un aumento dell'errore di predizione.

2.3 Analisi Survival

In statistica esiste una branca, denominata *Survival Analysis*, che si occupa dello sviluppo di metodi in grado di analizzare l'aspettativa temporale prima che uno o più eventi si verifichino. Ad esempio, la Survival Analysis può applicarsi in ambito clinico con il fine di studiare e modellare l'aspettativa di sopravvivenza di un paziente. I dati a disposizione, detti *dati survival*, sono registrati a partire da un evento iniziale, e raccolti nell'attesa che l'evento di interesse (*target event*) si verifichi.

Può accadere che, durante il tempo di osservazione, l'evento di interesse non venga registrato per tutte le istanze. Si pensi, ad esempio, alla registrazione di dati relativi a pazienti facenti parte di un trial clinico: l'evento iniziale è l'accesso del paziente al trial, l'evento di interesse può esserne la morte, che si verifica in un istante denominato *failure time*. A causa di limiti temporali nella conduzione del trial, può accadere che il follow-up di alcuni pazienti cessi prima che si sia osservato l'evento di interesse. Si parla, in questi casi, di istanze *censurate temporalmente*. Poichè l'istante temporale in cui si

registra l'evento di interesse è incognito, ma sicuramente successivo all'ultimo istante registrato, si parla di censura destra (ovvero dati *right-censored*). Tali istanze, seppur incomplete, portano in sè informazioni sulla sopravvivenza dell'oggetto di studio fino all'ultimo istante di misurazione, che è noto e registrato.

In casi come quello descritto sopra, l'Analisi Survival si occupa di gestire l'utilizzo dei dati right-censored, indagando come il failure time sia correlato agli attributi registrati nei soggetti ai vari istanti di misurazione.

2.4 Random Survival Forests

È presente in letteratura [25] un'estensione del metodo Random Forests all'Analisi Survival, detto Random Survival Forests (RSF), che permette l'analisi di dati survival right-censored.

Nel caso specifico di analisi di sopravvivenza di soggetti, i dati in input sono strutturati includendo, per ogni istanza, sia le variabili raccolte in successivi istanti temporali e utilizzate per la predizione (contenute in un vettore $\mathbf{x}$), sia due feature contenenti le informazioni survival del soggetto. In particolare, per descrivere la condizione del soggetto rispetto all'evento target di interesse, si riportano l'ultimo istante di registrazione per il soggetto (*time event*) e la variabile di stato (*status*) ad esso associata, che assume valore **1** se a quel time event si è verificato l'evento morte (failure event), oppure **0** se il soggetto è stato right-censored.

Ciò che si ottiene con RSF è un modello di predizione del rischio che il failure event atteso si verifichi.

Nella crescita della Foresta, come nel metodo originale di Breiman, vengono costruiti Alberi Decisionali binari tenendo conto, ai vari nodi, del valore delle informazioni survival: nelle RSF, quindi, i criteri di splitting non massimizzano più la divisione in classi, bensì raggruppano le istanze sulla base delle loro informazioni survival.

Questo metodo, di cui si illustrano i dettagli nel seguito, è implementato nel pacchetto R `randomForestSRC` (recente sostituzione del precedente `randomSurvivalForest`).

RANDOM FORESTS SURVIVAL - ALGORITMO

Per ciascun Albero $b = 1, ..., B$:

1. Siano date T istanze distinte costituenti il training set. Da esse, si selezionano T istanze in maniera casuale, ottenute come campionamento con ripetizione. Il bagging esclude in media il 37% dei dati, che costituiscono le istanze out-of-bag (OOB) per l'Albero corrente.

2. Il dataset così ottenuto viene utilizzato per costruire l'Albero.
   Siano M le feature di input di ciascuna istanza, ad esclusione di quelle target. Ad ogni nodo, vengono scelte come candidate per lo split $m << M$ variabili in maniera

casuale tra quelle disponibili. Il valore di m viene mantenuto costante per tutta la crescita della Foresta.
Lo splitting viene effettuato secondo un opportuno criterio, massimizzando la differenza survival tra i due nodi figli. Operativamente, questo significa che lo split ottimo coinvolge una feature ed un suo valore soglia tali da permettere la suddivisione delle istanze in due gruppi: dal punto di vista delle informazioni target, ciascun gruppo è al proprio interno omogeneo e i gruppi sono tra loro quanto più dissimili possibile.

3. Gli Alberi sono cresciuti senza imporre limiti alla loro profondità massima, costituendo iterativamente nuovi nodi.
L'unico vincolo nella costruzione coinvolge, ancora una volta, le informazioni survival: ciascun nodo foglia deve contenere almeno un'istanza contrassegnata dallo status "1"(morte).

Dopo aver costruito la Foresta, si fanno scorrere le istanze da classificare lungo gli Alberi, mostrando al classificatore solo le feature raccolte durante la storia medica del paziente, e non le informazioni survival.

A seconda della foglia in cui cade l'istanza, ciascun Albero è in grado di fornire la sua predizione, che in questo caso non sarà la stima della classe di appartenenza (come era per RF), ma una quantità legata al rischio di morte, detta *Cumulative Hazard Function* (CHF), definita nella Sezione 2.4.2.

Come nel caso delle RF, la predizione complessiva della Foresta viene calcolata come la media delle predizioni dei singoli Alberi, ovvero per RSF coincide con la media dei CHF in output agli Alberi. Questa grandezza è denominata *Ensemble CHF* e viene illustrata nella Sezione 2.4.3.

2.4.1 Parametri utente

Come nel caso di RF, l'utente ha la possibilità di intervenire su alcuni parametri nella creazione della Foresta:

- il numero **B** di Alberi che compongono la Foresta ha un peso sia dal punto di vista dell'accuratezza della predizione, sia dal punto di vista di complessità computazionale. Esso viene, anche in questo caso, indagato tramite metodi di Cross-Validation, secondo le modalità approfondite nel Capitolo 4;

- il numero **m** di feature considerate come candidate di split a ciascun nodo può essere scelto in prima istanza pari a $\sqrt{M}$, eventualmente arrotondato per eccesso, con M numero di feature predittive per ciascun soggetto. Anche in questo caso, tramite Cross-Validation si può individuare il valore ottimo per la Foresta corrente;

- è presente per RSF un parametro, detto **nodesize**, che definisce il numero minimo di istanze distinte che devono cadere in un nodo terminale; esso funge di parametro

di pruning. Il default nell'analisi survival è $nodesize = 3$, ma, ancora una volta, la Cross-Validation può aiutare a determinare il valore più adatto;

- l'utente ha, infine, arbitrarietà sul **criterio di splitting** da utilizzare nella creazione dei nuovi nodi. Per maggiori dettagli sui criteri di splitting per RSF, si rimanda alla Sezione 2.4.5;
- esistono altri parametri su cui l'utente ha la possibilità di intervenire. Per i dettagli si rimanda alla documentazione del pacchetto `randomForestSRC` [24], e alla loro discussione nel Capitolo 4.

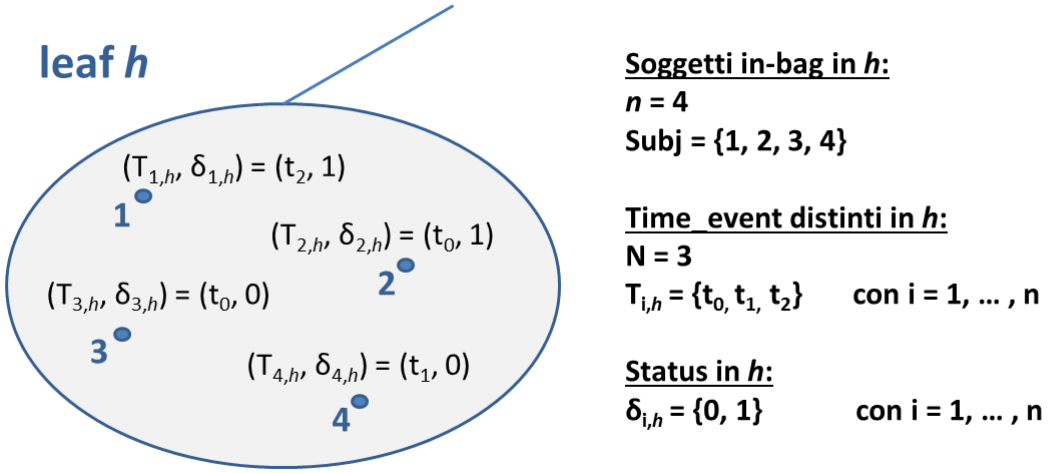

Figura 2.2: Esempio di nodo terminale di un Albero di RSF (foglia h).

2.4.2 Cumulative Hazard Function (CHF)

Si definisce in questa sezione il concetto di Cumulative Hazard Function.

Sia h un nodo terminale di un Albero della Foresta (vedi esempio in Figura 2.2) e siano in numero n le istanze in-bag che, nel processo di costruzione dell'Albero, cadono in esso.

Di queste, l'istanza i-esima ($i = 1, ..., n$) è caratterizzata dalle proprie feature contenute nel vettore $\mathbf{x}_i$, e dalle proprie informazioni survival, descritte come coppia $(T_{i,h}, \delta_{i,h})$, dove T indica il time event e δ lo stato.

I time event per la foglia h possono assumere N valori distinti $T_{i,h} = \{t_0, ..., t_N\}$, mentre lo stato assume valori $\delta_{i,h} = \{1$ se morto, 0 se censored$\}$.

Preso un tempo $t_{l,h}$, si definiscono le due quantità $d_{l,h}$ e $Y_{l,h}$ rispettivamente come il numero di morti ed il numero degli individui a rischio nella foglia h fino a quell'istante. Si

definiscono a rischio quei soggetti ancora vivi all'istante precedente a quello considerato (ovvero per i quali non si è ancora registrata informazione né di morte né di censura), che potrebbero quindi morire nell'istante corrente.

La stima CHF per la foglia h viene definita come lo stimatore di Nelson-Aalen:

$$CHF_h = \hat{H}_h(t) = \sum_{t_{l,h} \leq t} \frac{d_{l,h}}{Y_{l,h}} \tag{2.1}$$

Ciascun soggetto i, inserito nell'Albero, segue un cammino caratteristico a seconda dei valori delle sue variabili predittive $\mathbf{x}_i$ (sono escluse le informazioni survival). La natura binaria dell'Albero garantisce l'univocità del nodo terminale di caduta.

La predizione RSF dell'Albero per il soggetto i, indicata con $H(t|\mathbf{x}_i)$, coincide con la stima CHF del nodo foglia di arrivo:

$$H(t|\mathbf{x}_i) = \hat{H}_h(t) \tag{2.2}$$

Si riporta di seguito, a titolo esemplificativo, la computazione del CHF nel caso specifico della foglia in Figura 2.2.

Per ogni t_i, si calcola d come il numero di soggetti di cui si è registrata la morte esattamente in quell'istante, mentre Y corrisponde al numero di soggetti che fino ad allora risultavano vivi.

Per la foglia h di Figura 2.2 si compie quindi la seguente analisi:

- $t_l < t_0$: $CHF(t < t_0) = 0$
- $t_l = t_0$: censored = {soggetto 3}
  $d_{0,h} = \{\text{soggetto 2}\} = 1$
  $Y_{0,h} = \{\text{soggetti 1, 2, 3, 4}\} = 4$
  $CHF(t_0 \leq t < t_1) = \frac{d_{0,h}}{Y_{0,h}} = \frac{1}{4}$
- $t_l = t_1$: censored = {soggetto 4}
  $d_{1,h} = 0$
  $Y_{1,h} = \{\text{soggetti 1, 4}\} = 2$
  $CHF(t_1 \leq t < t_2) = \frac{d_{0,h}}{Y_{0,h}} + \frac{d_{1,h}}{Y_{1,h}} = \frac{1}{4} + \frac{0}{4} = \frac{1}{4}$
- $t_l = t_2$: censored = 0
  $d_{2,h} = \{\text{soggetto 1}\} = 1$
  $Y_{2,h} = \{\text{soggetto 1}\} = 1$
  $CHF(t \geq t_2) = \frac{d_{0,h}}{Y_{0,h}} + \frac{d_{1,h}}{Y_{1,h}} + \frac{d_{2,h}}{Y_{2,h}} = \frac{1}{4} + \frac{1}{4} + \frac{1}{1} = \frac{6}{4}$

Si osserva che:

(1) fintanto che non si acquisicono informazioni sulla morte di alcun soggetto, il CHF si mantiene costante e pari a zero;

(2) quando, ad un dato istante temporale, si verifica un evento morte o censura, Y si decrementa di una quantità pari al numero di soggetti persi in quell'istante;

(3) il rischio puntuale nel caso di soggetto censurato è pari a zero, dal momento che il numeratore resta nullo (nessun nuovo evento morte)

(4) di conseguenza, in caso di solo evento censored ad dato un istante temporale, il CHF complessivo rimane costante.

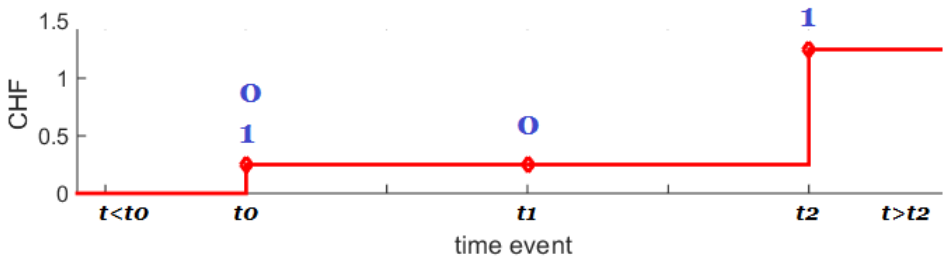

Figura 2.3: Calcolo del CHF ottenuto per la foglia h. In blu sono riportati gli stati 0 (censored) e 1 (morte) dei soggetti ai vari time event.

Si riscontra (vedi Figura 2.3) come la stima del CHF sia una funzione a gradini monotona crescente: in particolare, il CHF resta costante in corrispondenza di eventi censored e aumenta quando si osserva un evento morte.

Si noti che lo stimatore di Nelson-Aalen gode della proprietà di Conservazione degli Eventi [33]: ciò significa che la somma dei CHF su tutti i tempi t_i (sia di censura che di morte) è pari al numero totale di morti registrate. Nel caso della foglia di Figura 2.2, la verifica è immediata.

2.4.3 Ensemble CHF

La predizione complessiva del classificatore RSF per una nuova istanza i è la media dei CHF ottenuti dalla discesa lungo i singoli Alberi. Tale quantità prende il nome di *Ensemble CHF* e viene indicata con $H_e(t|\mathbf{x}_i)$.

Nel caso specifico di un'istanza i appartenente al training set ed utilizzata quindi nella costruzione della Foresta, si distinguono due tipi di Ensemble CHF, a seconda che si considerino solo gli Alberi della Foresta per cui i è OOB, oppure tutti:

- **OOB Ensemble CHF**
  Si prendono in considerazione solo gli Alberi in cui il soggetto i-esimo non è stato selezionato nel bagging per la costruzione, ovvero quelli per cui risulta OOB.

  Con $H_b^*(t|\mathbf{x}_i)$ si indica il CHF ottenuto facendo scendere l'istanza i lungo il generico Albero b-esimo. Si assegna quindi un flag $I_{i,b} = 1$ agli Alberi $b = 1, ..., B$ per cui i è OOB ed un flag $I_{i,b} = 0$ agli Alberi $b = 1, ..., B$ per cui i è in-bag.

  Si definisce allora l'OOB Ensemble CHF, indicato con $H_e^{**}(t|\mathbf{x}_i)$ come:

$$H_e^{**}(t|\mathbf{x}_i) = \frac{\sum_{b=1}^{B} I_{i,b} H_b^*(t|\mathbf{x}_i)}{\sum_{b=1}^{B} I_{i,b}}, \tag{2.3}$$

  ovvero la somma dei CHF dei soli Alberi in cui i è OOB, diviso il numero di tali Alberi.

- **Bootstrap Ensemble CHF**
  Se si considera invece tutta la Foresta senza distinzioni legate al bagging, si possono omettere i flag e calcolare il CHF complessivo come media dei CHF risultanti dalla discesa dell'istanza i lungo tutti gli Alberi:

$$H_e^*(t|\mathbf{x}_i) = \frac{1}{B} \sum_{b=1}^{B} H_b^*(t|\mathbf{x}_i). \tag{2.4}$$

2.4.4 Ensemble Mortality

A partire dalle definizioni di Ensemble CHF e sfruttando la proprietà di Conservazione degli Eventi, si può definire un ulteriore output della Foresta RSF, detta *Ensemble Mortality* ed indicata con $M_{e,i}$.

Dato un soggetto descritto dal suo vettore delle feature $\mathbf{x}_i$, la Ensemble Mortality rappresenta il numero totali di morti attese, sotto l'ipotesi nulla di comportamento simile (in termini survival) per soggetti che cadono nello stesso nodo.

Operativamente, la Ensemble Mortality è calcolata come la somma degli Ensemble CHF H_e ottenuti per quel soggetto in tutti e soli gli n time event di morte T_j del dataset.

Dal momento che l'Ensemble CHF può essere ottenuto considerando solo gli Alberi per cui il soggetto i è OOB, oppure utilizzando tutta la Foresta (vedi Sezione 2.4.3), si avranno corrispondentemente due definizioni di Ensemble Mortality, entrambe condizionate dallo specifico vettore $\mathbf{x}_i$ e calcolate sui tempi T_j:

- **OOB Ensemble Mortality**
  Ricavata dall'OOB Ensemble CHF $H_e^{**}(t|\mathbf{x}_i)$, è definita come:

$$\hat{M}_{e,i}^{**} = \sum_{j=1}^{n} H_e^{**}(T_j|\mathbf{x}_i) \tag{2.5}$$

- **Bootstrap Ensemble Mortality**
  Ricavata dal Bootstrap Ensemble CHF $H_e^*(t|\mathbf{x}_i)$, è definita come:

$$\hat{M}_{e,i}^* = \sum_{j=1}^{n} H_e^*(T_j|\mathbf{x}_i) \tag{2.6}$$

2.4.5 Criteri di splitting

Si è fatto cenno, nell'algoritmo di costruzione della Foresta RSF, della necessità di utilizzare un opportuno criterio di splitting nella creazione di nuovi nodi. Questo deve tenere conto delle informazioni survival delle istanze e massimizzare la suddivisione in sottogruppi simili dal punto di vista di time event e status.

Nel caso di analisi survival, i criteri di splitting proposti [24] sono:

- **log-rank**: default per l'analisi survival, divide le istanze massimizzando il test dei ranghi logaritmici (log-rank test statistic) [41, 29];
- **log-rank score**: suddivide le istanze utilizzando una statistica log-rank standardizzata [22].

2.4.6 Errore di predizione

Per stimare la bontà del classificatore RSF, si calcola una stima dell'Errore di Predizione (Prediction Error, indicato talvolta con *PE*) utilizzando l'Indice di Concordanza di Harrell (*C-index*) [20].

Il C-index è una metrica correlata all'area sotto la curva ROC [21], che misura la bontà del modello nel predire diverse condizioni. Nel dettaglio, esso stima la probabilità che, presa in maniera casuale una coppia di soggetti, quello tra i due a morire per primo avesse effettivamente un rischio di morte predetto più alto (ovvero un outcome predetto peggiore).

Il C-index presenta come principale vantaggio la non dipendenza da un singolo istante temporale per la sua valutazione. La sua bontà è stata inoltre validata negli scenari con istanze censored.

CALCOLO DEL C-INDEX

Il C-index è calcolato secondo i seguenti passi:

1. Si formano tutte le possibili coppie di istanze a partire dai dati.
2. Si omettono tutte quelle coppie i e j (con $T_i < T_j$) per cui il time event inferiore T_i è associato ad uno stato censored: risultano infatti impossibili da confrontare da T_i in avanti, dal momento che lo stato del soggetto censored i da quell'istante in poi risulta incognito.

Si omettono, inoltre, le coppie in cui le istanze i e j hanno time event coincidenti $T_i = T_j$ e sono entrambe associate a stati censored.
Si indicano con *Permissible* (ammissibili) le coppie rimaste dopo questa selezione.

3. Per ogni coppia Permissibile con $T_i \neq T_j$ (si ipotizza, senza perdita di generalità, $T_i < T_j$) si conta:

   - 1, se il soggetto i, con minor survival time, presenta un predicted outcome peggiore, ovvero un indice legato al rischio più elevato (predizione corretta: soggetto i indicato come più a rischio),
   - 0.5, se il predicted outcome dei due soggetti coincide (predizione sbagliata, ma non invertita).

   Per ogni coppia Permissibile con $T_i = T_j$ ed in cui entrambi gli eventi registrati sono morte, si conta:

   - 1, se il predicted outcome è uguale (predizione corretta),
   - 0.5, se il predicted outcome è diverso per i due soggetti (predizione sbagliata).

   Per ogni coppia Permissibile con $T_i = T_j$, in cui un soggetto ha stato censored e l'altro risulta morto (si ricorda che l'opzione entrambi censored è stata omessa al punto 2), si conta:

   - 1, se il predicted outcome per il soggetto morto è maggiore (predizione corretta),
   - 0.5, se il predicted outcome maggiore è stato predetto per il soggetto censored (predizione sbagliata).

   La somma dei punteggi ottenuti su tutte le coppie Permissibile viene denominata *Concordance* (concordanza).

4. Il C-index è infine definito come:

$$C = \frac{Concordance}{Permissibile} \tag{2.7}$$

Il C-index è quindi una quantità sempre positiva, compresa tra 0 e 1, coi valori estremi assunti se tutte le predizioni sono sbagliate, o se sono tutte corrette, rispettivamente.

Un valore di $C = 0.5$ corrisponde a nessuna capacità predittiva (fornisce un risultato non migliore dell'assegnazione casuale), $C > 0.5$ indica una capacità di predizione da parte del modello, e $C < 0.5$ indica una antipredizione, nel senso che si ottiene una predizione è peggiore dell'assegnazione casuale, ma invertendo la direzione della predizione si ottiene un buon modello.

OOB Prediction Error

L'errore di predizione viene calcolato come il complemento a uno del C-index:

$$PE = 1 - C. \tag{2.8}$$

Si descrive di seguito la procedura di calcolo dell'OOB Prediction Error, ricavato a partire dal C-index sui soggetti OOB di tutti gli Alberi della Foresta.

Si osserva come la quantità Concordance sia legata alle predizioni, mentre Permissibile dipenda unicamente dalla natura dei dati. Per calcolare il C-index, quindi, è necessario avere per ciascuna istanza una predizione del rischio data da quella Foresta.

Si fa uso a tal proposito dell'OOB Ensemble CHF H_e^{**}, che si ricorda essere ottenuto, per il generico soggetto i-esimo, come media delle predizioni CHF di tutti gli Alberi in cui i non è stato sfruttato per la loro costruzione.

A partire dall'OOB Ensemble CHF, si calcola, similmente a quanto fatto per la computazione dell'OOB Ensemble Mortality (vedi Sezione 2.4.4), un predicted outcome cumulativo da utilizzare per il calcolo della Concordance.

Si selezionano m tempi distinti $t_1^o, ..., t_m^o$, ad esempio coincidenti con i time event $\{t_1, ..., t_N\}$.

Nel confronto tra due soggetti i e j, si afferma che il soggetto i ha predicted outcome peggiore se:

$$\sum_{l=1}^{m} H_e^{**}(t_l^o|\mathbf{x}_i) > \sum_{l=1}^{m} H_e^{**}(t_l^o|\mathbf{x}_j), \tag{2.9}$$

ovvero se la somma del suo OOB Ensemble CHF valutato negli istanti selezionati è maggiore di quella del soggetto j, o, ancora in altre parole, se il rischio di morte per il soggetto i risulta maggiore di quello del soggetto j, considerato questo campionamento temporale di H_e^{**}.

A partire da questo OOB predicted outcome, è quindi possibile calcolare il valore di Concordance e ricavare, quindi, l'OOB C-index. Per uniformità di notazione con l'OOB Ensemble CHF H_e^{**}, questo viene indicato con C^{**}.

Si ottiene, infine, l'OOB Prediction Error, definito come:

$$PE^{**} = 1 - C^{**}. \tag{2.10}$$

PE^{**} è a sua volta una quantità compresa tra 0 e 1, per la quale valgono considerazioni complementari a quelle fatte per i valori del C-index.

2.4.7 Importanza delle variabili (VIMP)

Il metodo RSF permette di ottenere in output una stima dell'importanza delle feature nel predire correttamente il rischio di morte del soggetto.

In particolare, a ciascuna variabile viene associata una grandezza, detta *VIMP* (**V**ariabile **Imp**ortance), proporzionale al contributo della feature. Intuitivamente, l'importanza della variabile sarà tanto maggiore, quanto più una sua perturbazione provoca un peggioramento nella predizione.

Nell'implementazione del pacchetto R `randomForestSRC` [24] sono indicate diverse possibilità di calcolo della VIMP. Si illustrano di seguito le principali.

Sia x una delle feature del vettore $\mathbf{x}$ associato ai soggetti.

- Il default calcola la VIMP mantenendo la modalità proposta da Breiman per le RF [8].
  Per ciascun Albero, si selezionano le sue istanze OOB:

  1. Queste vengono fatte scendere lungo l'Albero, ottenendo come predizione, per ciascuna, l'OOB CHF della sua foglia terminale. A partire da questa (vedi Sezione 2.4.6), si calcola il Concordance, quindi il C-index OOB e, infine, si registra l'OOB Prediction Error (PE) "originale" per quell'istanza in quell'Albero.
  2. In seguito, si permutano i valori di x all'interno del gruppo di istanze OOB considerato. Le istanze così ottenute vengono nuovamente fatte scendere lungo l'Albero, ottenendo la predizione e ricavando, da essa, l'OOB Prediction Error (PE) "permutato".
  3. La VIMP per x viene definita come la differenza tra l'OOB Prediction Error permutato e l'OOB Prediction Error originale, calcolata per ogni Albero e mediata su tutti:

$$VIMP = \frac{1}{B}\sum_{b=1}^{B}(\{OOB\ PE_{permutato}\}_b - \{OOB\ PE_{originale}\}_b) \qquad (2.11)$$

- Il metodo proposto da Ishwaran *et al.* [25, 24] nella definizione del metodo RSF si differenzia leggermente dal precedente.
  Per ciascun Albero, si selezionano le sue istanze OOB:

  1. Come nel caso precedente, si considerano le istanze OOB orginali per ciascun Albero. Ciascuna viene fatta scendere lungo il proprio Albero, ottenendo, dalla foglia di arrivo, come predizione il rispettivo OOB CHF.
     Rispetto al metodo precedente, l'operazione di media viene svolta direttamente a questo punto: per ogni istanza, vengono mediati gli OOB CHF ottenuti negli Alberi in cui l'istanza è OOB, ricavando l'OOB Ensemble CHF. A partire da questo, si calcola l'OOB Prediction Error (PE) "originale" medio.
  2. L'OOB Prediction Error (PE) "perturbato" (in questo caso non permutato) medio viene ottenuto facendo scendere sempre le istanze originali OOB lungo gli Alberi. In questo caso, infatti, il passo di perturbazione viene applicato direttamente ai nodi in cui lo split è costruito sulla variabile x di cui si vuole calcolare la VIMP.
     Sono possibili, nello specifico, tre modalità di perturbazione:

(i) *permutazione al nodo:* il valore di x per l'istanza corrente viene ottenuto dalla permutazione dei valori di x delle sole istanze OOB che, nella loro discesa, passano per quello stesso nodo;

(ii) *assegnazione casuale*: l'istanza che sta percorrendo l'Albero viene assegnata ad uno dei due nodi figli in maniera casuale, e non sulla base del valore di x rispetto alla soglia di split del nodo;

(iii) *antisplit*: l'istanza viene deviata lungo il ramo opposto rispetto a quello in cui scenderebbe dato il suo valore di x.

Dopo aver fatto scorrere le istanze OOB lungo gli Alberi, sottoponendole alla procedura di permutazione scelta, si procede come di consueto, calcolando il CHF per ogni Albero. Gli OOB CHF, mediati sui B Alberi, forniscono l'OOB Ensemble CHF medio. Questo viene sfruttato come predizione ed utilizzato nel calcolo dell'OOB Prediction Error "perturbato" medio.

3. La VIMP per x viene definita come la differenza tra l'OOB Prediction Error perturbato e l'OOB Prediction Error originale, entrambi già mediati su tutti gli Alberi:

$$VIMP = \{OOB\ PE_{permutato}\}_{mean} - \{OOB\ PE_{originale}\}_{mean} \tag{2.12}$$

Valori fortemente positivi della VIMP indicano che la variabile x è predittiva, mentre valori nulli identificano variabili non predittive. È interessante osservare come la VIMP possa assumere anche valori negativi: in questo caso, si può dire solo il rumore in x risulta più informativo dei valori originali della variabile.

Si osserva come spesso, nelle RF e nelle RSF, le variabili scelte per lo split nei nodi vicini alla radice apportino un gran contributo all'accuratezza della predizione [42]. Nel momento in cui si interviene su di esse con una permutazione, quando le istanze OOB permutate vengono fatte scendere lungo gli Alberi, la predizione è significativamente peggiore. Ne risulta quindi un OOB Precition Error permuted maggiore e, di conseguenza, valori della VIMP molto elevati.

Per feature utilizzate per lo split in nodi verso la parte terminale dell'Albero, da contro, si osserva un contributo inferiore alla bontà della predizione. A mano a mano che si scende lungo l'Albero, infatti, le istanze considerate ad ogni nodo sono in numero sempre minore ed una eventuale aggiunta di rumore influenza in maniera meno significativa la predizione [23].

Le variabili con bassa VIMP, poiché poco predittive, possono essere omesse dal vettore $\mathbf{x}$ in input alla Foresta, senza che questo modifichi sensibilmente la bontà della predizione del classificatore.

2.4.8 Confronto di RSF con altri metodi survival

I dati survival vengono solitamente analizzati utilizzando metodi che richiedono assunzioni molto strette come, ad esempio, la proporzionalità del rischio. Inoltre, dal momento

che tali metodi sono spesso parametrici, eventuali effetti non lineari delle variabili devono essere modellizzati tramite trasformazioni o espansioni, includendo in questo caso le funzioni di base nella matrice delle feature. In aggiunta, per determinare l'eventuale esistenza di effetti non lineari talvolta sono necessari approcci *ad hoc* (come ad esempio la Stepwise Regression). Anche l'identificazione di interazioni tra feature, specialmente nel caso in cui siano coinvolte variabili multiple, può rivelarsi problematico e deve essere svolta attraverso algoritmi a *forza bruta*, esaminando tutte le interazioni tra gruppi variabili, o sfruttando conoscenze a priori, non sempre disponibili.

Tutte queste necessità sono invece gestite automaticamente dalle Random Forests Survival, come dimostrato in [25].

Capitolo 3

Dati e preprocessing

3.1 I dati in analisi

Per l'analisi svolta in questo lavoro di tesi, si è considerato un subset di pazienti estratto dal database PRO-ACT. In particolare, si è utilizzato il pool di soggetti proposto per la challenge scientifica *Dream7: Phil Bowen ALS Prediction Prize4Life* [3].

DREAM7 fa parte delle sfide scientifiche lanciate sulla piattaforma *DREAM (Dialogue on Reverse Engineering Assessment and Methods) Challenges* [2]: periodicamente, vengono proposti alla comunità scientifica degli argomenti di ricerca, mettendo a disposizione grandi moli di dati biologici e biomedici open-source e proponendo l'oggetto dell'analisi, con l'obiettivo di accelerare la ricerca in quell'ambito. Come incentivo, viene assegnato un premio in denaro al team che apporta il miglior contributo. Nel 2012, una delle sfide ha appunto riguardato dati di pazienti SLA, utilizzando una prima versione del database PRO-ACT, che all'epoca non era ancora stato reso pubblico.

La scelta di utilizzare questa versione di PRO-ACT per l'analisi svolta in questa tesi è stata motivata dalla possibilità di sfruttare le modalità di preprocessing proposte dai team partecipanti alla sfida [28]. In futuro, si punta ad estendere gli studi di stratificazione proposti in questo lavoro alla versione aggiornata del database PRO-ACT, nonché ad altri database a disposizione delle varie realtà cliniche e scientifiche (per esempio registri ospedalieri o regionali di pazienti SLA).

L'analisi ha riguardato un subset di 1822 pazienti. I dati, meno standardizzati e non completi rispetto alla versione definitiva di PRO-ACT, sono contenuti in tre file, proposti dall'organizzazione della Challenge rispettivamente come *training set* (918 soggetti), *test set* (279 soggetti) e *validation set* (625 soggetti).

Si osserva come, nell'applicazione delle tecniche di classificazione, si sia soliti avere a disposizione un unico dataset. Questo viene usualmente suddiviso in due parti, una utilizzata per il training e l'altra per il test: su di esse, il classificatore viene costruito, testato e affinato. Dal momento che le istanze di training e test provengono dallo stesso dataset, fintanto che il classificatore si basa unicamente su di esse ci si aspettano performance comparabili per le due suddivisioni. Nell'eventualità in cui sia disponibile un secondo dataset, indipendente dal primo, il classificatore effettua invece predizione

su istanze meno omogenee, dal punto di vista dei valori delle feature, rispetto a quelle incontrate fino a quel momento. Il dataset di validazione viene utilizzato appositamente per testare l'abilità di generalizzazione del classificatore e misurarne le performance.

Nel caso specifico della Challenge, tuttavia, da documentazione [28] i dataset di test e validazione non risultano differire *a priori* nella complessità di classificazione delle istanze in essi contenuti.

Durante la Challenge, il dataset di training è stato fornito ai team partecipanti fin dall'inizio, per la fase di costruzione ed apprendimento del classificatore, con lo scopo, basandosi sui dati ALSFRS dei pazienti raccolti nei primi tre mesi, di predire la progressione della malattia nei nove mesi successivi.
La performance del classificatore così ottenuto poteva quindi essere testata *in itinere*, applicandolo alle istanze contenute nel test set. Basandosi sulle metriche relative alla bontà della predizione, il codice poteva essere ricorsivamente migliorato e il classificatore affinato nelle sue capacità. Una tale procedura rischia, tuttavia, di portare alla costruzione di un classificatore basato sulle particolari istanze del test set (una sorta di overfitting sul test set, vedi Supplementary di [28]).

Il validation set, nascosto ai partecipanti fino alla consegna definitiva del codice, è stato quindi utilizzato per verificare la capacità del classificatore nel predire su istanze mai viste e, quindi, al sicuro da fenomeni di overfitting.

In Figura 3.1 è riportato un esempio dei dati forniti. Ogni record riguarda una variabile (detta anche attributo, o feature) riferita ad un soggetto: sono riportati ID del soggetto, il tipo di dato, il tipo di feature ed il valore della feature. Le colonne rimanenti contengono i codici relativi ai tipi di dati adiacenti. Si osserva quindi come, per ciascun soggetto, siano presenti più righe, pari al numero di variabili misurate durante tutta la durata del trial.

Tali variabili possono essere suddivise in:

- **Feature statiche:** registrate solamente nel corso della prima visita ($delta = 0$) in quanto univoche per ciascun paziente, riguardano le informazioni demografiche, l'anamnesi familiare e personale e il tipo di trattamento a cui è stato sottoposto il paziente;

- **Feature dinamiche:** misurate ad ogni visita, comprendono gli score ALSFRS(-R) singoli e totale, le misure FVC e SVC, gli esami di laboratorio e le valutazioni dei segni vitali. Per ciascuna visita, viene riportato il *delta* corrispondente.

Si osserva (1) etereogeneità nei dati a disposizione, non solo per quanto riguarda le tipologie di feature misurate, ma anche per i tempi di campionamento delle stesse, in termini sia di distanza tra le visite, sia di tempo complessivo trascorso dal soggetto all'interno del trial.

Inoltre, si osserva che (2) per taluni pazienti non sono state registrate alcune feature (*missing value*), casi che richiedono di essere gestiti con opportune procedure di imputazione, e che (3) il dataset fornito può non contenere informazioni sullo stato del paziente dopo l'ultima visita registrata. È possibile, infatti, risalire al momento dell'ultimo controllo individuando il *delta* maggiore tra quelli presenti per quel soggetto, ma non vi è

	SubjectID		DataType			FeatureType	FeatureValue	
	V1	V2	V3	V4	V5	V6	V7	
1	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1203	Demographics Delta	0	Static
2	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1204	Ethnicity		
3	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1207	Race - Asian		
4	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1208	Race - Black/African American		
5	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1211	Race - Caucasian	1	
6	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1257	Age	48	
7	649	144	Demographics	5AC60165-78AA-4E1D-8CCF-F1A21B944A8B	1393	Race - Other		
8	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1213	1. Speech	3	Dynamic $\Delta = 0$
9	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1214	10. Respiratory	4	
10	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1215	2. Salivation	3	
11	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1216	3. Swallowing	2	
12	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1217	4. Handwriting	4	
13	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1218	5a. Cutting without Gastrostomy	4	
14	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1219	5b. Cutting with Gastrostomy		
15	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1220	6. Dressing and Hygiene	4	
16	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1221	7. Turning in Bed	4	
17	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1222	8. Walking	4	
18	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1223	9. Climbing Stairs	4	
19	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1225	ALSFRS Delta	0	
20	649	145	ALSFRS(R)	B2F1F8AC-6BEA-483C-9BC8-F13C51ED6FFB	1228	ALSFRS Total	36	
21	649	145	ALSFRS(R)	DFE7B123-4038-4044-89F3-738443022DBE	1213	1. Speech	3	$\Delta = 32$
22	649	145	ALSFRS(R)	DFE7B123-4038-4044-89F3-738443022DBE	1214	10. Respiratory	4	

Figura 3.1: Esempio di dati contenuti nella versione del database PRO-ACT proposta per la Challenge *Dream7: Phil Bowen ALS Prediction Prize4Life*. L'ID del soggetto è riportato nella colonna V1, il tipo di dato nella colonna V3, il tipo di feature nella colonna V6 ed il valore della feature in V7.

indicazione se in seguito il paziente sia deceduto, oppure uscito dal trial.

3.2 Preprocessing dei dati

Come accennato, per il preprocessing dei dati si sono replicate le metodologie proposte dal team vincitore della *Challenge DREAM7*, composto da Lester Mackey e Lilly Fang dell'Università di Stanford (CA) e riportate nel Supplementary di [28].

3.2.1 Rimozione record inconsistenti

In ciascun dataset, sono stati rimossi:

- i record contenenti feature che presentavano valori inconsistenti (ad esempio, *Subject Liters (Trial 1)* conteneva talvolta valori non numerici);

- i record relativi a feature dinamiche per cui non era indicato il valore del *delta* di misurazione;
- alcuni record che riportavano errori nell'istante temporale misurato. In certi casi, infatti, per uno stesso paziente sono presenti record multipli per una stessa feature ad un *delta* fissato. Non essendo possibile risalire al valore corretto tra quelli riportati, si è, per convenzione, mantenuto solo il primo in ordine di comparsa nel database;
- i record contenenti attributi dinamici misurati oltre il terzo mese dall'ingresso del soggetto nel trial clinici. Per utilizzare il metodo Random Survival Forests, infatti, è necessario fissare una soglia: al classificatore vengono passati solo gli attributi registrati prima dell'inizio del trial e durante i primi tre mesi dello studio clinico, e viene richiesto di predire la sopravvivenza nel futuro.

	subject.id	Onset.Delta	Symptom.WEAKNESS	Site.of.Onset.Onset..Limb	Race...Caucasian	Age	max.alsfrs.score	min.alsfrs.score
1	649	34.2034113	0.02668490	-0.07379078	-1.322176e-03	-0.876053803	0.222619553	0.379018919
2	2956	-102.9379323	0.03120578	0.02123356	3.198703e-03	0.766313392	0.005273863	0.062128016
3	3085	32.1982359	-0.06381855	-0.07379078	6.578123e-03	-1.677684515	0.605434047	0.753933115
4	3551	39.8348768	0.03032403	0.02035181	2.316954e-03	-0.042377487	0.165343054	0.129818122
5	4390	24.8851440	0.02749950	0.01752728	-5.075728e-04	-1.841451493	0.160627202	0.224893908
6	5155	-7.0380690	0.02945850	0.01948627	1.451418e-03	1.122619668	-0.330470408	-0.270121684
7	5936	22.0164450	-0.06381855	-0.07379078	-2.327700e-04	-0.182612303	0.445023882	0.508740982
8	7399	27.9547761	0.03332886	0.02335664	5.321787e-03	-2.208649697	0.364654780	0.125820516
9	8480	-9.9327165	0.03332886	0.02335664	5.321787e-03	2.065836644	0.267507363	0.222967933

Figura 3.2: Esempio di training set preprocessato, in cui ogni riga corrisponde ad un soggetto.

3.2.2 Importazione delle feature

Per ogni dataset, a partire dai record associati ad uno stesso paziente si è estratto un vettore di feature numeriche, riportato come riga nel dataset preprocessato (vedi Figura 3.2). Ciascun vettore viene utilizzato per il fit del classificatore, nel caso del training set, e per la predizione, sui dataset di test e validazione.

Si sono applicate diverse procedure per gestire le feature, a seconda del loro tipo, secondo quanto illustrato nelle prossime sezioni.

Variabili statiche

Per quanto riguarda le variabili statiche, si sono importate le feature riferite ai dati demografici (*Demographics*), alla storia medica del paziente (*Subject ALS History*) e della sua famiglia (*Family History*) e al trattamento a cui è stato sottoposto il soggetto (*Treatment Group*). In particolare, per ciascun soggetto:

- L'informazione *Onset Delta*, relativa alla comparsa dei primi sintomi, è stata importata direttamente. Essa corrisponde in tutti i casi a valori temporali negativi, come ci si può aspettare, dal momento che sono stati coinvolti nei trial clinici soggetti con sintomatologia e diagnosi conclamate.

- I sintomi registrati nella storia medica del paziente sono contenuti in diverse feature (*Symptom*, *Symptom - Other (Specify)*, *Location*). A seconda del trial, si osservano diverse codifiche per la registrazione dello stesso sintomo e i valori delle feature corrispondenti sono, di conseguenza, dismomogenei e talvolta costituiti da testo libero.
  Si sono quindi create undici nuove feature binarie, ciascuna corrispondente ad un sintomo specifico (weakness, swallowing, cramps, ecc.), contrassegnate da 1 se il sintomo è stato registrato per quel paziente, 0 altrimenti. Per la lista delle feature relative alla sintomatologia, si rimanda alla Tabella 3.1.

- La feature relativa al sito di esordio dei primi sintomi della malattia, *Site of Onset*, assume nel dataset originale valori disomogenei: *Onset:Bulbar* o valore *1* nel caso di esordio bulbare, *Onset:Limb* o valore *3* nel caso di esordio spinale, *Onset:Limb and Bulbar* se l'esordio è misto.
  Nel dataset preprocessato si sono quindi create tre nuove feature binarie, denominate *Site.of.Onset..Bulbar*, *Site.of.Onset..Limb* e *Site.of.Onset..Limb.and.Bulbar*, in cui solo la variabile corrispondente al sito di esordio del soggetto, se registrato, assume valore 1, 0 le altre.

- L'etnia dei soggetti è registrata nel database originale tramite le variabili binarie *Race - Asian*, *Race - Black/African American*, *Race - Caucasian* e *Race - Other*, riportate nel dataset preprocessato rispettivamente come *Race..Asian*, *Race..Black.African.American*, *Race..Caucasian*, *Race..Other*. Si è mantenuto il valore 1 per la feature corrispondente all'etnia del soggetto, 0 altrove.

- L'informazione relativa a *Age*, indicante gli anni compiuti del soggetto al momento dell'ingresso nel trial, è importata direttamente.

- Per quanto riguarda l'informazione demografica sul genere del paziente, si osservano nel database originale valori disomogenei per la feature *Sex*. Questa compare, infatti, talvolta etichettato come variabile categorica *Male* o *Female*, altre volte come valore numerico (nell'ordine) *1* o *2*.
  Si sono quindi create due nuove feature binarie nel database preprocessato, denominate rispettivamente *Sex.Male* e *Sex.Female*: viene assegnato valore 1 a quella corrispondente al sesso del paziente, 0 all'altra. Nel caso in cui non sia registrata l'informazione sul sesso, entrambe assumono valore 0.

- Per quanto riguarda l'anamnesi familiare, si è riportata una feature per ciascun componente della famiglia di cui può essere presente l'informazione medica. Queste variabili vengono contrassegnate con 1 nel caso in cui tale informazione sia presente, 0 altrimenti. Per la lista completa, si rimanda alla Tabella 3.1.

Si è inoltre aggiunta un'ulteriore feature binaria, denominata *Family*, che assume valore 1 se sono presenti informazioni su almeno un componente della famiglia, 0 altrimenti.

- Sempre estratte dall'anamnesi familiare, sono presenti alcune variabili indicanti patologie neurologiche in parenti del soggetto. Tali informazioni, descritte originariamente negli attributi *Neurological Disease* e *Neurological Disease Other Specify*, presentano valori diversi a seconda del trial. Si sono mantenute le variabili più frequenti, registrandole in nove nuove feature (vedi Tabella 3.1).

- Per alcuni soggetti, infine, è presente l'informazione sul Trattamento a cui sono stati sottoposti, denominata *Study Arm*. La label corrispondente (*PLACEBO*/*ACTIVE*) è affiancata dal valore temporale di registrazione *Treatment Group Delta*.
  Si è sfruttato il *delta* di registrazione per isolare le informazioni disponibili entro i primi tre mesi di trial del soggetto (ovvero si è selezionato *Treatment Group Delta* ≤ 92 giorni). Di queste, si è registrato il valore della label nelle due variabili binarie *Study.Arm.PLACEBO* e *Study.Arm.ACTIVE*, assegnando 1 a quella corrispondente al trattamento del soggetto, 0 all'altra.

La lista completa delle feature statiche ricavate nel preprocessing per ciascun soggetto è riportata in Tabella 3.1.

Variabili dinamiche

Le variabili dinamiche sono: punteggi *ALSFRS* singoli e totale, misure *FVC* e *SVC*, valutazioni dei *Vital Signs* (*Segni Vitali*: peso, altezza, tasso respiratorio, pressione sanguigna sistolica e diastolica) ed *Esami di Laboratorio*. Per ciascuna di queste feature, infatti, sono disponibili più misurazioni (una per ogni visita).

Allo scopo di predire la sopravvivenza futura del paziente, si sono prese in cosiderazione solo le visite effettuate nei primi tre mesi del trial, ovvero con *delta* ≤ 91 giorni. Di seguito, il termine "misure"sarà quindi riferito sempre ai valori con queste limitazioni temporali.

Si è scelto di replicare le scelte sperimentali di Mackey e Fang, escludendo dall'analisi le variabili dinamiche associate agli esami di laboratorio (*Laboratory Data*). Tale scelta trova probabile giustificazione nell'eterogeneità degli stessi e nella presenza di molti missing values.

Per ciascuna serie temporale relativa ad *ALFSRS*, *FVC*, *SVC* e *Vital Signs* (*delta* $\in [0, 91]$ giorni), si sono estratte delle feature numeriche descrittive, in particolare:

Onset.Delta	Father
Symptom.Speech	Grandfather..Maternal.
Symptom.WEAKNESS	Grandmother
Symptom.OTHER	Grandmother..Maternal.
Symptom.Swallowing	Mother
Symptom.GAIT_CHANGES	Uncle
Symptom.Atrophy	Uncle..Maternal.
Symptom.Cramps	Uncle..Paternal.
Symptom.Fasciculations	Son
Symptom.SENSORY_CHANGES	Daughter
Symptom.Stiffness	Sister
Symptom..	Brother
Site.of.Onset.Onset..Bulbar	Family
Site.of.Onset.Onset..Limb	Neurological.Disease.OTHER
Site.of.Onset.Onset..Limb.and.Bulbar	Neurological.Disease.STROKE.NOS
Race...Asian	Neurological.Disease.DEMENTIA.NOS
Race...Black.African.American	Neurological.Disease.PARKINSON.S.DISEASE
Race...Caucasian	Neurological.Disease.DAT
Race...Other	Neurological.Disease.ALS
Age	Neurological.Disease.BRAIN.TUMOR
Sex.Female	Neurological.Disease.STROKE.ISCHEMIC
Sex.Male	Neurological.Disease.STROKE.HEMORRHAGIC
Aunt	Study.Arm.PLACEBO
Aunt..Maternal.	Study.Arm.ACTIVE
Cousin	

Tabella 3.1: Preprocessing - Feature statiche.

- Valore massimo delle misure
- Valore minimo delle misure
- Ultimo valore misurato
- Media delle misure
- Numero di misure
- Somma delle misure
- Valore del *delta* associato alla prima misura
- Valore del *delta* associato all'ultima misura
- Media dei quadrati delle misure
- Deviazione standard delle misure

- Pendenza (slope) della serie temporale, definita come:

$$\frac{(ultimo\ valore\ misurato - primo\ valore\ misurato)}{(delta\ dell'ultima\ misura - delta\ della\ prima\ misura)}$$

- Nel caso in cui ci fossero meno di due misure, si è creata la feature *lessthan2.nomevariabile*, a cui viene assegnato valore 1
- Nel caso in cui non ci fossero misure, si è creata la feature *no.data.nomevariabile*, a cui viene assegnato valore 1.

Le feature dinamiche ottenute dalle serie temporali di *ALFSRS*, *FVC*, *SVC* e *Vital Signs* sono riportate nella Tabella 3.2.

Per ciascuna delle tipologie di feature dinamiche di cui sopra, si è estratta un'ulteriore serie temporale, detta "serie derivativa". Questa è stata ricavata a partire da coppie di misure temporalmente consecutive, di cui è stata calcolata la pendenza (*slope*), come:

$$slope = \frac{(secondo\ valore\ misurato - primo\ valore\ misurato)}{(delta\ della\ seconda\ misura - delta\ della\ prima\ misura)}$$

Tale valore è stato quindi associato al *delta* intermedio tra le due misure, ottenuto come:

$$delta_{slope} = \frac{(delta\ della\ seconda\ misura - delta\ della\ prima\ misura)}{2}$$

Il procedimento di estrazione della serie derivativa è esemplificato in Figura 3.3.

Da ciascuna serie derivativa si è estratto un set di feature numeriche descrittive, analogamente a quanto fatto per la serie originale, contenenti la label *slope*. Tali attributi sono riportati in Tabella 3.3.

3.2.3 Riduzione del numero di feature

Dopo la prima fase di preprocessing, per ciascun dataset di partenza (training, test e validation set) si è ottenuto quindi un nuovo dataset, con un numero di righe corrispondente al numero di pazienti e 492 colonne (pari alla somma della colonna per il codice identificativo del soggetto, delle feature statiche preprocessate (49) e delle feature numeriche ottenute dalle serie temporali originali (221) e derivative (221) delle variabili dinamiche).

Dal momento che molte feature presentavano un gran numero di valori pari a *0* o mancanti (indicati con *NA*), si è applicata un'operazione di pruning, per mantenere solo le colonne di attributi più complete.

- Nel caso del training set preprocessato, sono state mantenute solo le colonne che presentavano valori diversi da 0 o *NA* per almeno metà dei pazienti ($\geq$ 918/2 soggetti).

- I dataset di test e validazione preprocessati sono stati uniformati a quello di training ottenuto dopo il pruning di cui sopra, mantenendo solo le colonne corrispondenti alle feature non rimosse al passo precedente.

Al termine di questo passaggio, il numero di colonne dei dataset preprocessati risulta pari a 238.

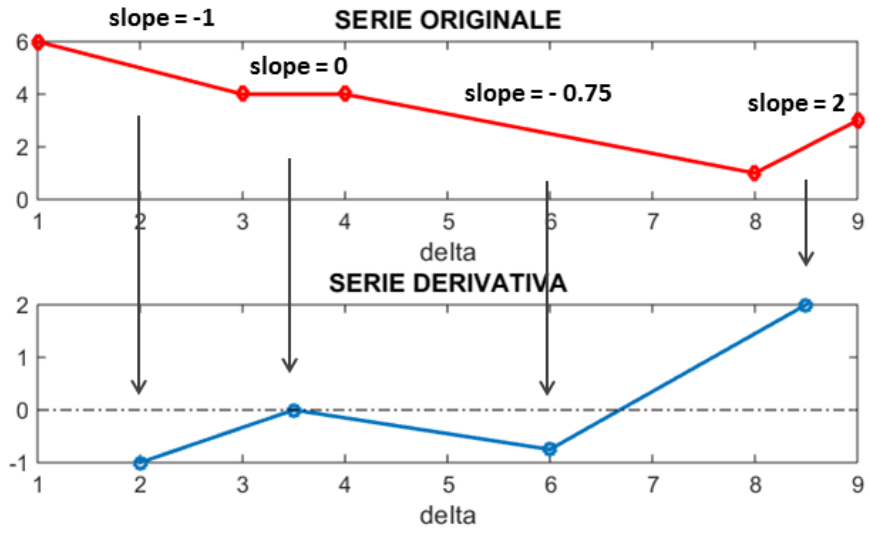

Figura 3.3: Feature dinamiche - Esempio di calcolo della serie derivativa.

max.alsfrs.score	min.alsfrs.score	last.alsfrs.score
mean.alsfrs.score	num.alsfrs.score.visits	sum.alsfrs.score
first.alsfrs.score.date	last.alsfrs.score.date	meansquares.alsfrs.score
sd.alsfrs.score	alsfrs.score.slope	lessthan2.alsfrs.score
no.alsfrs.score.data	max.speech	min.speech
last.speech	mean.speech	num.speech.visits
sum.speech	first.speech.date	last.speech.date
meansquares.speech	sd.speech	speech.slope
lessthan2.speech	no.speech.data	max.salivation
min.salivation	last.salivation	mean.salivation
num.salivation.visits	sum.salivation	first.salivation.date
last.salivation.date	meansquares.salivation	sd.salivation
salivation.slope	lessthan2.salivation	no.salivation.data
max.swallowing	min.swallowing	last.swallowing

mean.swallowing	num.swallowing.visits	sum.swallowing
first.swallowing.date	last.swallowing.date	meansquares.swallowing
sd.swallowing	swallowing.slope	lessthan2.swallowing
no.swallowing.data	max.handwriting	min.handwriting
last.handwriting	mean.handwriting	num.handwriting.visits
sum.handwriting	first.handwriting.date	last.handwriting.date
meansquares.handwriting	sd.handwriting	handwriting.slope
lessthan2.handwriting	no.handwriting.data	max.cutting
min.cutting	last.cutting	mean.cutting
num.cutting.visits	sum.cutting	first.cutting.date
last.cutting.date	meansquares.cutting	sd.cutting
cutting.slope	lessthan2.cutting	no.cutting.data
max.dressing	min.dressing	last.dressing
mean.dressing	num.dressing.visits	sum.dressing
first.dressing.date	last.dressing.date	meansquares.dressing
sd.dressing	dressing.slope	lessthan2.dressing
no.dressing.data	max.turning	min.turning
last.turning	mean.turning	num.turning.visits
sum.turning	first.turning.date	last.turning.date
meansquares.turning	sd.turning	turning.slope
lessthan2.turning	no.turning.data	max.walking
min.walking	last.walking	mean.walking
num.walking.visits	sum.walking	first.walking.date
last.walking.date	meansquares.walking	sd.walking
walking.slope	lessthan2.walking	no.walking.data
max.climbing.stairs	min.climbing.stairs	last.climbing.stairs
mean.climbing.stairs	num.climbing.stairs.visits	sum.climbing.stairs
first.climbing.stairs.date	last.climbing.stairs.date	meansquares.climbing.stairs
sd.climbing.stairs	climbing.stairs.slope	lessthan2.climbing.stairs
no.climbing.stairs.data	max.fvc.liters	min.fvc.liters
last.fvc.liters	mean.fvc.liters	num.fvc.liters.visits
sum.fvc.liters	first.fvc.liters.date	last.fvc.liters.date
meansquares.fvc.liters	sd.fvc.liters	fvc.liters.slope
lessthan2.fvc.liters	no.fvc.liters.data	max.svc.liters
min.svc.liters	last.svc.liters	mean.svc.liters
num.svc.liters.visits	sum.svc.liters	first.svc.liters.date
last.svc.liters.date	meansquares.svc.liters	sd.svc.liters
svc.liters.slope	lessthan2.svc.liters	no.svc.liters.data
max.weight	min.weight	last.weight
mean.weight	num.weight.visits	sum.weight
first.weight.date	last.weight.date	meansquares.weight
sd.weight	weight.slope	lessthan2.weight
no.weight.data	max.height	min.height

last.height	mean.height	num.height.visits
sum.height	first.height.date	last.height.date
meansquares.height	sd.height	height.slope
lessthan2.height	no.height.data	max.resp.rate
min.resp.rate	last.resp.rate	mean.resp.rate
num.resp.rate.visits	sum.resp.rate	first.resp.rate.date
last.resp.rate.date	meansquares.resp.rate	sd.resp.rate
resp.rate.slope	lessthan2.resp.rate	no.resp.rate.data
max.bp.diastolic	min.bp.diastolic	last.bp.diastolic
mean.bp.diastolic	num.bp.diastolic.visits	sum.bp.diastolic
first.bp.diastolic.date	last.bp.diastolic.date	meansquares.bp.diastolic
sd.bp.diastolic	bp.diastolic.slope	lessthan2.bp.diastolic
no.bp.diastolic.data	max.bp.systolic	min.bp.systolic
last.bp.systolic	mean.bp.systolic	num.bp.systolic.visits
sum.bp.systolic	first.bp.systolic.date	last.bp.systolic.date
meansquares.bp.systolic	sd.bp.systolic	bp.systolic.slope
lessthan2.bp.systolic	no.bp.systolic.data	

Tabella 3.2: Preprocessing - Feature numeriche ricavate dalle variabili dinamiche, serie originali.

max.slope.alsfrs.score	min.slope.alsfrs.score
last.slope.alsfrs.score	mean.slope.alsfrs.score
num.slope.alsfrs.score.visits	sum.slope.alsfrs.score
first.slope.alsfrs.score.date	last.slope.alsfrs.score.date
meansquares.slope.alsfrs.score	sd.slope.alsfrs.score
slope.alsfrs.score.slope	lessthan2.slope.alsfrs.score
no.slope.alsfrs.score.data	max.slope.speech
min.slope.speech	last.slope.speech
mean.slope.speech	num.slope.speech.visits
sum.slope.speech	first.slope.speech.date
last.slope.speech.date	meansquares.slope.speech
sd.slope.speech	slope.speech.slope
lessthan2.slope.speech	no.slope.speech.data
max.slope.salivation	min.slope.salivation
last.slope.salivation	mean.slope.salivation
num.slope.salivation.visits	sum.slope.salivation
first.slope.salivation.date	last.slope.salivation.date
meansquares.slope.salivation	sd.slope.salivation
slope.salivation.slope	lessthan2.slope.salivation
no.slope.salivation.data	max.slope.swallowing

min.slope.swallowing	last.slope.swallowing
mean.slope.swallowing	num.slope.swallowing.visits
sum.slope.swallowing	first.slope.swallowing.date
last.slope.swallowing.date	meansquares.slope.swallowing
sd.slope.swallowing	slope.swallowing.slope
lessthan2.slope.swallowing	no.slope.swallowing.data
max.slope.handwriting	min.slope.handwriting
last.slope.handwriting	mean.slope.handwriting
num.slope.handwriting.visits	sum.slope.handwriting
first.slope.handwriting.date	last.slope.handwriting.date
meansquares.slope.handwriting	sd.slope.handwriting
slope.handwriting.slope	lessthan2.slope.handwriting
no.slope.handwriting.data	max.slope.cutting
min.slope.cutting	last.slope.cutting
mean.slope.cutting	num.slope.cutting.visits
sum.slope.cutting	first.slope.cutting.date
last.slope.cutting.date	meansquares.slope.cutting
sd.slope.cutting	slope.cutting.slope
lessthan2.slope.cutting	no.slope.cutting.data
max.slope.dressing	min.slope.dressing
last.slope.dressing	mean.slope.dressing
num.slope.dressing.visits	sum.slope.dressing
first.slope.dressing.date	last.slope.dressing.date
meansquares.slope.dressing	sd.slope.dressing
slope.dressing.slope	lessthan2.slope.dressing
no.slope.dressing.data	max.slope.turning
min.slope.turning	last.slope.turning
mean.slope.turning	num.slope.turning.visits
sum.slope.turning	first.slope.turning.date
last.slope.turning.date	meansquares.slope.turning
sd.slope.turning	slope.turning.slope
lessthan2.slope.turning	no.slope.turning.data
max.slope.walking	min.slope.walking
last.slope.walking	mean.slope.walking
num.slope.walking.visits	sum.slope.walking
first.slope.walking.date	last.slope.walking.date
meansquares.slope.walking	sd.slope.walking
slope.walking.slope	lessthan2.slope.walking
no.slope.walking.data	max.slope.climbing.stairs
min.slope.climbing.stairs	last.slope.climbing.stairs
mean.slope.climbing.stairs	num.slope.climbing.stairs.visits
sum.slope.climbing.stairs	first.slope.climbing.stairs.date
last.slope.climbing.stairs.date	meansquares.slope.climbing.stairs

sd.slope.climbing.stairs	slope.climbing.stairs.slope
lessthan2.slope.climbing.stairs	no.slope.climbing.stairs.data
max.slope.fvc.liters	min.slope.fvc.liters
last.slope.fvc.liters	mean.slope.fvc.liters
num.slope.fvc.liters.visits	sum.slope.fvc.liters
first.slope.fvc.liters.date	last.slope.fvc.liters.date
meansquares.slope.fvc.liters	sd.slope.fvc.liters
slope.fvc.liters.slope	lessthan2.slope.fvc.liters
no.slope.fvc.liters.data	max.slope.svc.liters
min.slope.svc.liters	last.slope.svc.liters
mean.slope.svc.liters	num.slope.svc.liters.visits
sum.slope.svc.liters	first.slope.svc.liters.date
last.slope.svc.liters.date	meansquares.slope.svc.liters
sd.slope.svc.liters	slope.svc.liters.slope
lessthan2.slope.svc.liters	no.slope.svc.liters.data
max.slope.weight	min.slope.weight
last.slope.weight	mean.slope.weight
num.slope.weight.visits	sum.slope.weight
first.slope.weight.date	last.slope.weight.date
meansquares.slope.weight	sd.slope.weight
slope.weight.slope	lessthan2.slope.weight
no.slope.weight.data	max.slope.height
min.slope.height	last.slope.height
mean.slope.height	num.slope.height.visits
sum.slope.height	first.slope.height.date
last.slope.height.date	meansquares.slope.height
sd.slope.height	slope.height.slope
lessthan2.slope.height	no.slope.height.data
max.slope.resp.rate	min.slope.resp.rate
last.slope.resp.rate	mean.slope.resp.rate
num.slope.resp.rate.visits	sum.slope.resp.rate
first.slope.resp.rate.date	last.slope.resp.rate.date
meansquares.slope.resp.rate	sd.slope.resp.rate
slope.resp.rate.slope	lessthan2.slope.resp.rate
no.slope.resp.rate.data	max.slope.bp.diastolic
min.slope.bp.diastolic	last.slope.bp.diastolic
mean.slope.bp.diastolic	num.slope.bp.diastolic.visits
sum.slope.bp.diastolic	first.slope.bp.diastolic.date
last.slope.bp.diastolic.date	meansquares.slope.bp.diastolic
sd.slope.bp.diastolic	slope.bp.diastolic.slope
lessthan2.slope.bp.diastolic	no.slope.bp.diastolic.data
max.slope.bp.systolic	min.slope.bp.systolic
last.slope.bp.systolic	mean.slope.bp.systolic

num.slope.bp.systolic.visits	sum.slope.bp.systolic
first.slope.bp.systolic.date	last.slope.bp.systolic.date
meansquares.slope.bp.systolic	sd.slope.bp.systolic
slope.bp.systolic.slope	lessthan2.slope.bp.systolic
no.slope.bp.systolic.data	

Tabella 3.3: Preprocessing - Feature numeriche ricavate dalle variabili dinamiche, serie derivative.

3.2.4 Imputazione dei missing values

Si è quindi proceduto con un passaggio di imputazione, ovvero di sostituizione dei valori mancanti (NA) nei database risultanti dopo il pruning con valori plausibili della feature corrispondente.

Nonostante il metodo Random Survival Forests sia in grado di gestire dataset con missing value, applicando al suo interno opportune procedure di imputazione (si omettono i dettagli, reperibili in [25]), si è preferito aggiungere al preprocessing un passaggio dedicato.

Nel dettaglio:

- si è calcolata la mediana di ciascuna colonna del training set e la si è sostituita ai valori *NA* per tutti i soggetti. La scelta della mediana, rispetto alla media, è stata motivata dalla maggiore robustezza agli outlier;
- i valori mediani ottenuti per le colonne del training set sono stati utilizzate per imputare sia il training set stesso, sia il test ed il validation set.

La scelta di sfruttare in tutti i casi le mediane del training set è dettata dall'ottica di applicazione del classificatore: se ne prospetta, infatti, un utilizzo in real-time, con inserimento dei dati relativi ad un singolo paziente e conseguente immediata predizione.

3.2.5 Normalizzazione

Si è poi implementato un passaggio di standardizzazione delle feature, sottraendo alle colonne la loro media e centrandone i valori sullo zero.

Questo passaggio è probabilmente quello che rende meno intuitiva la consultazione dei dataset preprocessati, dal momento che la sottrazione della media può portare variabili che ci si aspetterebbe sempre positive (ad esempio, l'età del soggetto) ad assumere valori negativi.

Nello specifico:

- si è calcolata la media di ciascuna colonna del training set preprocessato;
- le medie ottenute dal training set sono state sottratte alle colonne corrispondenti dei dataset.

Si osserva come, anche in questo caso, si siano utilizzate le medie estratte dal training set per normalizzare test e validazione. La motivazione coincide con gli obiettivi applicativi illustrati nel caso dell'imputazione.

3.2.6 Informazioni survival

Come osservato all'inizio di questo Capitolo, non sono presenti nei dataset della *Challenge DREAM7* le informazioni riguardanti lo stato dei pazienti vivi all'ultima visita registrata.

Si è fatto quindi uso di un'estensione di questo dataset, resa pubblica per un'ulteriore Challenge sulla SLA nel 2015 (*DREAM ALS Stratification Prize4Life Challenge*), contenente per ciascun soggetto le informazioni survival.

A partire da questi dati di sopravvivenza, registrati per circa 7300 pazienti, si sono incrociati gli identificativi dei soggetti per estrarre le informazioni per il subset di interesse.

Le informazioni survival ottenute sono:

- il *time_event*, indicante il tempo di censura o di decesso del paziente;
- lo *status*, ovvero una feature binaria che assume valore 1 in caso di morte, 0 in caso di censura temporale.

Queste due feature sono state aggiunte come colonne finali ai dataset preprocessati.

I dataset di training, test e validazione sono costituiti, alla fine del preprocessing, da un totale di 240 colonne ciascuno, compresa una prima colonna contenente gli identificativi dei soggetti (vedi Tabella 3.4 per la lista definitiva delle feature).

Onset.Delta	Symptom.WEAKNESS
Site.of.Onset.Onset..Limb	Race...Caucasian
Age	max.alsfrs.score
min.alsfrs.score	last.alsfrs.score
mean.alsfrs.score	num.alsfrs.score.visits
sum.alsfrs.score	last.alsfrs.score.date
meansquares.alsfrs.score	sd.alsfrs.score
alsfrs.score.slope	max.speech
min.speech	last.speech

mean.speech	num.speech.visits
sum.speech	last.speech.date
meansquares.speech	max.salivation
min.salivation	last.salivation
mean.salivation	num.salivation.visits
sum.salivation	last.salivation.date
meansquares.salivation	max.swallowing
min.swallowing	last.swallowing
mean.swallowing	num.swallowing.visits
sum.swallowing	last.swallowing.date
meansquares.swallowing	max.handwriting
min.handwriting	last.handwriting
mean.handwriting	num.handwriting.visits
sum.handwriting	last.handwriting.date
meansquares.handwriting	max.cutting
min.cutting	last.cutting
mean.cutting	num.cutting.visits
sum.cutting	last.cutting.date
meansquares.cutting	max.dressing
min.dressing	last.dressing
mean.dressing	num.dressing.visits
sum.dressing	last.dressing.date
meansquares.dressing	max.turning
min.turning	last.turning
mean.turning	num.turning.visits
sum.turning	last.turning.date
meansquares.turning	max.walking
min.walking	last.walking
mean.walking	num.walking.visits
sum.walking	last.walking.date
meansquares.walking	max.climbing.stairs
min.climbing.stairs	last.climbing.stairs
mean.climbing.stairs	num.climbing.stairs.visits
sum.climbing.stairs	last.climbing.stairs.date
meansquares.climbing.stairs	max.fvc.liters
min.fvc.liters	last.fvc.liters
mean.fvc.liters	num.fvc.liters.visits
sum.fvc.liters	last.fvc.liters.date
meansquares.fvc.liters	sd.fvc.liters
fvc.liters.slope	lessthan2.svc.liters
no.svc.liters.data	max.weight
min.weight	last.weight
mean.weight	num.weight.visits

sum.weight	last.weight.date
meansquares.weight	sd.weight
weight.slope	lessthan2.height
no.height.data	max.resp.rate
min.resp.rate	last.resp.rate
mean.resp.rate	num.resp.rate.visits
sum.resp.rate	last.resp.rate.date
meansquares.resp.rate	sd.resp.rate
resp.rate.slope	max.bp.diastolic
min.bp.diastolic	last.bp.diastolic
mean.bp.diastolic	num.bp.diastolic.visits
sum.bp.diastolic	last.bp.diastolic.date
meansquares.bp.diastolic	sd.bp.diastolic
bp.diastolic.slope	max.bp.systolic
min.bp.systolic	last.bp.systolic
mean.bp.systolic	num.bp.systolic.visits
sum.bp.systolic	last.bp.systolic.date
meansquares.bp.systolic	sd.bp.systolic
bp.systolic.slope	max.slope.alsfrs.score
min.slope.alsfrs.score	last.slope.alsfrs.score
mean.slope.alsfrs.score	num.slope.alsfrs.score.visits
sum.slope.alsfrs.score	first.slope.alsfrs.score.date
last.slope.alsfrs.score.date	meansquares.slope.alsfrs.score
sd.slope.alsfrs.score	slope.alsfrs.score.slope
num.slope.speech.visits	first.slope.speech.date
last.slope.speech.date	num.slope.salivation.visits
first.slope.salivation.date	last.slope.salivation.date
num.slope.swallowing.visits	first.slope.swallowing.date
last.slope.swallowing.date	num.slope.handwriting.visits
first.slope.handwriting.date	last.slope.handwriting.date
num.slope.cutting.visits	first.slope.cutting.date
last.slope.cutting.date	num.slope.dressing.visits
first.slope.dressing.date	last.slope.dressing.date
num.slope.turning.visits	first.slope.turning.date
last.slope.turning.date	num.slope.walking.visits
first.slope.walking.date	last.slope.walking.date
num.slope.climbing.stairs.visits	first.slope.climbing.stairs.date
last.slope.climbing.stairs.date	max.slope.fvc.liters
min.slope.fvc.liters	last.slope.fvc.liters
mean.slope.fvc.liters	num.slope.fvc.liters.visits
sum.slope.fvc.liters	first.slope.fvc.liters.date
last.slope.fvc.liters.date	meansquares.slope.fvc.liters
sd.slope.fvc.liters	slope.fvc.liters.slope

lessthan2.slope.svc.liters	no.slope.svc.liters.data
max.slope.weight	min.slope.weight
last.slope.weight	mean.slope.weight
num.slope.weight.visits	sum.slope.weight
first.slope.weight.date	last.slope.weight.date
meansquares.slope.weight	sd.slope.weight
slope.weight.slope	lessthan2.slope.height
no.slope.height.data	max.slope.resp.rate
min.slope.resp.rate	mean.slope.resp.rate
num.slope.resp.rate.visits	sum.slope.resp.rate
first.slope.resp.rate.date	last.slope.resp.rate.date
meansquares.slope.resp.rate	sd.slope.resp.rate
slope.resp.rate.slope	max.slope.bp.diastolic
min.slope.bp.diastolic	last.slope.bp.diastolic
mean.slope.bp.diastolic	num.slope.bp.diastolic.visits
sum.slope.bp.diastolic	first.slope.bp.diastolic.date
last.slope.bp.diastolic.date	meansquares.slope.bp.diastolic
sd.slope.bp.diastolic	slope.bp.diastolic.slope
max.slope.bp.systolic	min.slope.bp.systolic
last.slope.bp.systolic	mean.slope.bp.systolic
num.slope.bp.systolic.visits	sum.slope.bp.systolic
first.slope.bp.systolic.date	last.slope.bp.systolic.date
meansquares.slope.bp.systolic	sd.slope.bp.systolic
slope.bp.systolic.slope	time_event
status	

Tabella 3.4: Preprocessing - Feature definitive ottenute per i tre dataset di training, test e validation analizzati.

Capitolo 4

Applicazione del metodo RSF

Allo scopo di ottenere una stima individuale del rischio di mortalità e della probabilità di sopravvivenza dei soggetti a partire dalle informazioni cliniche e demografiche registrate durante i primi tre mesi di trial, si è applicato il metodo Random Survival Forests ai dataset preprocessati di training, test e validazione.

A livello operativo, si è utilizzato il pacchetto R `randomForestSRC` (*Random Forests for Survival, Regression and Classification*) che implementa diversi metodi basati sulle Random Forests proposte da Breiman [8], tra cui RSF.

Le due funzioni principali, tra quelle proposte nel pacchetto, sono:

- `rfsrc`: cresce una Foresta di Alberi Decisionali a partire da un training set fornito dall'utente, permettendo di scegliere la tipologia di analisi tra Classificazione, Regressione e Survival, a seconda che l'output atteso sia rispettivamente categorico, numerico o right-censored.
  Una volta che RSF è stato allenato sul training set (fit), si ottengono diversi output: la Foresta, costruita secondo i parametri decisi dall'utente, la predizione sul training set ed alcune valutazioni della performance del modello (es. Error rate, VIMP);

- `predict.rfsrc`: permette di sfruttare la Foresta costruita con `rfsrc` per implementare una classificazione, una regressione o una survival analysis su nuovi dataset, come per esempio test e validation set.
  In output vengono fornite le predizioni sul nuovo dataset ed alcune valutazioni della performance della Foresta su quel dataset specifico (es. Error rate, VIMP).

Sono inoltre disponibli ulteriori funzioni che permettono per esempio di gestire i plot degli output della Foresta, o di selezionare variabili nei dataset, secondo le esigenze dell'utente.

4.1 Fit del Classificatore RSF

La costruzione della Foresta è stata svolta utilizzando la funzione `rfsrc`, selezionando la metodologia di Analisi Survival RSF.

Per la fase di costruzione degli Alberi della Foresta, che coincide per questo metodo con l'apprendimento del classificatore, si è fatto uso del training set preprocessato (vedi Capitolo 3). Si ricorda che il training set è costituito da 918 righe, pari al numero di soggetti, e 240 colonne, contenti l'ID dei soggetti, 237 le variabili predittive e le 2 informazioni survival.

Il training set viene utilizzato sia per costruire la Foresta, sia per ottenere una stima delle performance del classificatore stesso. Le stesse istanze, infatti, sono fatte scendere lungo gli Alberi, nello specifico negli Alberi per cui sono OOB, ottenendo così una predizione e, da essa, la computazione dell'Errore di Predizione sul training set (vedi Sezione 2.4.6).

I risultati relativi ad output e performance per il training set sono riportati nel dettaglio nel Capitolo 5.

4.1.1 Tuning dei parametri utente

La funzione `rfsrc` permette all'utente di impostare i valori di alcuni parametri che determinano natura, forma e caratteristiche della foresta.
Si illustrano di seguito le scelte effettuate.

- Si è innanzitutto selezionata la tipologia Analisi Survival, che comporta l'implementazione del metodo Random Survival Forests come descritto da Ishwaran [25]. Vengono indicate all'algoritmo le colonne del dataset di training contenenti le informazioni survival {*time event, status*}.

- Si è omessa al classificatore la colonna degli ID dei soggetti, per evitare che venisse considerata come feature predittiva.

- Il protocollo di Bagging utilizzato (*by root*) seleziona i soggetti utilizzati nella costruzione di ciascun albero effettuando un campionamento con ripetizione da tutto il pool del training set. Per ciascun albero, i soggetti hanno tutti la stessa probabilità di essere scelti come in-bag. Si ricorda che, per quanto riguarda la percentuale di OOB sul totale, essi sono in media il 37% ad ogni Bagging, come da [25].

- Non si è imposto limite di profondità agli alberi.

- Per lo splitting ai nodi (vedi Sezione 2.4.5), è stato scelto il criterio deterministico basato sulla massimizzazione del test dei ranghi logaritmici. Per ciascun nodo, nella scelta casuale delle variabili candidate, a ciascuna feature è stata assegnata uguale probabilità di essere selezionata. Si è inoltre ammessa l'esistenza di una possibile relazione tra le feature utilizzate per lo split e le feature survival.

- Il calcolo della VIMP per le feature in input alla Foresta viene computato utilizzando il metodo proposto da Ishwaran per le RSF: la VIMP è calcolata come differenza tra l'OOB Prediction Error perturbato medio e l'OOB Prediction Error originale medio (vedi Sezione 2.4.7). In particolare, si è selezionata la modalità di permutazione all'interno del singolo nodo.
  La scelta di tale metodo permette di ottenere (1) una VIMP che rispecchi l'effetto di tutta la Foresta, e non la media degli effetti sui singoli Alberi, dal momento che i Prediction Error utilizzati sono già riferiti all'intera Foresta, e (2) un vantaggio computazionale in termini di velocità, specialmente in problemi con un gran numero di istanze, come nel caso dell'Analisi Survival [24].

- Dal momento che il preprocessing ha precedentemente imputato eventuali missing values nel training set, non è stato necessario gestire questo aspetto all'interno del metodo RSF.

4.1.2 Tuning dei parametri *B*, *m* e *nodesize*

Ci sono, inoltre, parametri che richiedono una procedura di tuning per determinarne il valore ottimo dato il training set di input: si tratta del numero **B** di alberi della foresta, del numero **m** di feature candidate per lo split ad ogni nodo e del numero minimo di istanze distinte che devono cadere in un nodo terminale (**nodesize**).

Per le considerazioni specifiche su queste quantità, si rimanda alla Sezione 2.4.1.

CROSS-VALIDATION

I valori ottimi dei parametri **B**, **m** e **nodesize** sono stati individuati applicando una 5x5-fold Cross-Validation (CV) sul training set in input.

In generale, una NxK-fold Cross-Validation costruisce Foreste utilizzando diverse combinazioni di valori dei tre parametri (*grid search*) e sottoinsiemi di istanze, calcolando per ciascun tentativo l'errore di predizione sulle istanze non sfruttate per il fit.

Per definire la griglia di possibili valori dei parametri, si sono sfruttate le indicazioni fornite nella documentazione del pacchetto [24]. Nello specifico, dati i 918 soggetti e le 237 variabili da utilizzare per la predizione, i valori indagati per i parametri sono:

· **B**: i numeri compresi tra 100 e 1000, a passo 100;

· **m**: gli arrotondamenti agli interi più vicini di $\sqrt{237}$, 237/2 e 237/3;

· **nodesize**: i numeri interi compresi tra 1 e 6.

Si riporta di seguito la descrizione della 5x5-fold Cross-Validation sul dataset di training preprocessato.

- Per N=5 volte, il training set viene suddiviso in K=5 parti uguali. A turno, quattro di queste parti vanno a costituire l'internal training *int_tr*, mentre la quinta viene utilizzata come internal test *int_ts*. In totale, nella 5x5-fold CV si sfruttano quindi 25 diverse suddivisioni del training set.

- Si seleziona una delle 25 suddivisioni del training set.
  Per ogni possibile combinazione dei parametri **B**, **m** e **nodesize** sulla grid di indagine:

  - Si considera *int_tr* e vi si applica una 5x2-fold CV: per cinque volte, *int_tr* viene suddiviso in due parti, che vengono prese a turno come training per il tuning (*int_tr_tun*) e test per il tuning (*int_ts_tun*).
  - Con la combinazione di parametri corrente, si cresce una RSF su *int_tr_tun*.
  - Si fanno quindi scendere le istanze di *int_ts_tun* lungo la Foresta ottenuta al passo precedente e si ottengono le predizioni.
  - Sfruttando le predizioni su *int_ts_tun*, si calcola l'Error rate della Foresta. Complessivamente, per la 5x2-fold CV interna, si ottengono 10 Error rate per lo stesso set di parametri.
  - Si calcola l'Error rate mediano tra i 10 ottenuti.

- Per ciascuna delle 25 suddivisioni del training set, ovvero per ciascun *int_tr* si ottiene un vettore di Error rate mediani in corrispondenza delle varie combinazioni di parametri.
  Si seleziona quindi, per ciascun *int_tr*, la combinazione di parametri che porta al minor Error rate mediano.

- Si allena su ciascun *int_tr* una RSF col miglior set di parametri selezionato per quella suddivisione. Si ottengono 25 modelli di RSF.

- Ciascuna RSF viene utilizzata sul proprio *int_ts*, ottenendo le predizioni.

- Dalle predizioni, si ricava l'Error rate della Foresta.
  Al termine della CV, quindi, si avranno 25 set di parametri, ciascuno utilizzato su una suddivisione del training set, e 25 Error rate corrispondenti.

- Si seleziona infine il set di parametri {**B**, **m** e **nodesize**} che si ripete più spesso o, in caso di set con la stessa frequenza, quello tra essi che produce il minimo Error rate.

Nel caso del training set di studio, i valori ottimi individuati per i parametri con la CV sono: **B=500** alberi, **m=15** variabili candidate per lo split a ciascun nodo e **nodesize=4** istanze distinte minime che cadono in ciascun nodo terminale.

4.2 Predizione sui dataset di test e validazione

Il modello di Random Survival Forests ottenuto nella fase di fit è stato quindi applicato ai dataset di test e di validazione, sfruttando la funzione `predict.rfsrc`.

Si ricorda che, nel caso specifico dei dataset proposti dalla Challenge, non è specificata [28] una differenza *a priori* tra le istanze contenute nei dataset di test e di validazione, in termini di difficoltà attesa nella predizione su di esse.

Nello svolgimento della Challenge, il test set veniva utilizzato durante la costruzione del classificatore per avere un feedback sulle sue performance ed, eventualmente, essere di indicazione su come modificarne i parametri (vedi Sezione 3.1).

Nel protocollo adottato con RSF, invece, la natura stessa della Cross-Validation coinvolge, nel tuning dei parametri della Foresta, solo il training set. I dataset di test e validazione fungono quindi, in questo lavoro, entrambi come dataset indipendenti, mai visti dal classificatore. Possono essere, in questo senso, intesi entrambi come set di istanze su cui verificare e valutare le performance della Foresta, senza aspettative *a priori*.

Le nuove istanze dei dataset di test e validazione vengono fatte scorrere lungo gli Alberi della Foresta creata sul training set. Per ciascuna, si ottiene una predizione, legata al CHF delle foglie. Anche in questo caso, viene svolto il calcolo della VIMP per valutare il peso delle singole feature nella predizione: i parametri forniti in input alla funzione `predict.rfsrc` determinano, per la computazione della VIMP sui nuovi dataset, le stesse modalità di permutazione ai nodi adottate per il training set.

Gli output ottenuti vengono riportati e commentati nel Capitolo 5, in cui vengono illustrati i risultati nel dettaglio.

Capitolo 5

Analisi dei risultati RSF

Le funzioni `rfsrc` e `predict.rfsrc` forniscono in output i risultati della predizione survival per i soggetti in input.

Dal momento che la Foresta è costruita sul training set, gli output per tutti i dataset sono relativi ai time event del training.
In particolare, si osserva che le predizioni ottenute col pacchetto `randomForestSRC` sono relative ad un sottoinsieme dei time event, corrispondente a tutti e soli i tempi di morte dei soggetti del training set, ovvero i time event associati a status=1 (dead). Questi istanti temporali sono indicati come *time of interest*.

Per lo specifico training set di 918 soggetti utilizzato per la costruzione di questa Foresta, i time of interest sono 100, compresi tra 338 e 955 giorni dalla data di inizio del trial, distribuiti in maniera non omogenea.

Sui time of interest sono calcolate le grandezze riportate di seguito.

- **Ensemble Cumulative Hazard Function**:
  per ciascuna istanza è riportata la sua Ensemble CHF, ovvero la media delle funzioni cumulative del rischio di morte per quel soggetto, calcolate ad ogni Albero con lo stimatore di Nelson-Aalen (vedi Sezione 2.4.3) sui time of interest.

  Per il training set si analizza l'output *chf.oob*, corrispondente all'OOB Ensemble CHF. Si considerano, infatti, per il training set solo le istanze OOB, al fine di ridurre gli effetti di bias dovuti all'utilizzo delle stesse istanze sia per costruire la Foresta, sia per ottenere su di esse la predizione.
  Non vale la stessa distinzione tra istanze OOB e in-bag, invece, per i dataset di test e validazione, dal momento che le loro istanze non intervengono nella costruzione della Foresta. Più genericamente, quindi, per i dataset di test e validazione si fa riferimento all'Ensemble CHF, riportato come output *chf*.

  Come si osserva nei grafici riportati nelle prossime Sezioni per i tre dataset, *chf.oob* e *chf* sono funzioni a gradino monotone crescenti, che rappresentano l'aumento del rischio di morte nel tempo per i soggetti.

- **Ensemble Mortality:**
  per ciascun soggetto, viene inoltre fornito in output il valore della Ensemble Mortality (vedi Sezione 2.4.4). Si ricorda che questa quantità corrisponde alla somma, sui soli tempi di morte, del valore dell'Ensemble CHF per quella istanza. Dal momento che l'Ensemble CHF è, nello specifico di `randomForestSRC`, fornito sui soli time of interest, per ciascun soggetto tale quantità corrisponde alla somma dei suoi valori *chf.oob/chf*.
  Anche in questo caso, viene effettuata una distinzione tra i dataset, a seconda che le istanze in essi contenute siano state utilizzate o meno per la costruzione della Foresta. Nel caso del training set, nuovamente si fa riferimento alle istanze OOB, ottenendo l'OOB Ensemble Mortality (indicata nel pacchetto con *predicted.oob*) a partire dal'OOB Ensemble CHF *chf.oob*. Per i dataset di test e validazione, invece, viene calcolata la più generica Ensemble Mortality (indicata solo con *predicted*), ottenuta a partire dall'Ensemble CHF *chf*.

- **Survival Function:**
  un'altra tipologia di output tipica dell'Analisi Survival, fornita spesso in alternativa o insieme al CHF, è la Survival Function. Mentre il CHF rappresenta la probabilità che in un individuo si verifichi l'evento morte, la Survival Function descrive la probabilità che l'individuo sopravviva, ovvero eviti l'evento morte.

  Formalmente, la Survival Function può essere stimata a partire dai time event (sia di morte che di censura), utilizzando il metodo di Kaplan-Meier [26].
  Mantenendo la notazione della Sezione 2.4.2, siano $T_i = t_0, ..., t_n$ i time event, distinti ed ordinati, per il gruppo di soggetti considerati. Preso tra questi un tempo t_l ed indicando con d_l e Y_l rispettivamente il numero di soggetti morti all'istante t_l e quelli a rischio fino a quel momento, la probabilità di sopravvivenza a t_l è ottenuta come:

$$S(t_l) = S(t_{l-1})(1 - \frac{d_l}{Y_l}),\ con\ t_{-1} = 0\ e\ S(t_{-1}) = 1. \tag{5.1}$$

  Ensemble CHF $H_e(t)$ e Survival Function $S(t)$ sono legati [13] da:

$$H_e(t) = -[logS(t)]. \tag{5.2}$$

  Come nel caso dell'Ensemble CHF, la Survival Function in `randomForestSRC` è fornita solo sui time of interest ed è presentata in output col nome di *survival.oob* per il training set, *survival* per test e validation set. Riguardo all'uso delle sole istanze OOB per il training set, valgono le medesime osservazioni fatte ai punti precedenti.

Inoltre, per ciascun dataset di input, viene fornita la stima della **Variabile Importance** (VIMP) per le feature utilizzate per la predizione (vedi Sezione 2.4.7).

La VIMP nel caso del training set è calcolata in modo standard, prendendo in ciascun Albero le istanze OOB e, su di esse, agendo con la permutazione ai nodi della feature di interesse. Per i dataset di test e validazione, invece, non potendo nuovamente parlare di OOB, la VIMP è calcolata facendo scendere tutte le istanze lungo gli Alberi della Foresta, prima senza perturbazioni, poi intervenendo con la permutazione ai nodi.

Il pacchetto `randomForestSRC` permette di ottenere un grafico a barre in cui sono riportate le 237 feature e, per ciascuna, il valore corrispondente della VIMP. Un esempio è riportato in Figura 5.1.

Si osserva come, ad un'ispezione visiva, siano facilmente identificabili le variabili più predittive (VIMP>0, riportate in blu), quelle non predittive (VIMP$=0$, nella parte centrale del grafico) e quelle antipredittive (VIMP<0, riportate in rosso). Si ricorda che, in quest'ultimo caso, valori negativi della VIMP stanno a significare che la perturbazione dei valori della variabile è più predittiva della variabile coi valori originali.

Nelle prossime Sezioni, in cui si analizzano nel dettaglio gli output di ciascun dataset, il grafico della VIMP viene riportato solo per le prime 20 variabili più predittive, allo scopo di identificare quelle che hanno maggiore impatto sulla predizione.

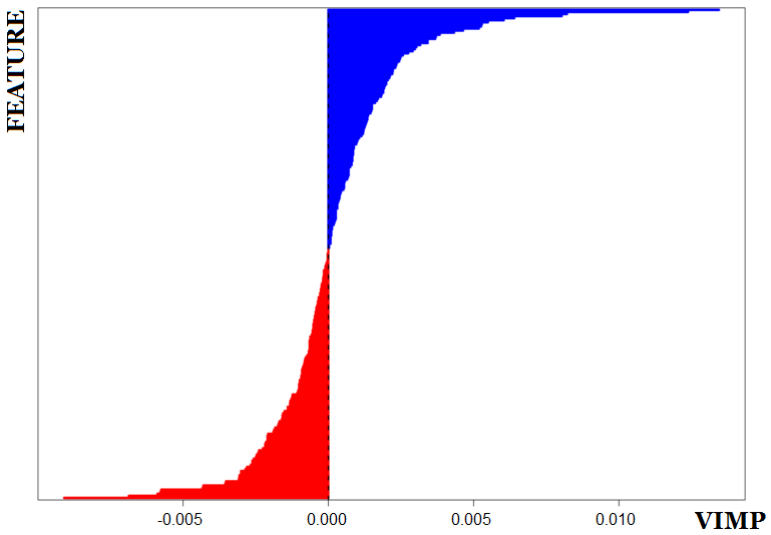

Figura 5.1: VIMP - Esempio di grafico a barre della VIMP di ogni feature. In blu i valori della VIMP per le feature più predittive, in rosso quelli per le meno predittive.

5.1 Training set

Il riepilogo delle caratteristiche della Foresta costruita sul training set con `rfsrc` è riportato in Tabella 5.1.

Sample size:	918
Number of deaths:	137
Number of trees:	500
Minimum terminal node size:	4
Average no. of terminal nodes:	128.496
No. of variables tried at each split:	15
Total no. of variables:	237
Analysis:	RSF
Family:	surv
Splitting rule:	logrank
Error rate:	34.73%

Tabella 5.1: Riepilogo della Foresta RSF costruita sul training set preprocessato.

PREDICTION ERROR

Si osserva per il training set un Error rate pari al 34.73%, che, per riportarsi alla notazione adottata nella Sezione 2.4.6, corrisponde all'OOB Prediction Error PE^{**}, ottenuto a partire dall'OOB Ensemble CHF.

OOB ENSEMBLE CHF

In Figura 5.2 sono riportate le curve OOB Ensemble CHF per i soggetti contenuti nel training set.

OOB ENSEMBLE MORTALITY

In Figura 5.3 sono riportati, in ordine crescente, i valori della OOB Ensemble Mortality per i 918 soggetti del training set.

Per meglio interpretarne la distribuzione, in Figura 5.4 è riportato l'istogramma dei valori assunti dalla OOB Ensemble Mortality per i soggetti del training set.

OOB SURVIVAL FUNCTION

In Figura 5.5 sono riportate le curve della OOB Survival Function per i soggetti contenuti nel training set.

VIMP

La VIMP delle feature utilizzate per la predizione sul training set è riportata in Figura 5.6.

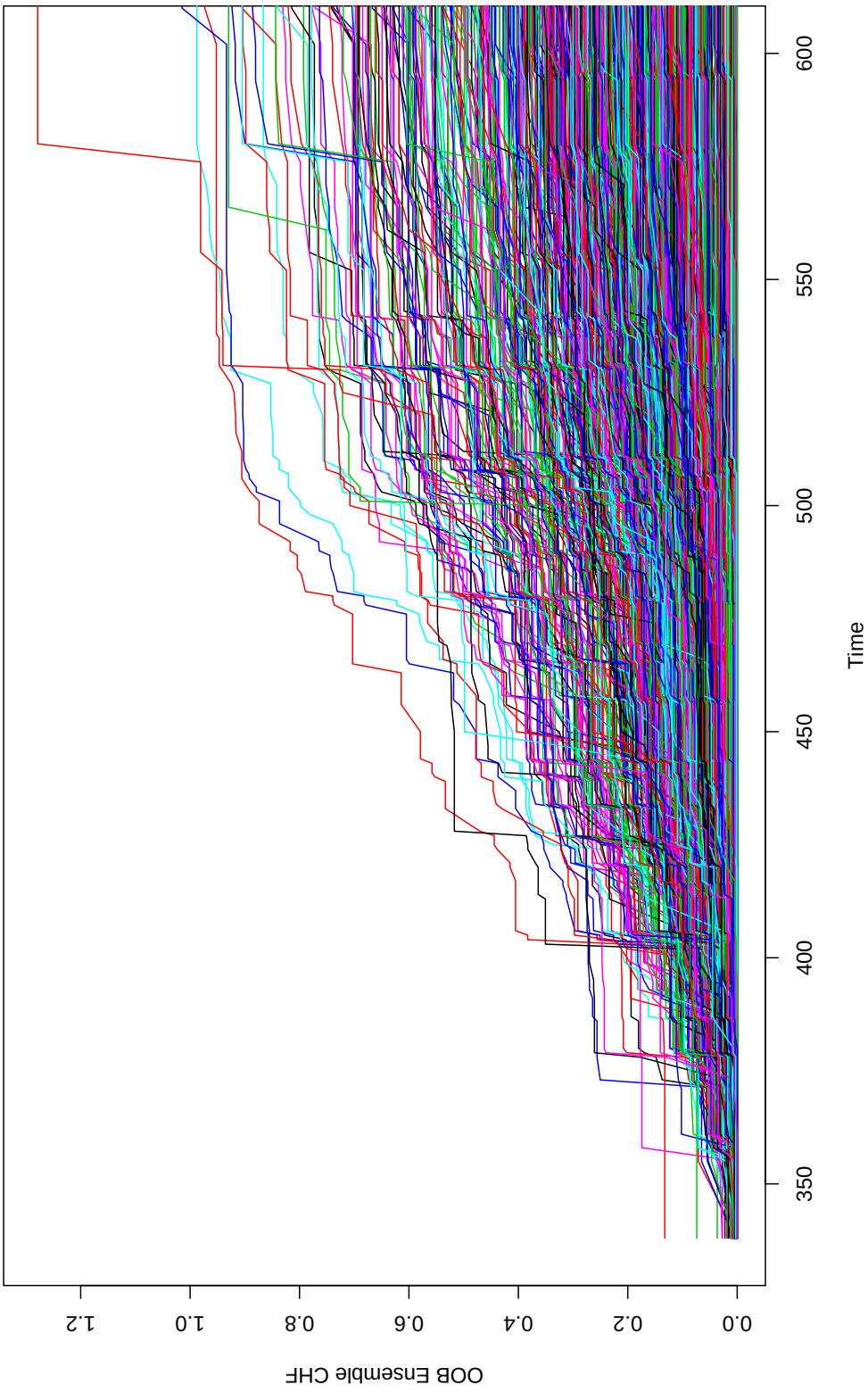

Figura 5.2: Training set - OOB Ensemble CHF.

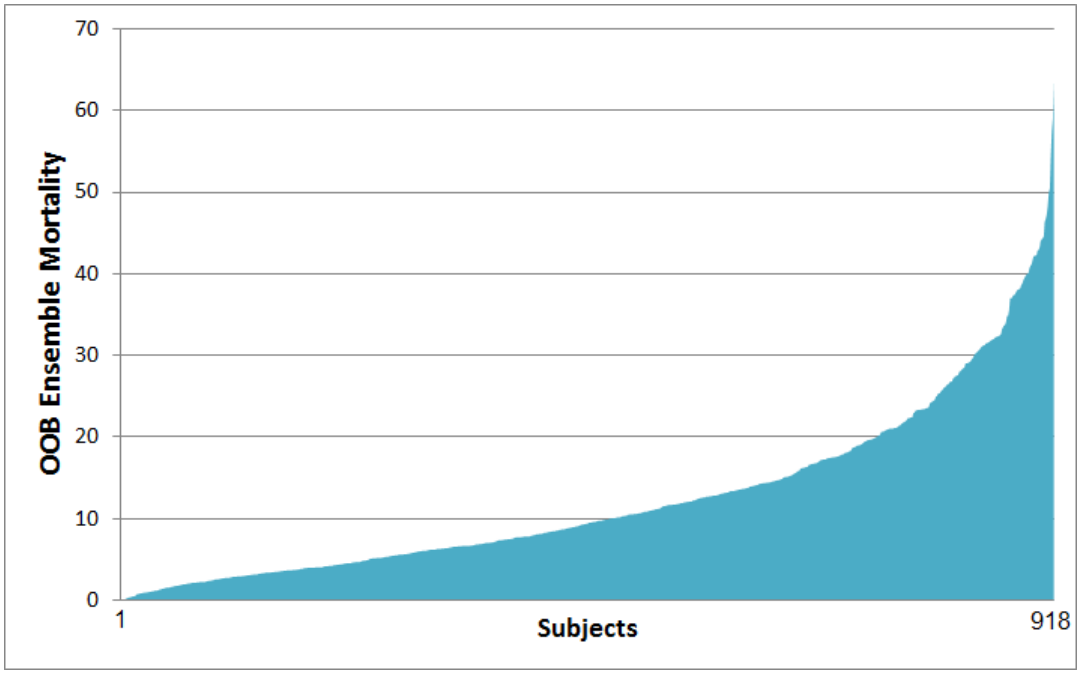

Figura 5.3: Training set - OOB Ensemble Mortality (ordine crescente) per ciascun soggetto.

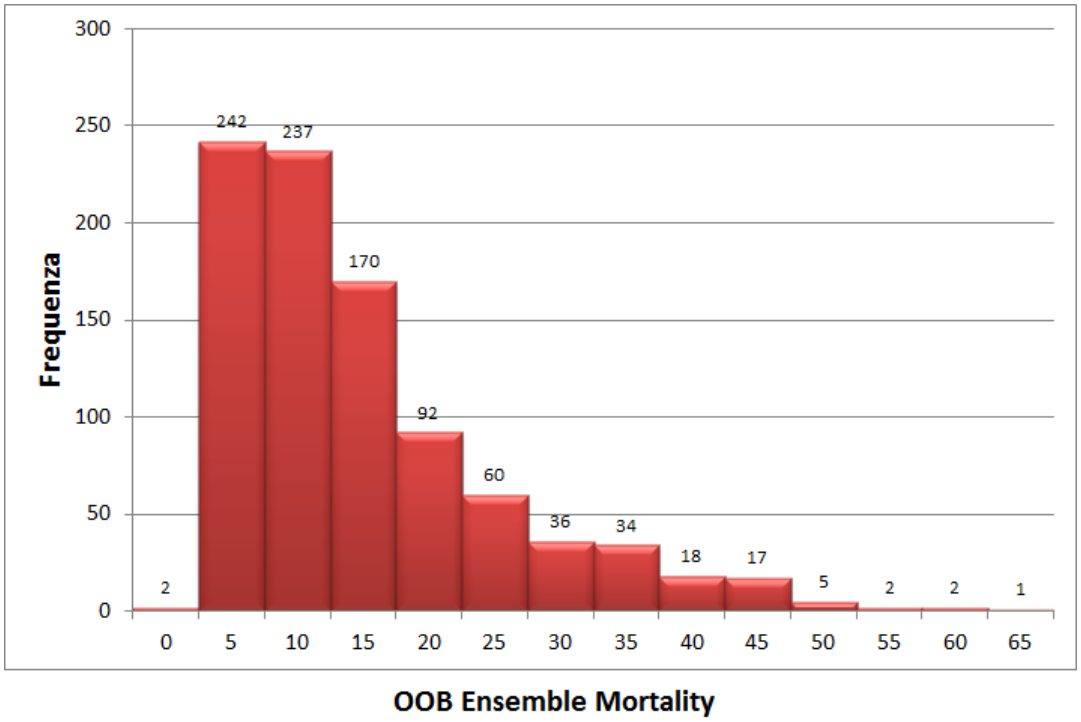

Figura 5.4: Training set - istogramma dell'OOB Ensemble Mortality.

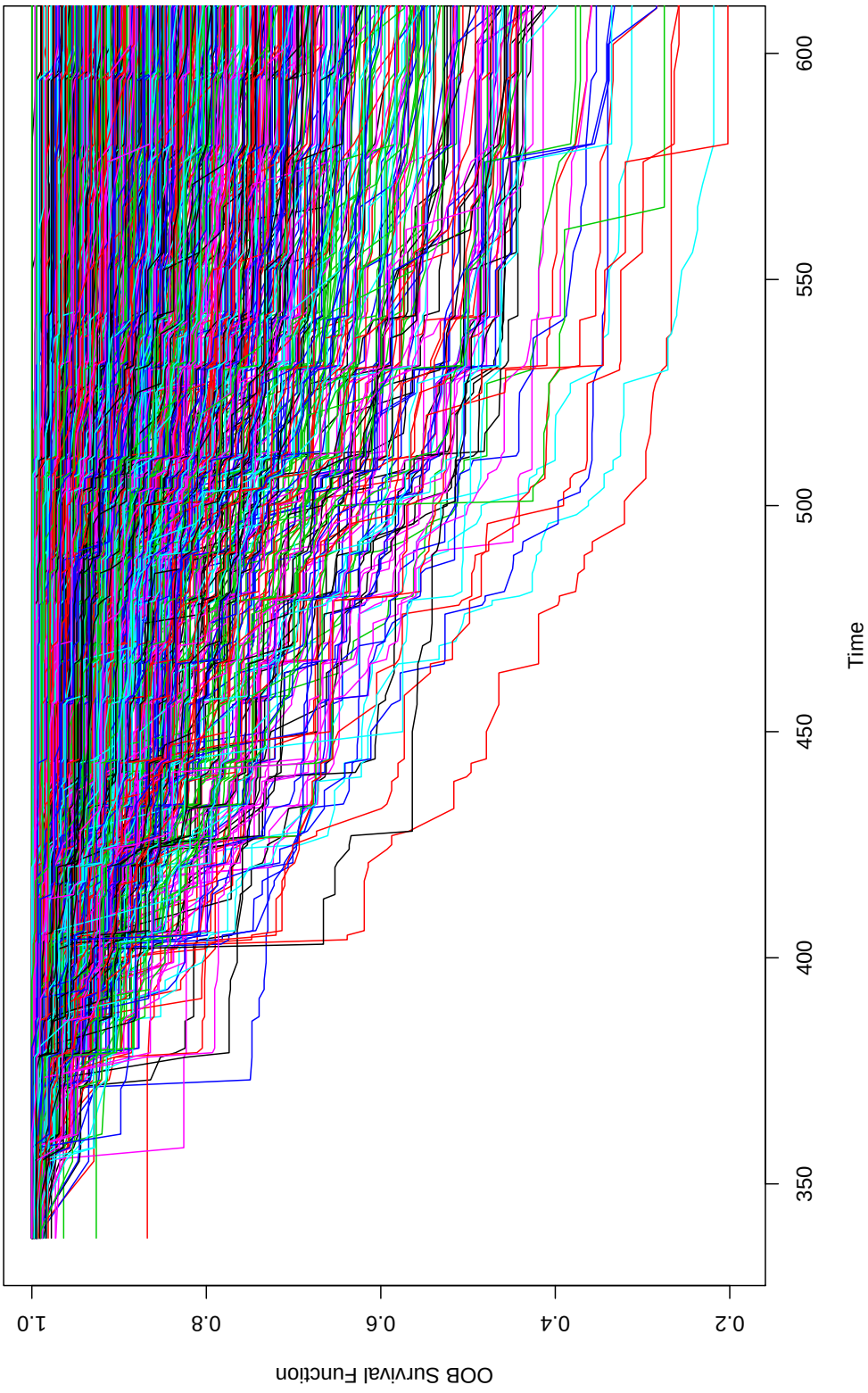

Figura 5.5: Training set - OOB Survival Function.

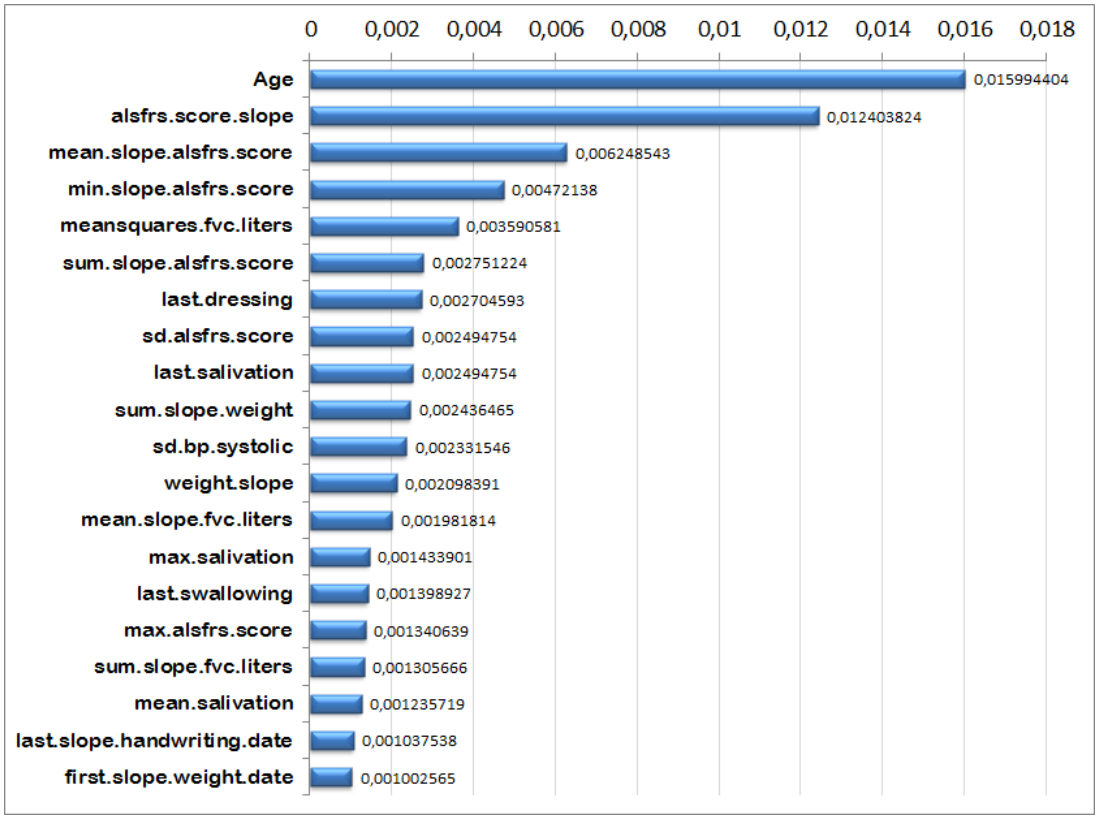

Figura 5.6: Training set - VIMP per le 20 feature più significative.

5.2 Test set

Le informazioni di riepilogo relative alla predizione sul dataset di test sono riportate in Tabella 5.2.

Sample size of test (predict) data:	279
Number of deaths in test data:	44
Number of grow trees:	500
Average no. of terminal nodes:	128.496
Total no. of variables:	237
Analysis:	RSF
Family:	surv
Test set Error rate:	30.52%

Tabella 5.2: Riepilogo della predizione della Foresta RSF sul dataset di test preprocessato.

PREDICTION ERROR

Dal momento che sono note le informazioni survival del test set, benché non utilizzate per la predizione, esse possono essere sfruttate per calcolare il Prediction Error della Foresta, che per questo dataset è pari al 30.52%.

ENSEMBLE CHF

In Figura 5.7 sono riportate le curve Ensemble CHF per i soggetti contenuti nel test set.

ENSEMBLE MORTALITY

In Figura 5.8 sono riportati, in ordine crescente, i valori della Ensemble Mortality per i 279 soggetti del test set.

Per meglio interpretarne la distribuzione, in Figura 5.9 è riportato l'istogramma dei valori assunti dalla Ensemble Mortality per i soggetti del test set.

SURVIVAL FUNCTION

In Figura 5.10 sono riportate le curve della Survival Function per i soggetti contenuti nel test set.

VIMP

La VIMP delle feature utilizzate per la predizione sul test set è riportata in Figura 5.11.

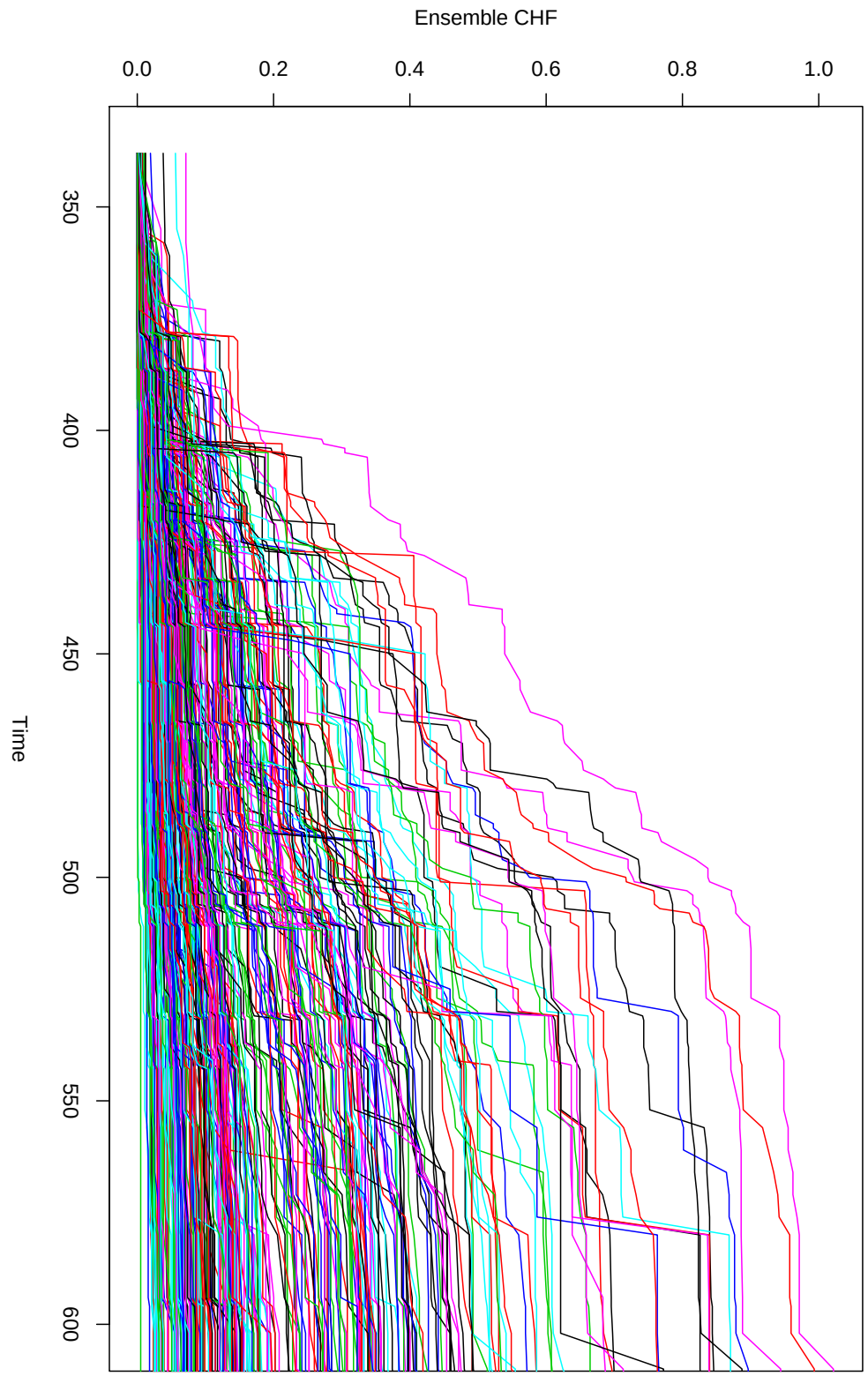

Figura 5.7: Test set - Ensemble CHF.

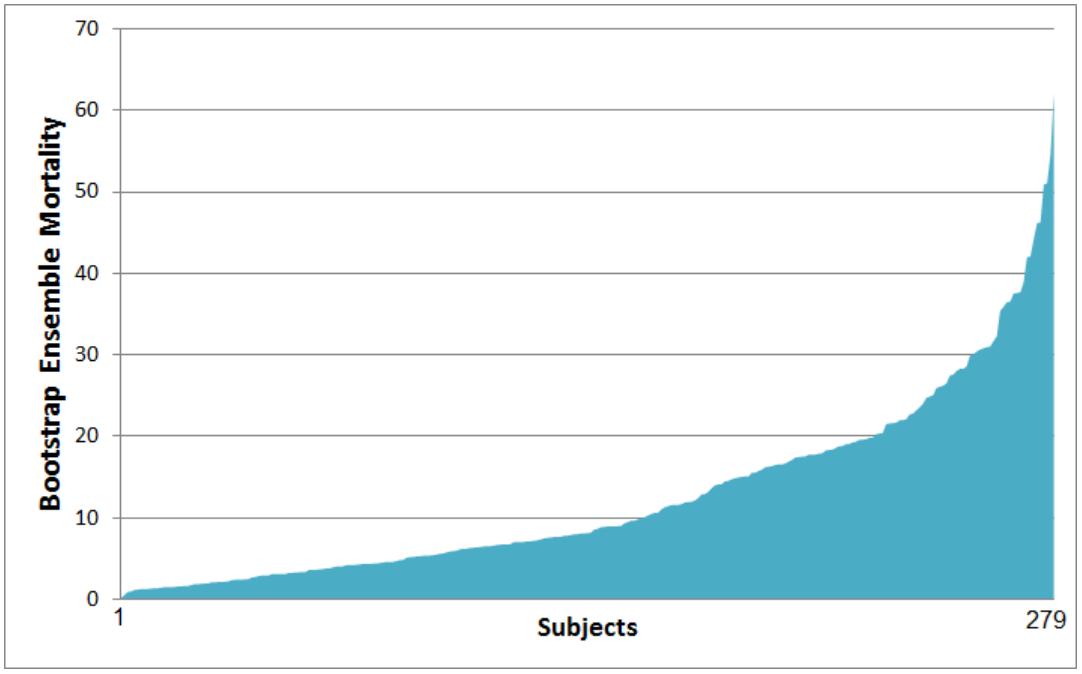

Figura 5.8: Test set - Ensemble Mortality (in ordine crescente) per ciascun soggetto.

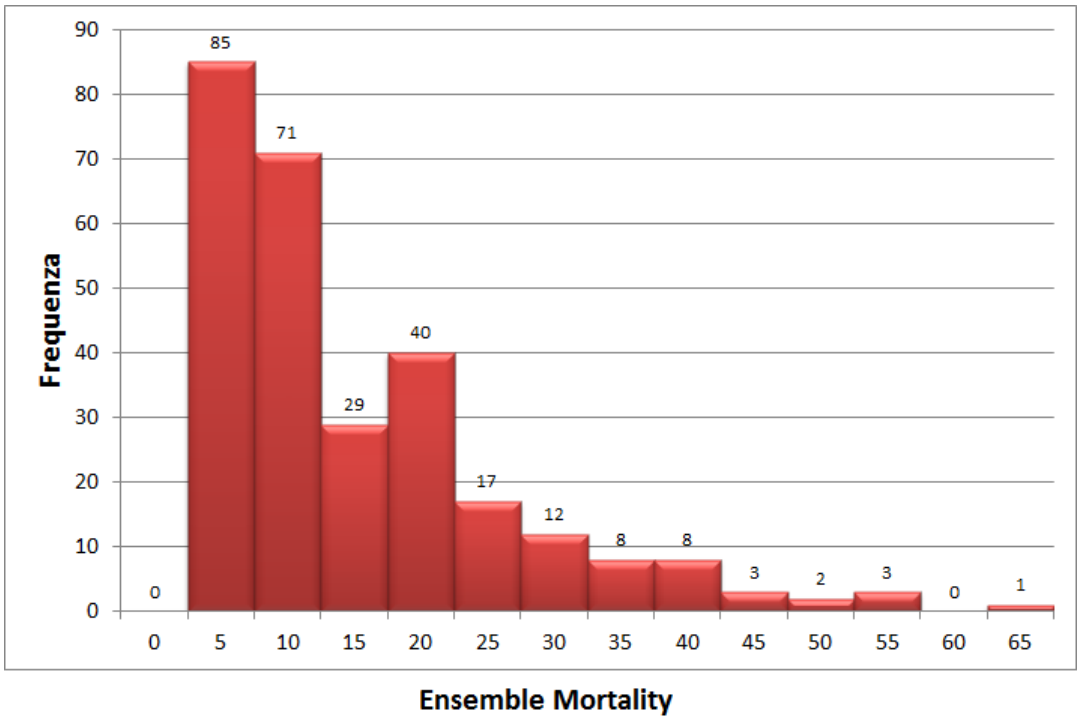

Figura 5.9: Test set - istogramma della Ensemble Mortality.

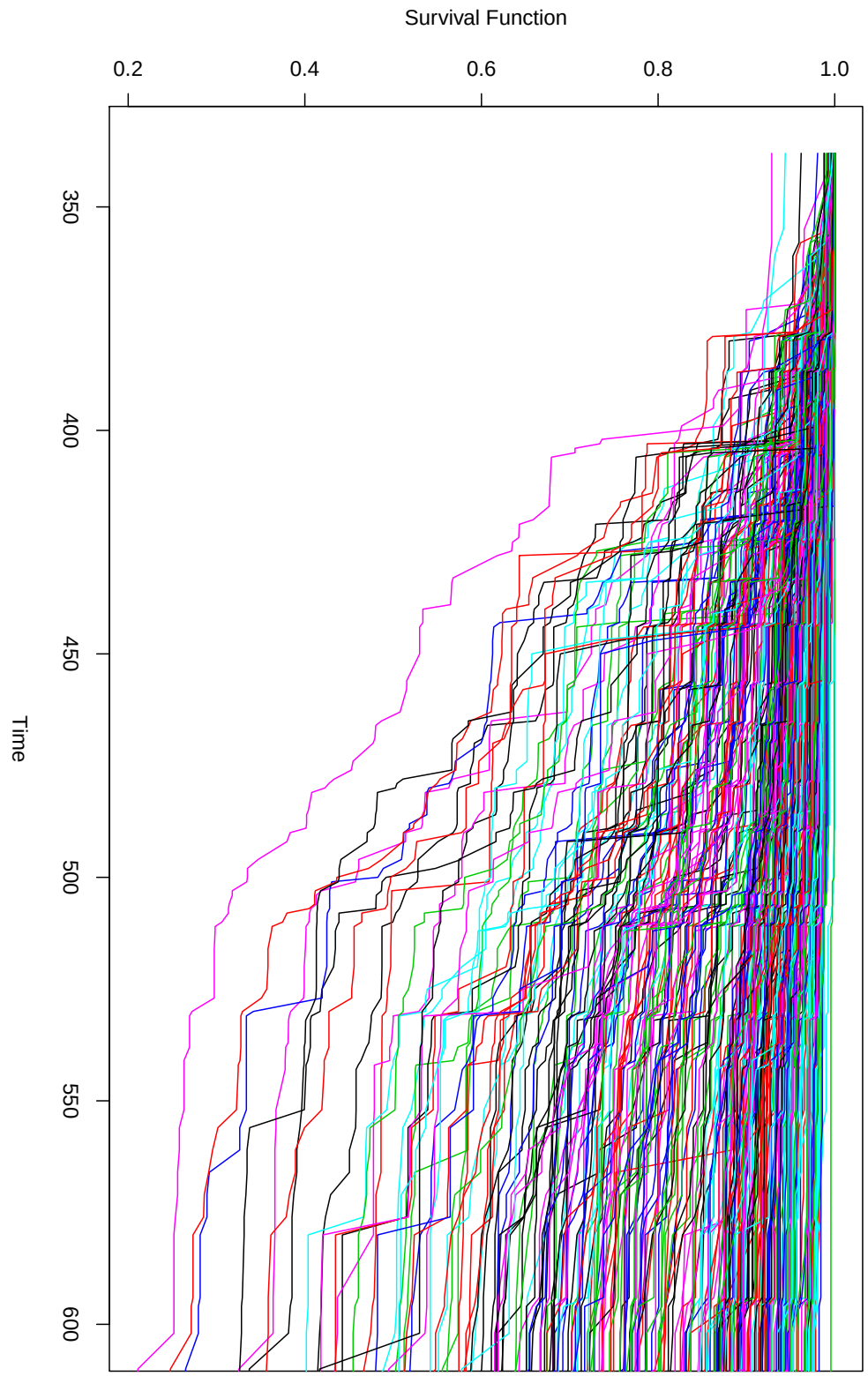

Figura 5.10: Test set - Survival Function.

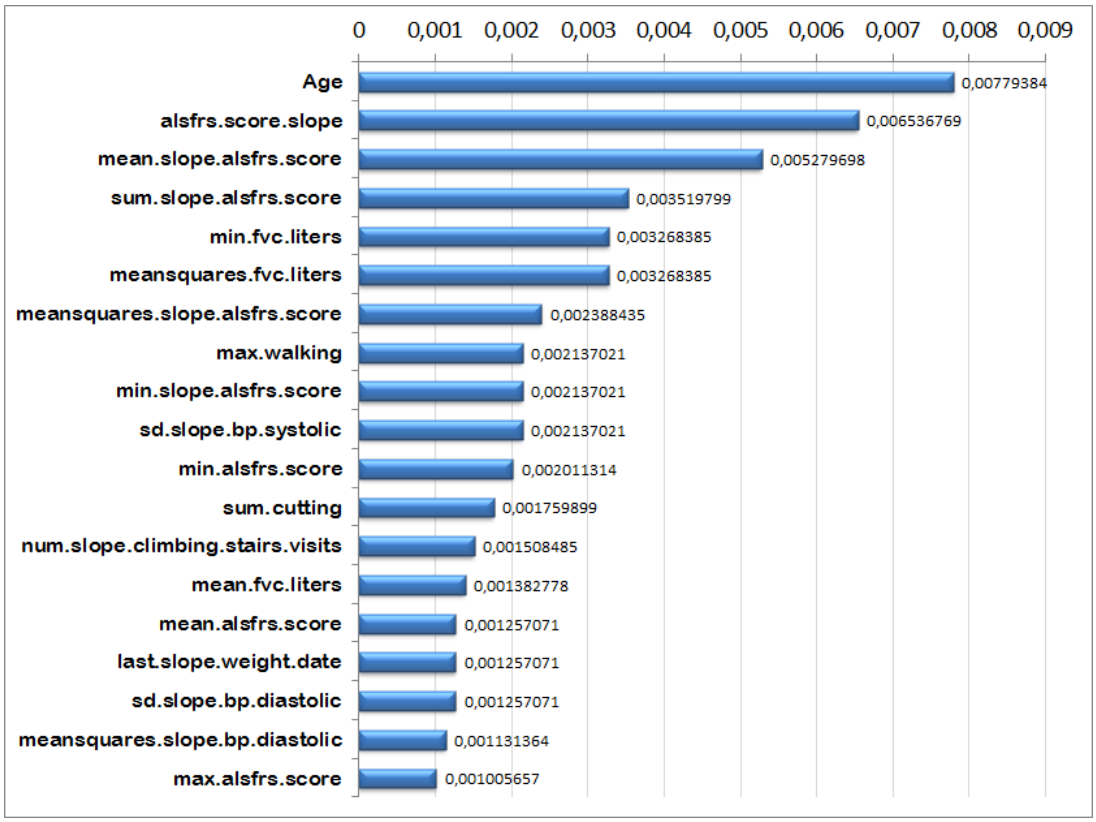

Figura 5.11: Test set - VIMP per le 20 feature più significative.

5.3 Validation set

Le informazioni di riepilogo relative alla predizione sul dataset di validazione sono riportate in Tabella 5.3.

Sample size of validation (predict) data:	625
Number of deaths in validation data:	95
Number of grow trees:	500
Average no. of terminal nodes:	128.496
Total no. of variables:	237
Analysis:	RSF
Family:	surv
Validation set Error rate:	41.88%

Tabella 5.3: Riepilogo della predizione della Foresta RSF sul dataset di validazione preprocessato.

PREDICTION ERROR

Anche per il dataset di validazione sono disponibili le informazioni survival, ed è pertanto possibile calcolare l'Errore della Foresta nella predizione.
Si osserva, in questo caso, un Prediction Error più elevato (41.88%) rispetto sia al training che al test set.

ENSEMBLE CHF

In Figura 5.12 sono riportate le curve Ensemble CHF per i soggetti contenuti nel validation set.

ENSEMBLE MORTALITY

In Figura 5.13 sono riportati, in ordine crescente, i valori della Ensemble Mortality per i 625 soggetti del validation set.

Per meglio interpretarne la distribuzione, in Figura 5.14 è riportato l'istogramma dei valori assunti dalla Ensemble Mortality per i soggetti del validation set.

SURVIVAL FUNCTION

In Figura 5.15 sono riportate le curve della Survival Function per i soggetti contenuti nel validation set.

VIMP

La VIMP delle feature utilizzate per la predizione sul validation set è riportata in Figura 5.16.

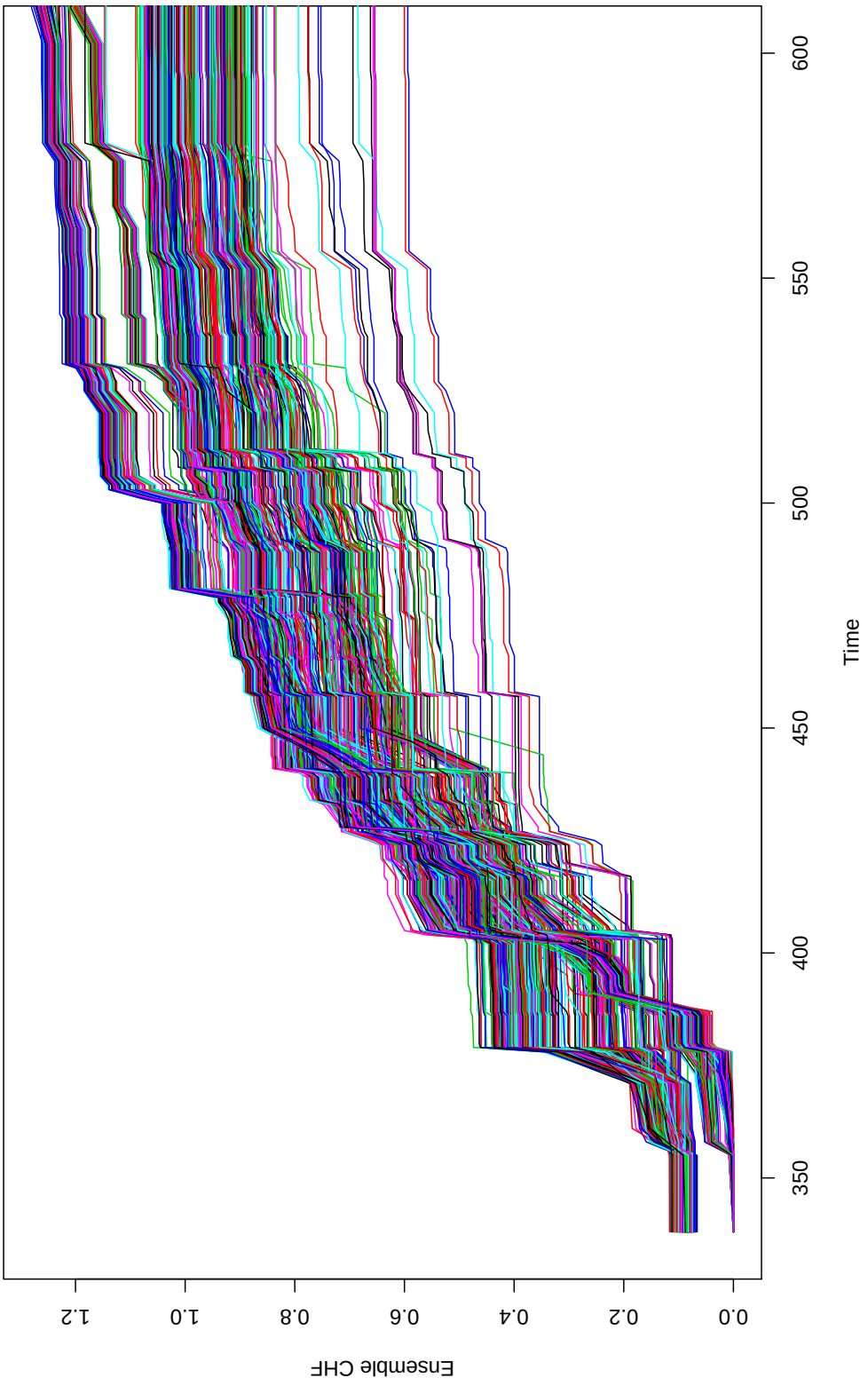

Figura 5.12: Validation set - Ensemble CHF.

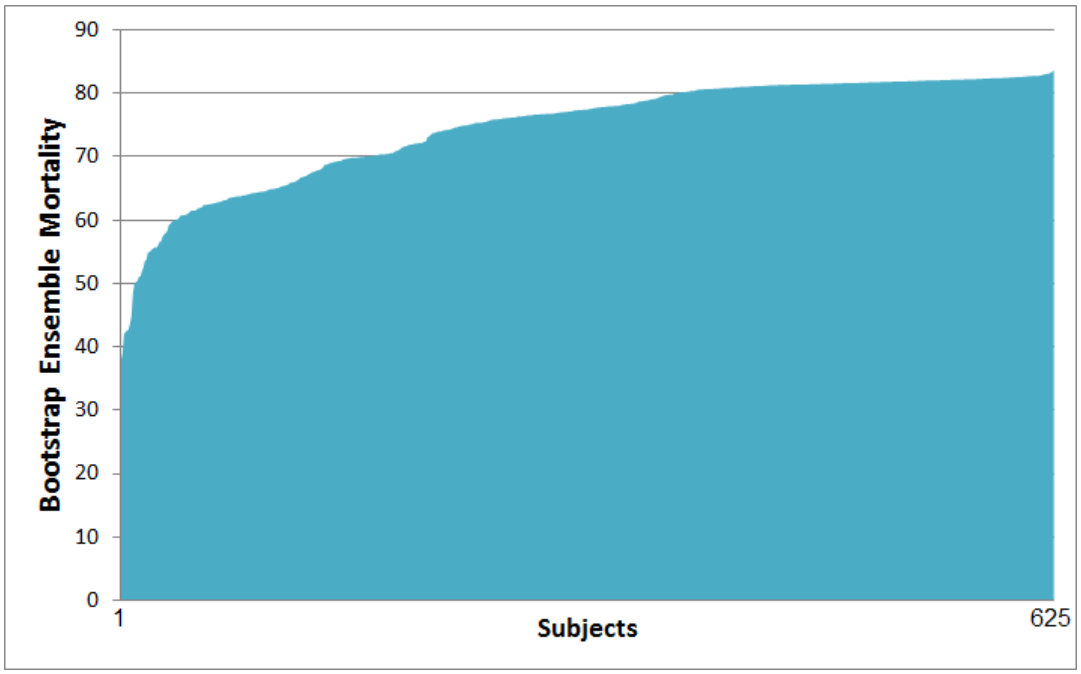

Figura 5.13: Validation set - Ensemble Mortality (in ordine crescente) per ciascun soggetto.

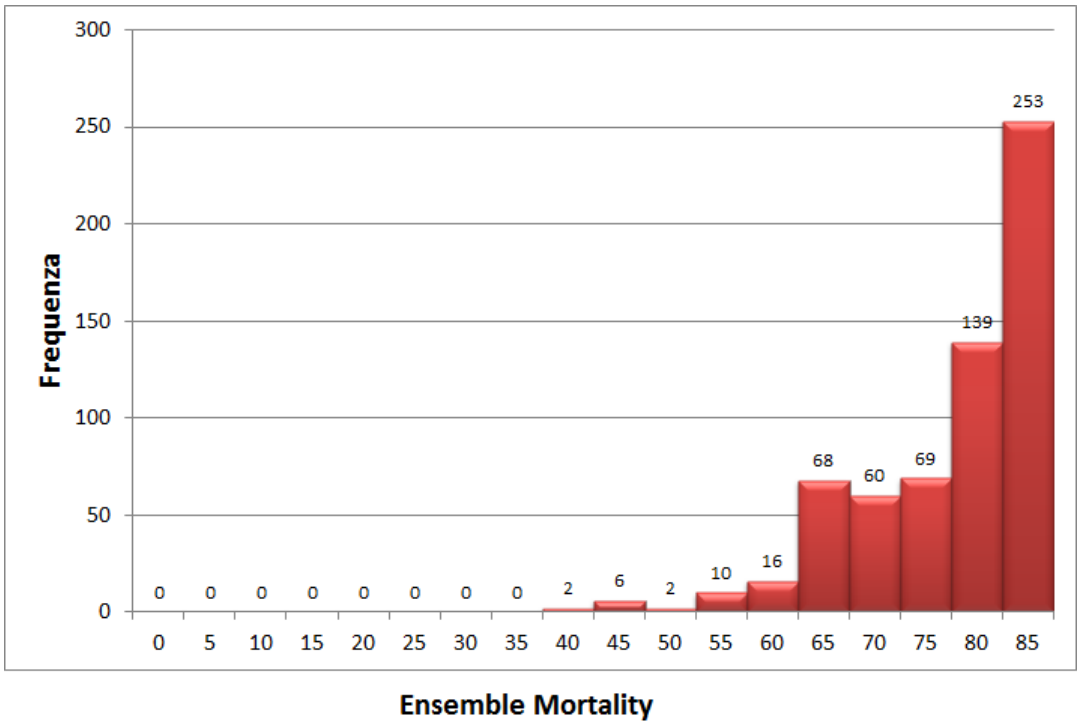

Figura 5.14: Validation set - istogramma della Ensemble Mortality.

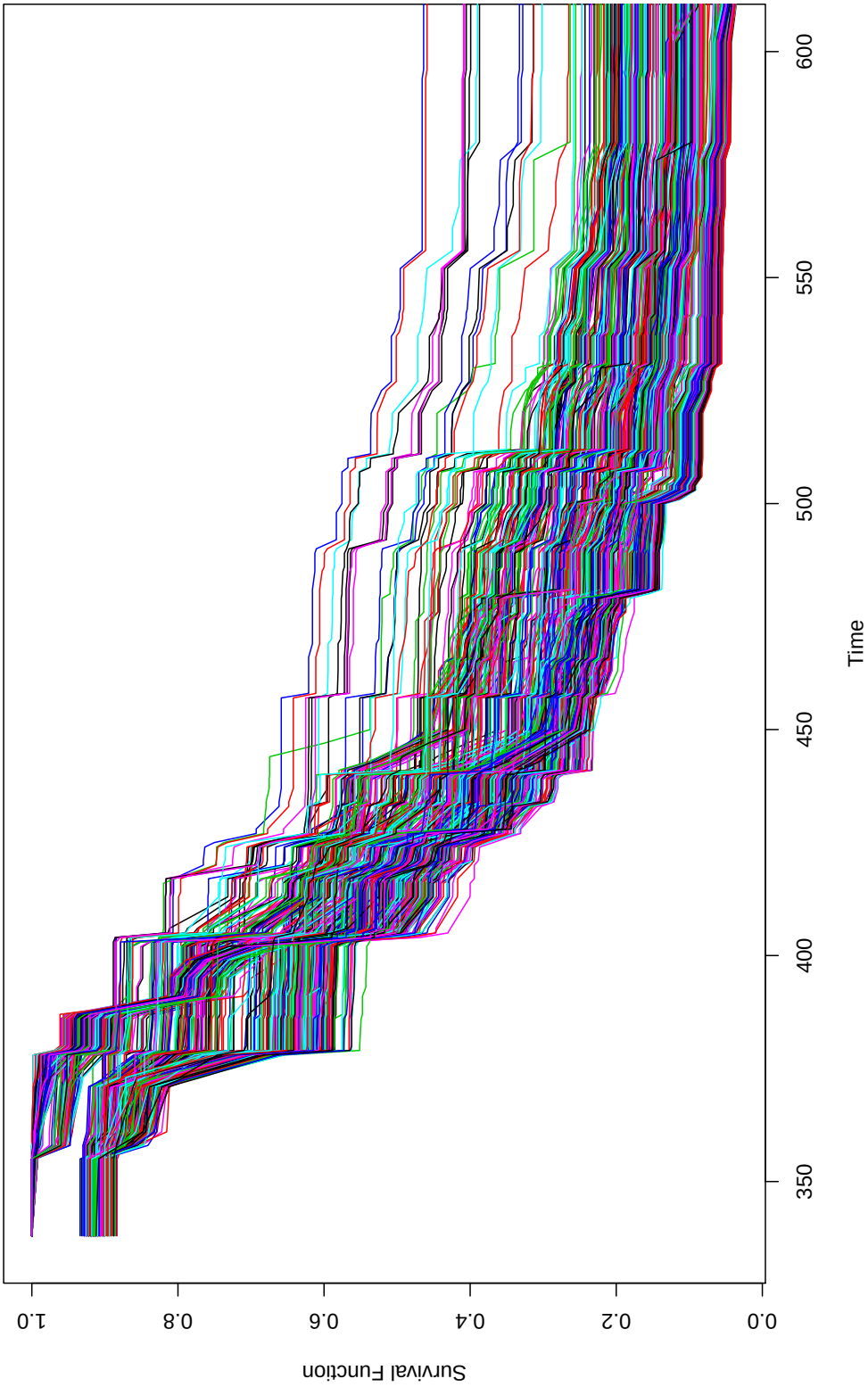

Figura 5.15: Validation set - Survival Function.

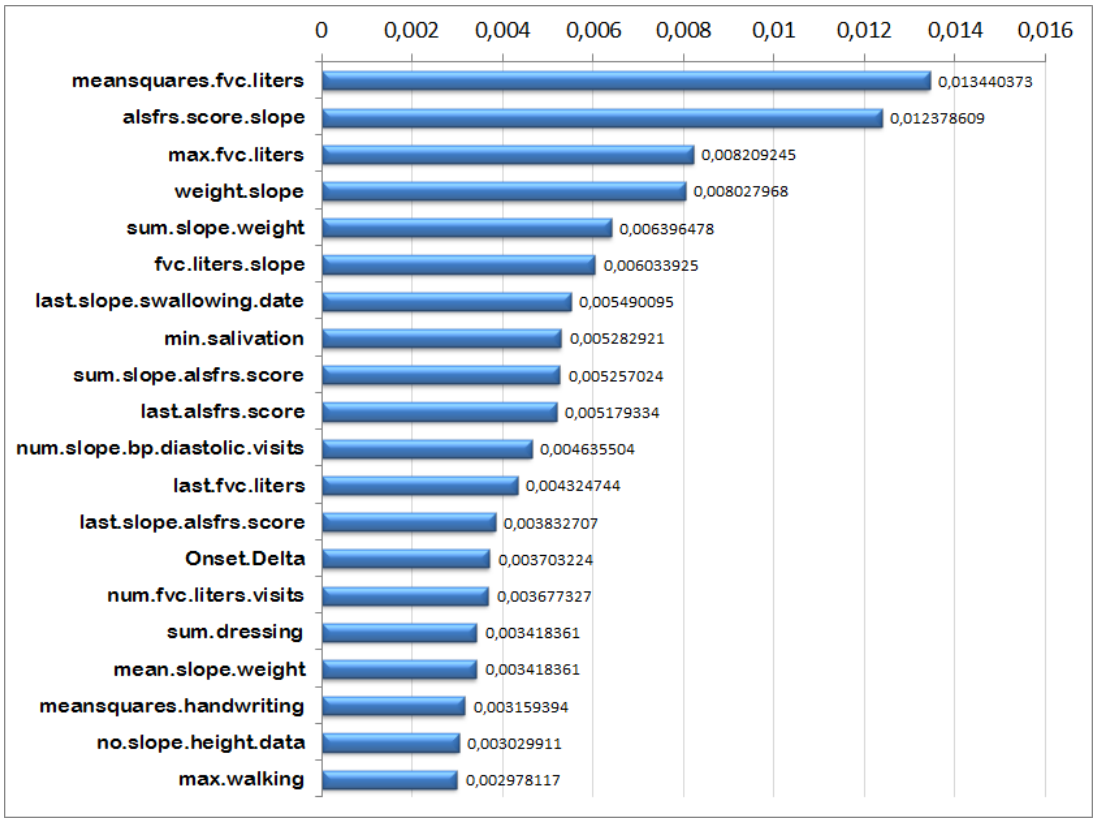

Figura 5.16: Validation set - VIMP per le 20 feature più significative.

5.4 Confronto e commento dei risultati

PREDICTION ERROR

Le performance della predizione della Foresta sono influenzate dal death rate del dataset utilizzato nella sua costruzione, ovvero dalla percentuale di soggetti nel training per cui si è registrato l'evento morte sul totale [25]. Infatti, maggiore è la percentuale di morti nel training set, migliore è la predizione survival, più accurato è il CHF e, di conseguenza, minore è il Prediction Error.

Nel caso specifico, il training set presenta death rate pari a $137/918 = 14.92\%$, una percentuale abbastanza bassa, che genera infatti una Foresta che, applicata al training set stesso, porta ad un Errore di Predizione (OOB) abbastanza elevato, pari al: 34.73%.

La variabilità nell'Errore di Predizione misurato per il test set (30.52%) ed il validation set (41.88%) è da imputarsi ai diversi valori delle feature fornite per la predizione, evidentemente più favorevoli (in termini di facilità di predizione con la Foresta ottenuta) per il dataset di test, meno per quello di validazione.

ENSEMBLE CHF

Le predizioni Ensemble CHF trovano, per la loro stessa natura, interpretazione a livello di singolo paziente. Proprio per questo motivo, si le si può intendere come caratterizzanti per il decorso della malattia e, quindi, pensare di utilizzarle per stratificare i pazienti sulla base del rischio di morte predetto. Un approccio simile viene applicato, nel Capitolo 6, alle curve Survival.

Dal confronto tra i tre dataset, è interessante osservare che, globalmente, mentre le curve per i pazienti di training e test set assumono distribuzione simile (Figure 5.2 e 5.7), quelle del dataset di validazione (Figura 5.12) mostrano andamenti meno vari. In particolare, si osserva come, per il validation set, in tempi relativamente brevi tutti i pazienti presentino rischio di morte più elevato: le curve salgono velocemente, fino a raggiungere plateau intorno e oltre $CHF = 0.5$.

A livello di predizione, si può pensare che sia questa sovrastima del rischio di morte a causare un Errore di Predizione per il validation set così elevato: è infatti probabile che, nel confronto dello status con la predizione, risultino predetti ad alto rischio pazienti per cui lo stato è invece censored.

Si nota, inoltre, che il grafico dell'Ensemble CHF per il dataset di validazione presenta molte curve sovrapposte. Se confrontato, per esempio, con il grafico corrispondente per il test set, si osserva come sia molto ridotto per il validation l'effetto "spaghetti plot", nonostante il pool di soggetti analizzati sia oltre due volte quello del test.
La sovrapposizione è da leggersi come la caduta di molti soggetti nelle stesse foglie terminali della Foresta: è da ipotizzare quindi, che, date le regole di split della Foresta costruita sul training set, le feature del validation set assumano valori che ne determinano percorsi analoghi dentro gli Alberi.

ENSEMBLE MORTALITY

Come diretta conseguenza dei trend complessivi osservati per l'Ensemble CHF, si osservano distribuzioni dell'Ensemble Mortality simili per i dataset di training e validazione, riportate negli istogrammi di Figure 5.4 e 5.9: la maggior parte dei soggetti ha Ensemble Mortality inferiore a 40, con un picco in valori intorno a 5-15.

Per il dataset di validazione, invece, si nota un andamento praticamente opposto: nessuno dei 625 soggetti presenta Ensemble Mortality al di sotto di 35, ed il picco della distribuzione si osserva in corrispondenza dell'intervallo di valori più elevati, compresi tra 80 e 85.

SURVIVAL FUNCTION

Vista la relazione riportata nell'Equazione 5.2 che lega Ensemble CHF e Survival Function, ci si aspettano considerazioni analoghe nel confronto tra le curve di sopravvivenza del training e del test set e quelle del validation set. Anche con questa tipologia di predizione, infatti, si osserva meno sparsità negli andamenti della Survival Function per il dataset di validazione e, in maniera opposta rispetto al CHF, un brusco calo delle curve già nella prima metà dei time of interest considerati.

Al pari delle curve Ensemble CHF, le funzioni Survival ottenute con il metodo RSF rapprensentano una caratteristica ricavata per ciascun paziente. A partire da queste, si sono applicati metodi di stratificazione basati sul Clustering, che vengono presentati nel Capitolo 6.

VIMP

Le prime 20 feature più predittive per ciascuno dei tre dataset utilizzati lungo la Foresta sono riportate nella Tabella 5.4.

Dal confronto, si può notare come ci siano alcune categorie di feature comuni ai tre dataset che si rivelano importanti per la predizione tramite la Foresta creata.

È il caso, ad esempio, delle feature estratte dai punteggi *ALSFRS*, complessivi o riguardanti aree funzionali specifiche. Compare anche l'informazione riguardante la variazione di *peso* del soggetto che, subendo con la malattia una progressiva dimunuzione, risulta probabilmente indicatore significativo dello stato di rischio. Si osserva, inoltre, ricorrenza anche nelle misurazioni relative alla *Capacità Vitale Forzata* (FVC), che riporta un'informazione utile rispetto alla funzionalità respiratoria del paziente.
La VIMP della variabile *Age*, nonostante questa non sia presente nelle prime venti feature ordinate per il validation set, assume i valori massimi per training e test.

Le feature così identificate trovano corrispondenza in letteratura [30, 15, 44], dove compaiono: età, sito di onset, genere, pendenza della progressione della malattia, pendenza dello score ALSFRS, punteggi FVC.

training	**test**	**validation**
Age	Age	meansquares.fvc.liters
alsfrs.score.slope	alsfrs.score.slope	alsfrs.score.slope
mean.slope.alsfrs.score	mean.slope.alsfrs.score	max.fvc.liters
min.slope.alsfrs.score	sum.slope.alsfrs.score	weight.slope
meansquares.fvc.liters	min.fvc.liters	sum.slope.weight
sum.slope.alsfrs.score	meansquares.fvc.liters	fvc.liters.slope
last.dressing	meansquares.slope.alsfrs.score	last.slope.swallowing.date
sd.alsfrs.score	max.walking	min.salivation
last.salivation	min.slope.alsfrs.score	sum.slope.alsfrs.score
sum.slope.weight	sd.slope.bp.systolic	last.alsfrs.score
sd.bp.systolic	min.alsfrs.score	num.slope.bp.diastolic.visits
weight.slope	sum.cutting	last.fvc.liters
mean.slope.fvc.liters	num.slope.climbing.stairs.visits	last.slope.alsfrs.score
max.salivation	mean.fvc.liters	Onset.Delta
last.swallowing	mean.alsfrs.score	num.fvc.liters.visits
max.alsfrs.score	last.slope.weight.date	sum.dressing
sum.slope.fvc.liters	sd.slope.bp.diastolic	mean.slope.weight
mean.salivation	meansquares.slope.bp.diastolic	meansquares.handwriting
last.slope.handwriting.date	max.alsfrs.score	no.slope.height.data
first.slope.weight.date	min.turning	max.walking

Tabella 5.4: Riepilogo delle 20 feature più predittive per i dataset di training, test e validation.

È inoltre attualmente in corso di studio la capacità predittiva del peso del paziente [35, 36, 38], che nell'analisi svolta in questo lavoro sembra contribuire alla bontà della definizione della prognosi.

La corrispondenza delle feature individuate in questa analisi con quelle presenti in letteratura permette di affermare la validità dell'algoritmo utilizzato. Ulteriori approfondimenti sono necessari per validare il potenziale di queste variabili e chiarire il loro ruolo nella fisiopatologia della SLA.

Capitolo 6

Stratificazione

L'utilizzo del metodo Random Survival Forests sui dataset contenenti dati clinici e demografici dei pazienti SLA ha permesso di ottenere, per ciascun soggetto, delle curve individuali, descrittive dell'aspettativa della prognosi del paziente: l'Ensemble CHF fornisce, nel tempo, il rischio di morte del paziente, mentre la Survival Function ne descrive la probabilità di sopravvivenza.

A partire da queste curve, si possono progettare studi di stratificazione, per individuare l'eventuale presenza di sottogruppi di pazienti omogenei dal punto di vista del rischio di morte o, equivalentemente, omogenei per quanto riguarda la loro probabilità di sopravvivenza.

Si è scelto, in questo lavoro, di impostare una prima analisi di stratificazione utilizzando le Survival Function ottenute per i soggetti del training set.

Le curve Survival sono state raggruppate tra loro sfruttando due metodi di classificazione non supervisionata: il *Clustering Gerarchico Agglomerativo* ed il *Clustering K-means*.

Si tratta di tecniche che, basandosi sulle informazioni disponibili per ciascun soggetto (e quindi, nello specifico, sulla Survival Function), si propongono di aggregare le istanze in gruppi (o *cluster*) sulla base della loro somiglianza reciproca.

Nelle prossime Sezioni, vengono introdotti dal punto di vista teorico gli algoritmi. In seguito, si descrive la loro applicazione alle Survival Function del training set, illustrando le scelte implementative adottate. Infine, vengono presentati e discussi i risultati ottenuti.

6.1 Clustering Gerarchico Agglomerativo

Il Clustering Gerarchico Agglomerativo è una tecnica utilizzata in Machine Learning, che permette di ottenere progressivi raggruppamenti di istanze.

Esso opera con un approccio "*bottom-up*" (dal basso verso l'alto), procedendo iterativamente nell'aggregazione di sempre più istanze in sempre meno sottogruppi.

Si riporta di seguito l'algoritmo nel dettaglio, nel caso di istanze numeriche.

- Si definisce inizialmente una *metrica* che quantifica la distanza tra coppie di istanze.
- Al primo passo, ciascuna istanza costituisce un cluster distinto.
  Vengono quindi calcolate le distanze reciproche tra tutte le possibili coppie di cluster.
- Ad ogni passo, l'algoritmo individua i due cluster più vicini e li fonde insieme in un nuovo cluster.
  Vengono quindi ricalcolate le distanze reciproche tra cluster, utilizzando un opportuno *criterio di linkage* basato sulla metrica di distanza scelta.
- Procedendo iterativamente, si ottiene, da ultimo, un unico cluster che raggruppa tutte le istanze.

A livello visivo, l'aggregazione porta alla costruzione di un Albero Binario, come quello riportato in Figura 6.1, detto *dendrogramma*.

Ogni livello dell'albero rappresenta una partizione delle istanze.

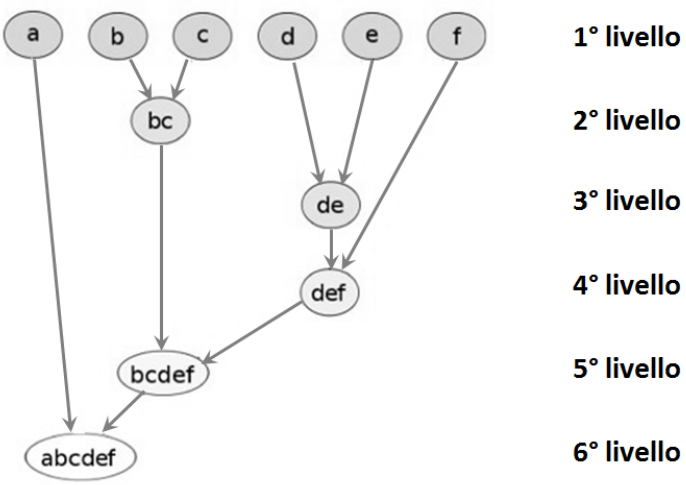

Figura 6.1: Clustering Gerarchico Agglomerativo.

6.1.1 Misure di distanza o similarità

Per definire la distanza o similarità tra due istanze, si fissa un'opportuna metrica.

Tale scelta influenza la forma dei cluster, poiché alcuni elementi possono risultare più "vicini" utilizzando una metrica e più "lontani" utilizzandone un'altra.

Prese due istanze generiche numeriche x e y, definite ciascuna come vettore di M elementi (si pensi per esempio ai valori della Survival Function nei vari time of interest), le metriche più comunemente applicate nel Clustering Gerarchico sono:

- **Distanza euclidea**:
$$d(x,y) = \sqrt{\sum_{i=1}^{M} (x_i - y_i)^2}$$

- **Distanza di Manhattan**:
$$d(x,y) = \sum_{i=1}^{M} |x_i - y_i|$$

- **Distanza di Minkowsky**:
$$d(x,y) = \left[\sum_{i=1}^{M} |x_i - y_i|^p\right]^{\frac{1}{p}}$$

6.1.2 Criteri di linkage

Nella formazione dei cluster, si procede iterativamente fondendo insieme, ad ogni passo, la coppia di cluster più vicina. Dal momento che ciascun cluster è costituito in generale da più istanze, è necessario definire le modalità di calcolo della distanza tra cluster. Si definisce pertanto un criterio di linkage, che definisce la distanza tra due cluster come funzione delle distanze tra le istanze contenute nei cluster stessi.

Analogamente alla scelta della metrica della distanza, anche la scelta del criterio di linkage comporta potenzialmente diverse suddivisioni, e bisogna pertanto fare attenzione ad applicare il criterio più opportuno per i dati in analisi.

In generale, i criteri più utilizzati per definire la distanza tra cluster sono:

- **Nearest Neighbor o Single Linkage (SL)**
  La distanza tra i cluster X e Y è la distanza minima tra istanze appartenenti ai due cluster:
$$d(X,Y) = min\{d(\forall x \in X, \forall y \in Y)\}$$

- **Fartherst Neighbor o Complete Linkage (CL)**
  La distanza tra i cluster X e Y è la distanza massima tra istanze appartenenti ai due cluster:
$$d(X,Y) = max\{d(\forall x \in X, \forall y \in Y)\}$$

- **Average Linkage (AL)**
  La distanza tra i cluster X e Y è la media delle distanze tra le singole istanze dei due cluster:
  $$d(X,Y) = \frac{1}{|X||Y|} \sum_{x \in X, y \in Y} d(x,y)$$

- **Distanza di Ward**
  Al generico passo k, le N istanze analizzate sono suddivise in $N-k$ cluster distinti.

  Si definisce per ciascun cluster il suo *centroide* m, ottenuto come media delle istanze in esso contenute.
  Il centroide m_r per l'r-esimo cluster X_r, costituito da n_r istanze, sarà quindi:
  $$m_r = \frac{1}{n_r} \sum_{x_i \in X_r} x_i.$$

  Per il cluster X_r si può calcolare la *varianza intra-cluster*, ovvero la somma dei quadrati delle distanze di ciascuna delle istanze in essa contenute rispetto al centroide, normalizzata:
  $$s_r^2 = \frac{\sum_{x_i \in X_r} \parallel x_i - m_r \parallel^2}{n_r - 1}.$$
  Si definisce la somma delle varianze intra-cluster su tutti i cluster, al passo k, come:
  $$E_k \doteq \sum_{r=1}^{N-k} s_r^2.$$

  Si considerano quindi tutte le possibili aggregazioni di coppie di cluster. Per ciascuna, si calcola la somma delle varianze intra-cluster al passo successivo E_{k+1}.

  Si definisce la *funzione obiettivo* ΔE_{k+1}, detta *distanza di Ward* [45], come:
  $$\Delta E_{k+1} = E_{k+1} - E_k,$$

  che rappresenta come varierebbe la somma delle varianze intra-cluster con la fusione dei due cluster considerati.

  Dal momento che, all'aumentare del numero di cluster, si vuole aumentare il meno possibile la varianza intra-cluster (che rispecchia l'efficacia della suddivisione nel raggruppare istanze simili), l'aggregazione migliore al passo k-esimo sarà quella che minimizza la funzione obiettivo.

6.1.3 Scelta del numero di cluster

L'applicazione del metodo di Clustering Gerarchico Agglomerativo presenta un punto chiave: il numero di cluster risultanti, e la rispettiva assegnazione delle istanze, dipende dal livello a cui si osserva l'albero gerarchico di output (vedi Figura 6.2).

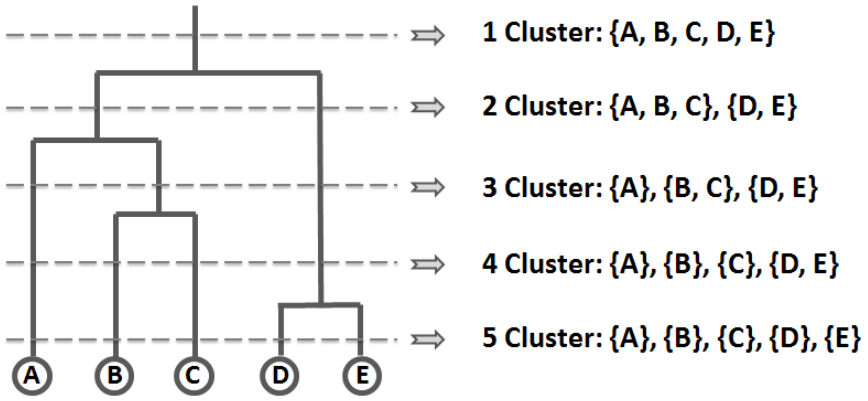

Figura 6.2: Esempi di taglio del dendrogramma ottenuto col Clustering Gerarchico Agglomerativo.

Per rendere fruibili i risultati di questo metodo, pertanto, è necessario compiere una scelta sul numero di cluster.

Alcune indicazioni sul numero ottimo si possono ottenere tramite:

- **conoscenze *a priori***, nel caso in cui si abbiano già informazioni sul sistema studiato e si possa quindi stimare il numero di sottogruppi atteso (se è noto, per esempio, il numero delle possibili forme di una patologia);

- un'**analisi della varianza** dei cluster: dal momento che il clustering ottimo minimizza la varianza intra-cluster VAR_{intra} e massimizza la varianza inter-cluster VAR_{inter} (ovvero vengono raggruppate insieme istanze simili, e divise istanze diverse), si può scegliere il numero di cluster che porta ad una suddivisione tale da minimizzare, secondo un criterio di parsimonia, il rapporto VAR_{intra}/VAR_{inter};

- l'implementazione di una **Analisi alle Componenti Principali** (PCA, Principal Component Analysis) sulle istanze analizzate, che fornisce una stima della reale dimensione dei dati.

Tra i vantaggi del Clustering Gerarchico Agglomerativo c'è sicuramente la facilità di implementazione del metodo e il basso costo dal punto di vista computazionale, nonostante sia necessario tenere conto delle esigenze di RAM per memorizzare le misure di similarità tra le istanze ad ogni passo. Inoltre, questa tecnica fornisce in output un dendrogramma di facile lettura, per la costruzione del quale non è necessario fissare a priori il numero di cluster.

Da contro, fissate la metrica di distanza e il criterio di linkage, la struttura dell'albero risulta deterministica e rigida, dal momento che le fusioni degli elementi non sono revocabili in passi successivi. Inoltre, per poter sfruttare il risultato definendo le assegnazioni

delle istanze ai vari cluster, è richiesto all'utente di intervenire sul livello di osservazione del dendrogramma.

6.2 Clustering K-means

Il Clustering K-Means è un metodo che richiede *a priori* l'indicazione, da parte dell'utente, del numero K di cluster desiderato. A partire da questo, l'algoritmo compie iterativamente dei tentativi di assegnazione delle istanze ai cluster, andando ad ottimizzare una specifica funzione costo, come per esempio, la somma delle varianze intra-cluster VAR_{intra}, o il rapporto VAR_{intra}/VAR_{inter}.

L'algoritmo nel dettaglio si sviluppa come segue:

- l'utente seleziona il numero di cluster K e definisce una metrica di distanza tra le istanze (vedi Sezione 6.1.1);
- vengono selezionate in maniera casuale K istanze, che costituiscono al primo passo i centroidi dei K cluster;
- si calcola la distanza tra ciascuna istanza non ancora assegnata ed i K centroidi: ogni istanza viene quindi assegnata al cluster di centroide ad essa più vicino;
- si ricalcolano i centroidi per ciascun cluster, come media delle istanze in esso contenute;
- si procede iterativamente nell'assegnazione delle istanze al cluster di centroide più vicino e al ricalcolo dei centroidi, finché non si raggiunge una condizione di termine.

6.2.1 Condizioni di termine dell'algoritmo

La riassegnazione delle istanze ai cluster ed il ricalcolo dei centroidi si interrompe quando si verifica una delle seguenti condizioni:

· è stato raggiunto un numero massimo di iterazioni T prefissato;

· meno di P istanze sono state assegnate a nuovi cluster nell'ultima iterazione, con P prefissato;

· nessuna assegnazione delle istanze a cluster diversi riduce ulteriormente la funzione costo

Il metodo K-means permette di fare clustering su dataset di dimensioni maggiori rispetto agli approcci gerarchici, rivelandosi poco oneroso e senza il rischio di incorrere in problemi di spazio nella memorizzazione delle distanze, dal momento che queste vengono sovrascritte, e non conservate, ad ogni passo. Inoltre, si dimostra un metodo flessibile dal

punto di vista dell'assegnazione delle variabili, che possono essere riallocate dopo essere state assegnate ad un cluster, fino a raggiungere la collocazione ottima nella partizione.

Da contro, richiede che il numero di cluster venga fissato *a priori*, implicando conoscenze da parte dell'utente che non sono sempre disponibili. Come visto per il Clustering Gerarchico nella Sezione 6.1.3, alcune indicazioni possono essere ricavate dalla letteratura del sistema in esame, o calcolate tramite PCA.

Un'ulteriore criticità del metodo K-means è intrinsecamente legata alla scelta casuale delle istanze con cui inizializzare i centroidi, che può portare a soluzioni di clustering sub-ottime. Una soluzione, per ovviare a questo rischio, può essere iterare l'algoritmo più volte con diverse assegnazioni iniziali dei centroidi e selezionare, tra le partizioni ottenute, quella che minimizza la funzione costo.

6.3 Implementazione dei metodi di clustering

A partire dalle Survival Function ottenute con `randomForestSRC` sul dataset di training preprocessato, si sono applicati i metodi di Clustering Gerarchico Agglomerativo e K-means per fare stratificazione sui pazienti.

6.3.1 Scelta del numero di cluster

Entrambi i metodi di clustering richiedono, seppur in fasi diverse, l'indicazione da parte dell'utente del numero di cluster da implementare.

Per gestire questa necessità, si è utilizzata la funzione `NbClust`, contenuta nell'omonimo pacchetto R [11], che determina in modo automatico il numero di cluster ottimo per i dati correnti.

`NbClust` richiede in ingresso: il dataset con le istanze da suddividere nei cluster, la metrica per definire la distanza tra le istanze, il metodo di clustering desiderato e l'intervallo di cluster da indagare.
Nello specifico, sia il Clustering Gerarchico che K-means sono stati implementati utilizzando la *distanza euclidea* come metrica di distanza. Per il Clustering Gerarchico si è inoltre scelta la *distanza di Ward* come criterio di linkage. Per entrambi i metodi, si è indagato l'intervallo di cluster compreso tra 2 e 10.

Per ciascun numero di cluster compreso nell'intervallo indicato, la funzione produce una partizione secondo il metodo scelto.
Vengono quindi calcolati alcuni indici di bontà sulla partizione così ottenuta. Nello specifico, per il clustering delle Survival Function sono stati calcolati 26 indici, per i dettagli dei quali si rimanda alla documentazione del pacchetto [11].

Dopo aver calcolato tutte le partizioni coi diversi numeri di cluster ed aver ottenuto gli indici corrispondenti, ciascun indice vota per il numero di cluster che ne ha mantenuto più basso il valore. Il numero di cluster che risulta più votato al termine di questa operazione viene quindi restituito come numero ottimo per il metodo scelto.

Come si può vedere rispettivamente nelle Figure 6.3 e 6.4, i valori ottimi del numero di cluster sono 2 per il Clustering Gerarchico e 3 per il Clustering K-means.

È interessante osservare come, nel caso del Clustering Gerarchico Agglomerativo, l'opzione di 4 cluster abbia ricevuto un totale di 7 voti, contro gli 8 della più votata.

Per l'analisi seguente, per semplicità si è mantenuto l'utilizzo dell'indice più votato, ma sarebbe interessante approfondire la natura degli indici per poter compiere una scelta più consapevole.

```
*******************************************************************
* Among all indices:
* 8 proposed 2 as the best number of clusters
* 3 proposed 3 as the best number of clusters
* 7 proposed 4 as the best number of clusters
* 1 proposed 7 as the best number of clusters
* 1 proposed 9 as the best number of clusters
* 1 proposed 10 as the best number of clusters

                 ***** Conclusion *****

* According to the majority rule, the best number of clusters is  2

*******************************************************************
```

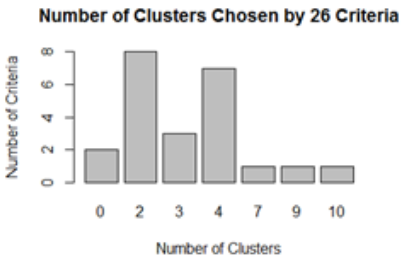

Figura 6.3: Individuazione del numero ottimo di cluster per il metodo di Clustering Gerarchico Agglomerativo tramite la funzione `NbClust`.

```
*******************************************************************
* Among all indices:
* 6 proposed 2 as the best number of clusters
* 9 proposed 3 as the best number of clusters
* 4 proposed 4 as the best number of clusters
* 1 proposed 5 as the best number of clusters
* 1 proposed 6 as the best number of clusters
* 1 proposed 7 as the best number of clusters
* 1 proposed 9 as the best number of clusters
* 1 proposed 10 as the best number of clusters

                 ***** Conclusion *****

* According to the majority rule, the best number of clusters is  3

*******************************************************************
```

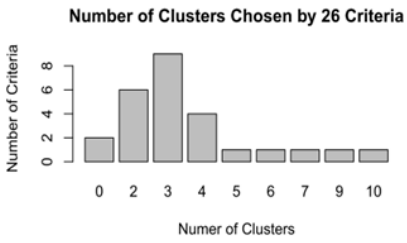

Figura 6.4: Individuazione del numero ottimo di cluster per il metodo di Clustering K-means tramite la funzione `NbClust`.

6.4 Analisi dei risultati

In Figura 6.5 è riportata la suddivisione del dendrogramma dei soggetti nei due cluster ottenuti tramite il Clustering Gerarchico Agglomerativo.

Dal momento che la funzione `NbClust` restituisce direttamente, per la migliore partizione individuata, un vettore con l'assegnazione delle istanze ai cluster, il dendrogramma

è stato ottenuto utilizzando la funzione classica per il Clustering Gerarchico `hclust`, mantenendo le scelte di *distanza euclidea* e *distanza di Ward*, rispettivamente come metrica di similarità e criterio di linkage.
Il dendrogramma, che riporta sull'asse orizzontale un'etichetta corrispondente a ciascun soggetto, è stato quindi tagliato al livello di 2 soli cluster, come indicato da `NbClust`.

Per visualizzare la suddivisione dei soggetti nei cluster, si è fatto uso della funzione `clusplot` contenuta nel pacchetto R `cluster`. Essa riporta le istanze sul piano definito dalle prime due Componenti Principali, che, nel caso del dataset in esame, spiegano il 91.3 % della variabilità dei dati.

In Figura 6.6 è riportata la suddivisione dei soggetti nei 2 cluster individuati col metodo Gerarchico Agglomerativo, mentre in Figura 6.7 è riportato il grafico analogo, nel caso della suddivisione nei 3 cluster ottenuti con K-means.

Confrontando le partizioni ottenute, ad una prima analisi si può giudicare migliore la soluzione proposta dal metodo Gerarchico Agglomerativo: a livello grafico, infatti, le istanze non mostrano suddivisioni nette e, nel caso K-means, i cluster presentano ampie aree sovrapposte, che lasciano ipotizzare la validità di un partizionamento in un numero di cluster minore.

Tramite l'applicazione dei metodi di Clustering Gerarchico Agglomerativo e Clustering K-means alle Survival Function ricavate con RSF, si è ottenuta quindi una suddivisione dei soggetti del training set rispettivamente in 2 e 3 sottogruppi.

Dal momento che si sono utilizzate tecniche di clustering non supervisionato, è difficile determinare se, dati i pazienti in input, l'aver individuato 2/3 sottogruppi sia un risultato solido. Nel caso in cui gli algoritmi si basassero sulla diversa velocità di progressione della malattia (o su una grandezza ad essa correlata), allora 2/3 sottogruppi sarebbero un risultato coerente con quanto noto sul decorso clinico della SLA. È tuttavia possibile che il clustering colga similarità più profonde, ed in tal caso sarebbe importante risalirvi.

Per approfondire i risultati di questo primo studio di stratificazione, potrebbe quindi essere interessante: (1) forzare entrambi gli algoritmi a dividere i soggetti nello stesso numero di cluster, e confrontarne le assegnazioni; (2) confrontare le feature relative ai soggetti che vengono assegnati ad uno stesso cluster, per cercare di ricostruire cosa ne determini assegnazioni simili. A tal fine, si potrebbe implementare un metodo di classificazione che, basandosi sulle feature demografiche e cliniche in input, predica l'assegnazione dei pazienti ai diversi cluster. Da un'analisi delle feature più predittive, si potrebbero individuare le variabili determinanti nella suddivisione dei pazienti nei sottogruppi.

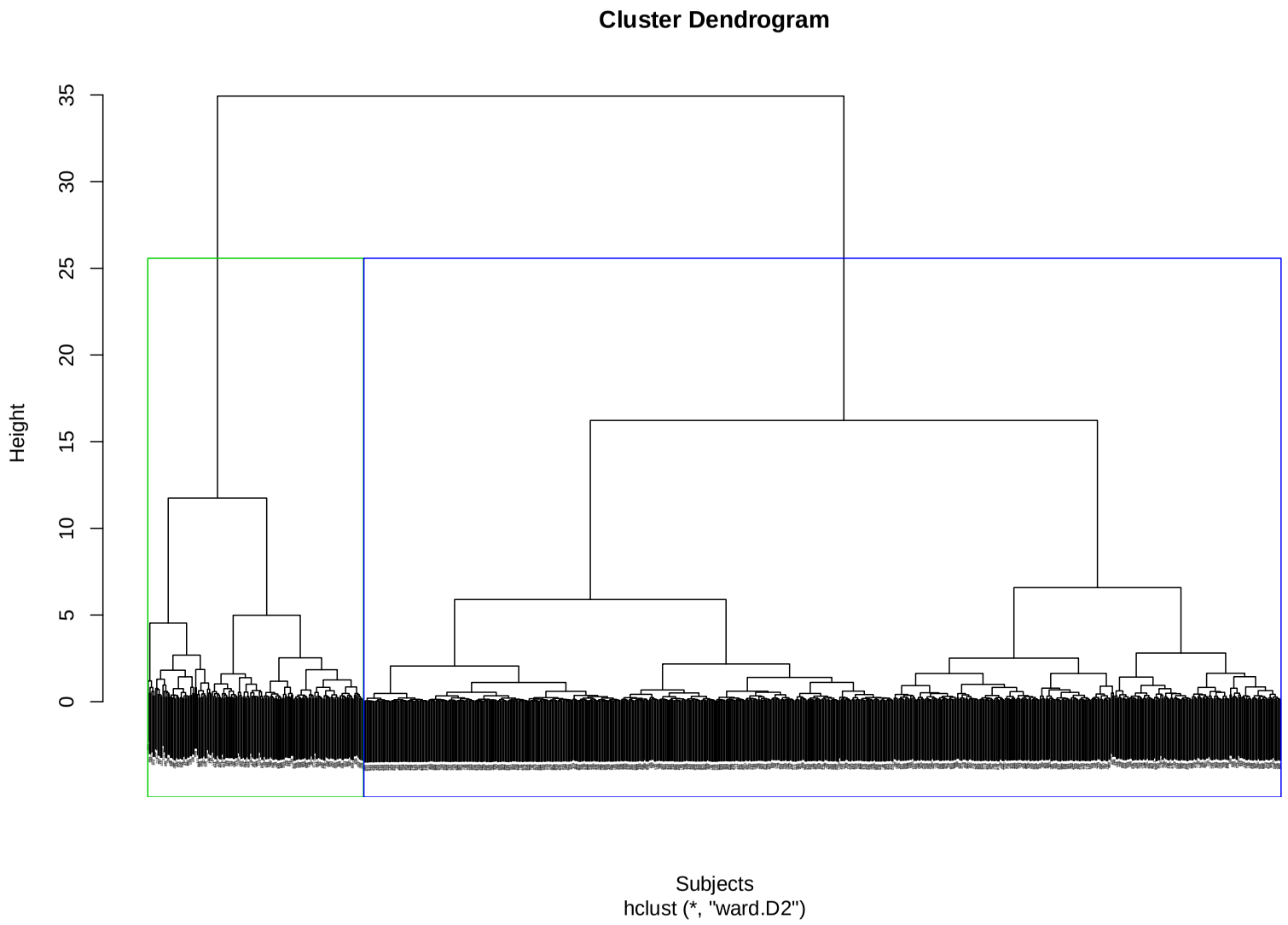

Figura 6.5: Clustering Agglomerativo Gerarchico sulle Survival Function del training set: partizione delle istanze in 2 cluster.

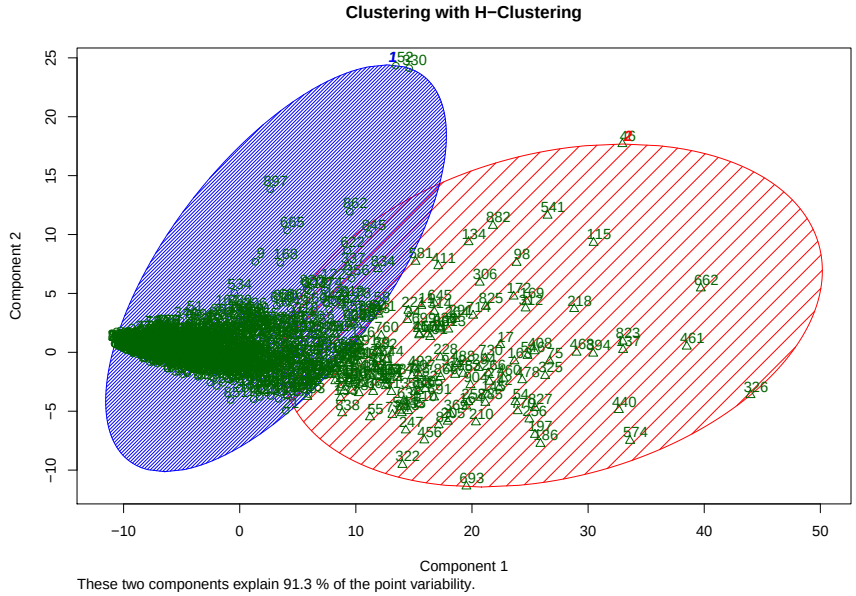

Figura 6.6: Clustering Gerarchico Agglomerativo sulle Survival Function del training set: partizione delle istanze in 2 cluster, riportate sul piano delle prime due Componenti Principali.

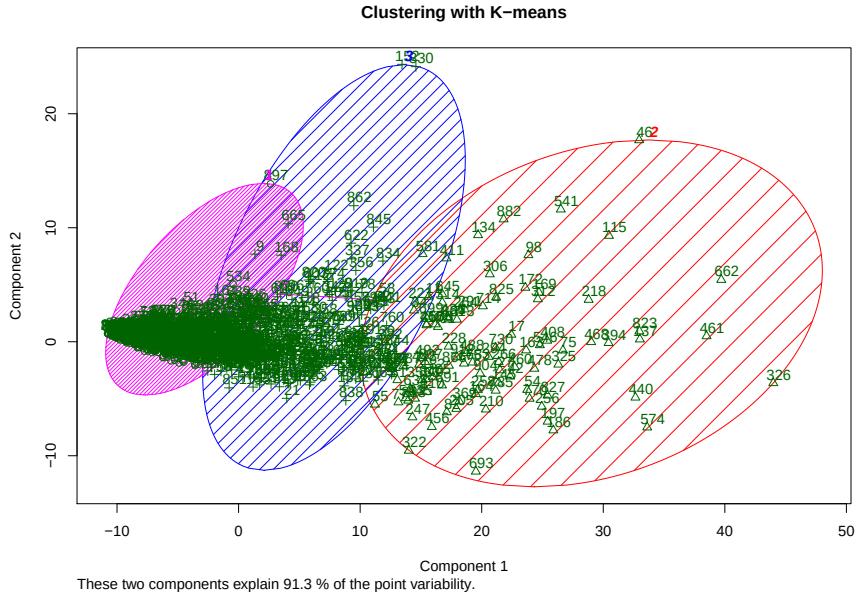

Figura 6.7: Clustering K-means sulle Survival Function del training set: partizione delle istanze in 3 cluster, riportate sul piano delle prime due Componenti Principali.

Capitolo 7

Conclusioni e sviluppi futuri

In questo lavoro di tesi si sono analizzati i dati demografici e clinici, estratti dal database PRO-ACT, di 1822 pazienti affetti da Sclerosi Laterale Amiotrofica (SLA), allo scopo di condurre uno studio di stratificazione.

Nell'ottica di individuare pazienti simili dal punto di vista della progressione della patologia, si è proceduto in due passi: (1) inizialmente, ai dati opportunamente preprocessati si è applicato il metodo di Analisi Survival *Random Survival Forests* (RSF) per estrarre le curve di rischio di morte e probabilità di sopravvivenza nel tempo per ciascun paziente; (2) successivamente, si sono implementate metolodologie di clustering non supervisionato sulle curve di sopravvivenza, ottenendo l'aggregazione dei pazienti in sottogruppi caratteristici.

Un'attenzione particolare è stata richiesta nella fase di preprocessing dei dati, in cui si sono implementati diversi protocolli di importazione delle variabili e metodologie di imputazione dei missing value. L'estrazione delle variabili dinamiche misurate durante i primi 3 mesi di trial, unite alle variabili statiche e alle informazioni survival disponibili per i pazienti, ha permesso di ottenere dataset preprocessati in un formato adatto all'applicazione del metodo RSF.

In questo lavoro, si è potuto avere un primo riscontro della potenza del metodo RSF nell'analisi di dati survival: si sono ottenute, infatti, informazioni sulla progressione della malattia e sulla probabilità di sopravvivenza dei soggetti sfruttando pienamente i dati a disposizione, grazie alla capacità di RSF di utilizzare anche l'informazione contenuta nelle istanze censored.

Le performance del classificatore sono state valutate su tre subset di pazienti (training, test e validation set), ottenendo Errori di predizione compresi tra il 30.52% ed il 41.88%. Tali valori possono trovare giustificazione nel numero ridotto di soggetti contenuti nel subset di training utilizzato per il fit della Foresta, nonché nel basso valore del death rate del training set stesso.

Per quanto riguarda le feature più predittive individuate con RSF, si è riscontrata corrispondenza con quanto riportato in letteratura. Tale risultato avvalora la bontà del

metodo RSF per l'analisi di questa tipologia di dati.
Sarebbe interessante analizzare le possibilità di impiego di queste feature in termini di affinamento del predittore: si potrebbe pensare, ad esempio, di fornire in input al classificatore solo le feature più significative e studiare l'impatto di questa scelta in termini di performance. Inoltre, si potrebbe approfondire il tipo di contributo che queste variabili possono apportare a livello clinico, nello studio della fisiopatologia della malattia o nell'uso delle stesse come biomarcatori prognostici.

L'uso dei metodi di Clustering Gerarchico Agglomerativo e K-means ha permesso di suddividere i soggetti rispettivamente in 2 e 3 sottogruppi, simili dal punto di vista della probabilità di sopravvivenza.

Posto quindi che un partizionamento sulla base del decorso clinico stimato è realizzabile, sarebbe necessario approfondire la natura delle partizioni ottenute. Infatti, dal momento che i metodi applicati sono non supervisionati, ovvero non è nota *a priori* l'effettiva differenza survival tra i pazienti, è necessario condurre ulteriori studi per ricostruire il legame tra le feature misurate nei soggetti e le assegnazioni ai cluster.
A tal fine, si potrebbe pensare di utilizzare le assegnazioni ottenute con questa analisi come target predittivo per un nuovo passo di classificazione: in questo modo, ricavando le feature maggiormente significative per la predizione, si dovrebbe riuscire ad individuare su cosa si differenzino i sottogruppi di pazienti, dal punto di vista demografico e/o clinico.

Complessivamente, questo lavoro di tesi ha dimostrato le potenzialità del metodo Random Survival Forests nell'Analisi Survival di dati relativi a pazienti affetti da Sclerosi Laterale Amiotrofica.

Sarebbe interessante estendere l'applicazione di questo classificatore a dataset più numerosi, sfruttando per esempio la nuova versione di PRO-ACT, composta attualmente dai dati di oltre 10700 soggetti, per verificarne in maniera più robusta le performance.

RSF, inoltre, si è rivelato un metodo efficiente, non eccessivamente rigido dal punto di vista della stuttura del dataset richiesto in input (per esempio, non è richiesta una normalizzazione sullo stesso range delle variabili), nonché potenzialmente in grado di gestire in modo autonomo problematiche tipiche delle collezioni di grandi dati (tra cui la presenza di missing value).

Per quanto riguarda la stratificazione basata su clustering, ulteriori analisi sono necessarie per arrivare ad un risultato chiaro dal punto di vista biologico e fruibile da parte dell'operatore clinico, ma l'implementazione dei due metodi svolta in questo lavoro lascia intravedere buone prospettive.

Continuando lungo la strada di ricerca intrapresa con questo lavoro di tesi, si spera di poter contribuire a far luce su questa patologia, restituendo alla ricerca clinica risultati chiari e fruibili.

La potenziale individuazione di marcatori prognostici, osservati nei pazienti idealmente nelle prime fasi della malattia, permetterebbe di ottenere indicazioni sul decorso clinico atteso. Tale informazione avrebbe ripercussioni positive in termini di supporto sia

alla programmazione del percorso terapeutico più adeguato, sia alla pianificazione delle risorse necessarie al paziente durante le varie fasi della malattia [5]. Infine, un contributo potrebbe essere dato anche alla pianificazione dei trial clinici ed all'interpretazione dei loro risultati, permettendo di confrontare il decorso clinico stimato per il paziente con quello riscontrato durante il trattamento sperimentale.

Bibliografia

[1] Associazione Italiana Sclerosi Laterale Amiotrofica - AISLA Onlus. `http://www.aisla.it`. Accessed: 2017-08-18.

[2] DREAM Challenges. `http://www.dreamchallenges.org`. Accessed: 2017-08-22.

[3] DREAM7: Phil Bowen ALS Prediction Prize4Life. `https://www.synapse.org/#!Synapse:syn2826267/wiki/71167`. Accessed: 2017-08-22.

[4] Pooled Resource Open-Access ALS Clinical Trials Database. `https://nctu.partners.org/ProACT/Home/Index`. Accessed: 2017-08-21.

[5] Ammar Al-Chalabi, Orla Hardiman, Matthew C Kiernan, Adriano Chiò, Benjamin Rix-Brooks, and Leonard H van den Berg. Amyotrophic lateral sclerosis: moving towards a new classification system. *The Lancet Neurology*, 15(11):1182–1194, 2016.

[6] Nazem Atassi, James Berry, Amy Shui, Neta Zach, Alexander Sherman, Ervin Sinani, Jason Walker, Igor Katsovskiy, David Schoenfeld, Merit Cudkowicz, et al. The pro-act database design, initial analyses, and predictive features. *Neurology*, 83(19):1719–1725, 2014.

[7] Leo Breiman. Out-of-bag estimation. *Technical report, Statistics Department, University of California Berkeley, Berkeley CA 94708, 1996b. 33, 34*, 1996.

[8] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001.

[9] B. R. Brooks, M. Sanjak, S. Ringel, J. England, J. Brinkmann, and M. Pestronk, A.and Charatan. The amyotrophic lateral sclerosis functional rating scale: Assessment of activities of daily living in patients with amyotrophic lateral sclerosis. *Archives of Neurology*, 53(2):141–147, 1996.

[10] Jesse M Cedarbaum, Nancy Stambler, Errol Malta, Cynthia Fuller, Dana Hilt, Barbara Thurmond, Arline Nakanishi, Bdnf Als Study Group, 1A complete listing of the BDNF Study Group, et al. The alsfrs-r: a revised als functional rating scale that incorporates assessments of respiratory function. *Journal of the neurological sciences*, 169(1):13–21, 1999.

[11] Malika Charrad, Nadia Ghazzali, Véronique Boiteau, and Azam Niknafs. NbClust: An R package for determining the relevant number of clusters in a data set. *Journal of Statistical Software*, 61(6):1–36, 2014.

[12] Adriano Chiò, Edward R Hammond, Gabriele Mora, Virginio Bonito, and Graziella Filippini. Development and evaluation of a clinical staging system for amyotrophic lateral sclerosis. *J Neurol Neurosurg Psychiatry*, pages jnnp–2013, 2013.

[13] TG Clark, MJ Bradburn, SB Love, and DG Altman. Survival analysis part i: basic concepts and first analyses. *British journal of cancer*, 89(2):232, 2003.

[14] Merit E Cudkowicz, Jon Katz, Dan H Moore, Gilmore O'neill, Jonathan D Glass, Hiroshi Mitsumoto, Stanley Appel, Bernard Ravina, Karl Kieburtz, Ira Shoulson, et al. Toward more efficient clinical trials for amyotrophic lateral sclerosis. *Amyotrophic Lateral Sclerosis*, 11(3):259–265, 2010.

[15] MA Del Aguila, WT Longstreth, V McGuire, TD Koepsell, and G Van Belle. Prognosis in amyotrophic lateral sclerosis a population-based study. *Neurology*, 60(5):813–819, 2003.

[16] Roxane Duroux and Erwan Scornet. Impact of subsampling and pruning on random forests. *arXiv preprint arXiv:1603.04261*, 2016.

[17] Ton Fang, Ahmad Al Khleifat, Daniel R Stahl, Claudia Lazo La Torre, Caroline Murphy, Uk-Mnd LicalS, Carolyn Young, Pamela J Shaw, P Nigel Leigh, and Ammar Al-Chalabi. Comparison of the king's and mitos staging systems for als. *Amyotrophic Lateral Sclerosis and Frontotemporal Degeneration*, 18(3-4):227–232, 2017. PMID: 28054828.

[18] Matilde Francescon. Analisi delle relazioni tra variabili demografiche e cliniche provenienti dal database pro-act tramite l'utilizzo di una rete bayesiana dinamica. Master's thesis, Università degli Studi di Padova, 2017.

[19] Roberto Gomeni, Maurizio Fava, and Pooled Resource Open-Access ALS Clinical Trials Consortium. Amyotrophic lateral sclerosis disease progression model. *Amyotrophic Lateral Sclerosis and Frontotemporal Degeneration*, 15(1-2):119–129, 2014.

[20] Frank E Harrell, Robert M Califf, David B Pryor, Kerry L Lee, and Robert A Rosati. Evaluating the yield of medical tests. *Jama*, 247(18):2543–2546, 1982.

[21] Patrick J Heagerty and Yingye Zheng. Survival model predictive accuracy and roc curves. *Biometrics*, 61(1):92–105, 2005.

[22] Torsten Hothorn and Berthold Lausen. On the exact distribution of maximally selected rank statistics. *Computational Statistics & Data Analysis*, 43(2):121–137, 2003.

[23] Hemant Ishwaran et al. Variable importance in binary regression trees and forests. *Electronic Journal of Statistics*, 1:519–537, 2007.

[24] Hemant Ishwaran and Udaya B Kogalur. *Random Forests for Survival, Regression, and Classification (RF-SRC)*, 2017. R package version 2.5.0.

[25] Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. *The annals of applied statistics*, pages 841–860, 2008.

[26] Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

[27] Matthew C Kiernan, Steve Vucic, Benjamin C Cheah, Martin R Turner, Andrew Eisen, Orla Hardiman, James R Burrell, and Margaret C Zoing. Amyotrophic lateral sclerosis. *The Lancet*, 377(9769):942–955, 2011.

[28] Robert Küffner, Neta Zach, Raquel Norel, Johann Hawe, David Schoenfeld, Liuxia Wang, Guang Li, Lilly Fang, Lester Mackey, Orla Hardiman, et al. Crowdsourced analysis of clinical trial data to predict amyotrophic lateral sclerosis progression. *Nature biotechnology*, 33(1):51–57, 2015.

[29] Michael LeBlanc and John Crowley. Survival trees by goodness of split. *Journal of the American Statistical Association*, 88(422):457–467, 1993.

[30] T Magnus, M Beck, R Giess, I Puls, M Naumann, and KV Toyka. Disease progression in amyotrophic lateral sclerosis: predictors of survival. *Muscle & nerve*, 25(5):709–714, 2002.

[31] Farrah J Mateen, Marco Carone, and Eric J Sorenson. Patients who survive 5 years or more with als in olmsted county, 1925–2004. *Journal of Neurology, Neurosurgery & Psychiatry*, pages jnnp–2009, 2010.

[32] Robert G Miller, JD Mitchell, Mary Lyon, and Dan H Moore. Riluzole for amyotrophic lateral sclerosis (als)/motor neuron disease (mnd). *Cochrane Database Syst Rev*, 1(1), 2007.

[33] D Naftel, E Blackstone, and M Turner. Conservation of events. *Unpublished notes*, 1985.

[34] Thais Mayumi Oshiro, Pedro Santoro Perez, and José Augusto Baranauskas. How many trees in a random forest? In *MLDM*, pages 154–168. Springer, 2012.

[35] Sabrina Paganoni, Jing Deng, Matthew Jaffa, Merit E Cudkowicz, and Anne-Marie Wills. Body mass index, not dyslipidemia, is an independent predictor of survival in amyotrophic lateral sclerosis. *Muscle & nerve*, 44(1):20–24, 2011.

[36] Sabrina Paganoni, Jing Deng, Matthew Jaffa, Merit E Cudkowicz, and Anne-Marie Wills. What does body mass index measure in amyotrophic lateral sclerosis and why should we care? *Muscle & nerve*, 45(4):612–612, 2012.

[37] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, Mar 1986.

[38] Ronit Reich-Slotky, Jinsy Andrews, Bin Cheng, Richard Buchsbaum, Diane Levy, Petra Kaufmann, and John LP Thompson. Body mass index (bmi) as predictor of alsfrs-r score decline in als patients. *Amyotrophic Lateral Sclerosis and Frontotemporal Degeneration*, 14(3):212–216, 2013.

[39] Jose C Roche, Ricardo Rojas-Garcia, Kirsten M Scott, William Scotton, Catherine E Ellis, Rachel Burman, Lokesh Wijesekera, Martin R Turner, P Nigel Leigh, Christopher E Shaw, et al. A proposed staging system for amyotrophic lateral sclerosis. *Brain*, 135(3):847–852, 2012.

[40] Lewis P Rowland and Neil A Shneider. Amyotrophic lateral sclerosis. *New England Journal of Medicine*, 344(22):1688–1700, 2001.

[41] Mark Robert Segal. Regression trees for censored data. *Biometrics*, pages 35–47, 1988.

[42] Carolin Strobl, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, 8(1):25, 2007.

[43] Irene Tramacere, Eleonora Dalla Bella, Adriano Chiò, Gabriele Mora, Graziella Filippini, and Giuseppe Lauria. The mitos system predicts long-term survival in amyotrophic lateral sclerosis. *Journal of Neurology, Neurosurgery & Psychiatry*, 86(11):1180–1185, 2015.

[44] Robert L Vender, David Mauger, Susan Walsh, Shoaib Alam, and Zachary Simmons. Respiratory systems abnormalities and clinical milestones for patients with amyotrophic lateral sclerosis with emphasis upon survival. *Amyotrophic Lateral Sclerosis*, 8(1):36–41, 2007.

[45] Joe H Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244, 1963.

[46] Neta Zach, David L Ennist, Albert A Taylor, Hagit Alon, Alexander Sherman, Robert Kueffner, Jason Walker, Ervin Sinani, Igor Katsovskiy, Merit Cudkowicz, et al. Being pro-active: What can a clinical trial database reveal about als? *Neurotherapeutics*, 12(2):417–423, 2015.