POLITECNICO DI TORINO
Corso di Laurea in Ingegneria Matematica

Tesi di Laurea Magistrale

Analisi di segnali fisiologici per la rilevazione della sonnolenza

Relatori
Professoressa Silvia Chiusano
Dottoressa Elena Daraio

Candidato
Edoardo Ottino

Dicembre 2018

Indice

Capitolo 1

Background

1.1 L'importanza del sonno

Dormire è un'attività di grande importanza per il corpo umano. Un insufficiente riposo notturno può avere serie ripercussioni sulla salute. Inoltre, individui assonnati causano quotidianamente numerosi incidenti stradali e, in generale, sono in grado di offrire performances ridotte nel loro lavoro e nella vita di tutti i giorni.
Secondo la classificazione utilizzata dalla National Sleep Foundation, il sonno si può suddividere in diverse fasi che si susseguono secondo un preciso ordine [1]:

- Fase N1 (Non-REM 1): fase di dormiveglia, in cui il soggetto può essere svegliato facilmente. Si caratterizza per un rallentamento dei movimenti oculari e dell'attività muscolare.
- Fase N2 (Non-REM 2): il corpo si prepara al sonno vero e proprio. La frequenza cardiaca e l'attività cerebrale rallentano, la temperatura corporea si abbassa e i movimenti oculari terminano.
- Fase N3 (Non-REM 3): fase di sonno profondo, in questa fase possono verificarsi incubi ed episodi di sonnambulismo. Le onde cerebrali rallentano ulteriormente e prendono il nome di *onde delta*. Si registra una diminuzione ulteriore della temperatura corporea ed un rallentamento di frequenza respiratoria e pressione sanguigna
- Fase REM (Rapid Eye Movement): in questa fase avvengono i sogni più vividi. Gli occhi si muovono molto rapidamente, aumentano la frequenza respiratoria e la frequenza cardiaca.

Nella Figura 1.1 è rappresentata l'evoluzione delle onde cerebrali durante le varie fasi del sonno appena descritte. Un ciclo completo di sonno dura dai 90 ai 120 minuti. Gli ultimi cicli durante la notte sono caratterizzati da una fase REM più lunga a scapito delle fasi di sonno profondo.
La fase di sonno profondo è molto importante perchè in essa viene rilasciato l'ormone della crescita, che ha effetti benefici sull'apparato muscolare. Inoltre, anche il sistema immunitario si riposa e il cervello si predispone all'apprendimento per il giorno successivo.
Per quanto riguarda il numero di ore giornaliere che sono necessarie per un adeguato riposo, questo varia in funzione dell'età: i bambini e gli adolescenti dovrebbero dormire di più, mentre per gli adulti sono raccomandate dalle 7 alle 9 ore di sonno. Anche dormire troppo è sconsigliato in quanto comporta rischi analoghi a dormire troppo poco [1].
Portare avanti sistematicamente abitudini di sonno errate può portare a conseguenze negative a lungo termine per la salute [3]. In particolare, sono state trovate correlazioni tra abitudini di scarso sonno e varie patologie quali:

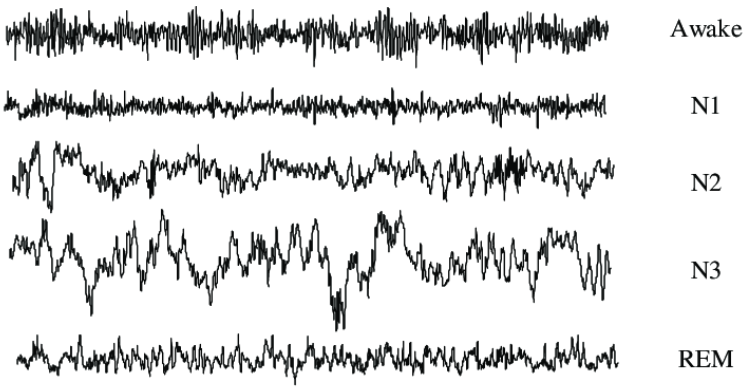


Figura 1.1: Evoluzione delle onde cerebrali in funzione della fase del sonno [2]

- Aumento dell'indice di massa corporea (BMI) e, nei casi più gravi, obesità
- Diabete di tipo 2
- Patologie cardiovascolari come ipertensione o infarti
- Debolezza del sistema immunitario

La relazione tra l'aumento dell'indice di massa corporea e la carenza di sonno è legata alla secrezione di ormoni che viene alterata da cattive abitudini relative al sonno. Questi ormoni sono la *grelina*, che stimola la fame, e la *leptina*, che la inibisce [4]. In una ricerca condotta in Australia su una popolazione di over 55 [5] si è quantificato il rischio relativo di sviluppare un elevato BMI in seguito ad abitudini di sonno di durata inadeguata: per chi dorme meno di 6 ore a notte il rischio relativo ammonta a 1.52 rispetto a chi dorme regolarmente 7 ore. Sempre rispetto a dormire 7 ore a notte, dormire 6 ore a notte è associato ad un rischio relativo di 1.42, mentre dormire più di 9 ore a notte è associato ad un rischio relativo di 1.19.
In uno studio del 2003 [6], invece, si è analizzato il rischio di sviluppare malattie cardiovascolari e infarti seguendo nel tempo una popolazione di 121700 donne tra i 30 e i 55 anni. Emergono rischi relativi alti e in larga parte statisticamente significativi, differenti per ogni fascia di ore di sonno quotidiane. Particolarmente rilevante è il risultato che riguarda i soggetti che dormono 5 o meno ore a notte, per i quali il tasso di rischio relativo ammonta a 1.82, ovvero hanno un rischio di sviluppare le patologie sopracitate che è quasi doppio rispetto a chi dorme regolarmente 8 ore. La tabella 1.1 riporta una sintesi dei risultati dell'intero studio.

Sonno quotidiano	**Rischio relativo**	**Intervallo di confidenza**
$<$ 5 h	1.82	(1.34,2.41)
6 h	1.30	(1.08,1.57)
7 h	1.06	(0.89,1.26)
8 h	1.00	(1.00,1.00)
$>$ 9 h	1.57	(1.18,2.11)

Tabella 1.1: Rischio di sviluppare malattie cardiovascolari in funzione delle ore di sonno

Di conseguenza, dormire abitualmente un numero scarso o elevato di ore influenza al ribasso l'aspettativa di vita. Ciò è stato confermato da uno studio del 2007 [7], in cui si è registrato un

consistente aumento della mortalità in entrambi i casi, più rilevante per quanto riguarda gli uomini. Nel caso di soggetti che dormono meno di 7 ore a notte, la mortalità aumenta del 26% per gli uomini e del 21% per le donne. Nel caso di soggetti che dormono più di 8 ore a notte, la mortalità aumenta del 24% per gli uomini e del 17% per le donne.
In conclusione, mantenere abitudini di sonno errate porta a lungo termine a pesanti conseguenze sulla salute ed inficia notevolmente la qualità e la durata della vita.

1.2 Impatto sociale di un riposo inadeguato

Oltre ai rischi per la salute che ricadono sul singolo individuo, uno scarso riposo può avere effetti negativi anche sulla collettività. Si pensi ad esempio ad un calo della produttività lavorativa, a possibili errori da parte di soggetti con mansioni di alta responsabilità, oppure a danni derivanti da incidenti stradali.
Il problema della diminuzione di efficienza lavorativa a causa di abitudini di sonno sbagliate è stato affrontato in uno studio effettuato in Australia nel periodo 2016-2017 [14], che ne quantifica il costo per l'economia dello Stato. Sono stati presi in esame i costi imputabili a lavoratori assenti dal posto di lavoro o inefficienti, ad una ridotta occupazione e a decessi prematuri. I risultati sono riepilogati nella tabella 1.2, da cui si evince che il costo totale per l'economia australiana nel periodo preso in esame è stato di 12,19 miliardi di dollari.

Tipologia di danno	**Costo in milioni di dollari**
Assenze dal lavoro	1729,3
Ridotta produttività	4632,2
Ridotta occupazione	5223,1
Morti premature	609,7
Totale	12194,4

Tabella 1.2: Impatto di errate abitudini di sonno sull'economia australiana nel biennio 2016-2017

Tralasciando l'aspetto economico, ci sono categorie di lavoratori la cui inefficienza può avere un costo notevole in termini di vite umane. Un caso emblematico è quello degli operatori sanitari, come i medici. Uno studio del 2004 [8] confronta le performances di due gruppi di operatori sanitari. I due gruppi si differenziano per il fatto che uno dei due è sottoposto a turni di lavoro più intensi che limitano il riposo notturno. L'efficienza di quest'ultimo gruppo è risultata di gran lunga peggiore: le persone appartenenti ad esso commettono in media il 36% in più di errori di tipo operativo. Lo stesso gruppo ha anche un rischio relativo di commettere errori diagnostici classificati come gravi che ammonta a 5.6.
Un impatto pesante in termini di vite umane e di costi per la collettività è da attribuire anche agli incidenti stradali causati da privazione di sonno. La sonnolenza del guidatore è un fattore che incrementa considerevolmente il rischio di provocare un incidente stradale. La quantificazione precisa dell'impatto diretto di questo aspetto non è un obiettivo banale, tuttavia diversi studi suggeriscono che la percentuale di incidenti stradali in cui sono coinvolti guidatori stanchi o affaticati sia non trascurabile. In particolare, si stima che in un range che va dal 7% al 30% degli incidenti stradali totali siano coinvolti veicoli il cui conducente non è adeguatamente riposato [9, 10, 11, 12]. In Italia, il fenomeno è associato ad un quinto degli incidenti totali [13]. Un elemento di cui è interessante tenere conto è il fatto che la tipologia di incidenti in oggetto causa generalmente danni più gravi a cose e persone: ad esempio, secondo una ricerca condotta negli Stati Uniti, l'incidenza della sonnolenza sul numero totale degli incidenti si attesta al 7%, ma tale percentuale sale al 13.1% per gli incidenti che causano ricoveri ospedalieri e addirittura al 16.5% nel caso di incidenti

mortali [9].
Si può quindi comprendere l'importanza della questione, che ha fatto sì che negli ultimi anni siano stati svolti svariati studi dedicati all'approfondimento del fenomeno, nel tentativo di comprendere meglio quali fattori di rischio condizionino la probabilità di addormentarsi alla guida e conseguentemente ipotizzare l'attuazione di strategie di prevenzione.
I sopracitati studi si basano prevalentemente su indagini statistiche che hanno prodotto una considerevole mole di dati. Questi generalmente sono stati raccolti tramite due diversi approcci: tramite questionari sottoposti a campioni della popolazione oppure tramite esperimenti che prevedono l'impiego di simulatori di guida o il monitoraggio delle abitudini di guida reale dei soggetti selezionati.

Dati racccolti tramite questionari Un approccio largamente utilizzato nella raccolta di dati relativi alla sonnolenza alla guida è quello della somministrazione di questionari. Esso ha il vantaggio di poter indagare in modo rapido le abitudini legate al sonno di una vasta porzione della popolazione così da poter ipotizzare eventuali correlazioni con l'addormentamento durante la guida.
Gli elementi che compaiono più frequentemente associati ad episodi di sonnolenza e che possono quindi essere considerati fattori di rischio, si suddividono in due categorie:

- elementi relativi alle caratteristiche del singolo soggetto preso in esame come il sesso, l'età anagrafica, regolarità e qualità del sonno e numero di chilometri percorsi annualmente.
- elementi appartenenti al contesto esterno come lunghezza del tragitto, fascia oraria del viaggio e tipologia di strada percorsa.

Per quanto riguarda gli elementi che caratterizzano il soggetto in esame, emerge che gli individui di sesso maschile hanno un rischio di addormentamento superiore di 1.79 volte, mentre in generale tale rischio tende a diminuire con l'avanzare dell'età anagrafica. Rispetto agli over 70, ad esempio, gli individui tra i 17 e i 30 anni hanno un rischio superiore di 1.56 volte, quelli tra i 51 e i 70 di 1.67. [11].
Come prevedibile, la quantità di sonno della notte precedente influisce pesantemente: nel 66% dei casi, chi è colpito da un colpo di sonno ha dormito meno di 6 ore la notte prima dell'episodio [15].
Inoltre, non va sottovalutato l'impatto di disturbi del sonno, i quali possono essere sconosciuti a chi ne soffre, pur essendo relativamente facili da diagnosticare. Uno dei più comuni è l'apnea ostruttiva del sonno (OSAS) che aumenta il rischio di addormentamento di 3.48 volte [11].
Infine, percorrere molti chilometri durante l'anno rende più probabile il verificarsi di un colpo di sonno. Chi percorre più di 20000 chilometri è 2.02 volte più soggetto a colpi di sonno [11].
Per quanto riguarda, invece, gli elementi esterni al soggetto in esame si è rilevato che il fenomeno dell'addormentamento avviene nel 47% dei casi durante la prima ora di viaggio e nel 58% dei casi in autostrada [15]. Queste osservazioni, unitamente alle considerazioni fatte precedentemente riguardo al numero di chilometri percorsi annualmente, portano ad individuare alcune categorie a rischio, come quelle degli autisti professionisti ed in particolare dei camionisti. A conferma di ciò, si è osservato che i camionisti tendono 3.02 volte tanto ad ottenere punteggi bassi nel test PSQI *(Pittsburgh Scale Quality Index)*, che misura la qualità del sonno del soggetto sulla base delle risposte date ad un questionario sulle proprie abitudini. Inoltre, è associato un rischio ancora maggiore, 4.99, a lavoratori che svolgono turni notturni [16].
Infine, un ulteriore aspetto generale che cattura l'attenzione è inerente alla percezione del rischio da parte della popolazione, che non ne è sempre pienamente cosciente: infatti il 57% degli intervistati dichiara di non fermarsi una volta avvertiti sintomi di sonnolenza. Questo avviene nonostante ben il 37% affermi di essersi addormentato almeno una volta alla guida [15] e, in un altro studio, il 17% ammetta che tale perdita di coscienza alla guida è avvenuta almeno una volta negli ultimi 2 anni [11].

Dati raccolti durante sessioni di guida Mentre la somministrazione di questionari è per sua natura fortemente influenzata dalla percezione del soggetto che fornisce le risposte, esistono altri approcci che consentono di acquisire dati in maniera più diretta.
In particolare, può essere interessante analizzare il comportamento alla guida di un campione selezionato di soggetti. Ciò può avvenire in due modi:

- in modo *diretto*, analizzando le prestazioni dei soggetti nell'atto di pilotare il proprio veicolo su strada
- in modo *indiretto*, analizzando le prestazioni dei soggetti durante sessioni di guida simulata

Nel primo caso è necessario fare uso di sistemi di videosorveglianza o altri strumenti di rilevazione di immagini. Lo scopo è, qualora si verifichino incidenti, misurare il livello di sonnolenza del guidatore nei minuti immediatamente precedenti. Ciò si può fare tramite indicatori quali:

- la frazione di tempo in cui la percentuale di chiusura delle palpebre supera una certa soglia. Questo indice viene chiamato *PERCLOS* [12]
- l'intervallo che intercorre tra i micromovimenti dell'occhio, che sono detti *saccadi* [17]

Questo modo di procedere, tuttavia, presenta alcune criticità:

- il numero di volontari e la finestra di monitoraggio devono essere sufficientemente ampi per poter ottenere una mole adeguata di dati
- la qualità delle immagini potrebbe essere insoddisfacente, così da scartare alcune osservazioni
- la quantità di parametri monitorabili agendo in questo modo risulta limitata

Particolarmente rilevante è il primo punto, se si pensa che in uno studio americano che utilizza questa metodologia [12] 3593 guidatori sono stati seguiti per 38 mesi. Questo comporta difficoltà organizzative non facilmente gestibili.
Possono, quindi, essere utilizzati con profitto simulatori di guida che riproducano le reali condizioni di guida su strada. Quest'ultimo approccio ha un ulteriore aspetto vantaggioso, ovvero la possibilità di approfondire un determinato parametro di interesse. Esso può appartenere sia alla sfera soggettiva del paziente che ai fattori esterni. Tale analisi avviene tramite la suddivisione dei partecipanti all'esperimento in diversi gruppi oppure tramite sessioni diverse di guida sotto mutate condizioni [18, 19, 20].
Solitamente, i volontari sottoposti all'esperimento compiono la sessione di guida in solitudine, senza altri passeggeri, e dopo essere stati precedentemente addestrati all'utilizzo del simulatore tramite una sessione di prova. In generale, la configurazione dell'esperimento, sia per quanto riguarda il campione dei soggetti partecipanti sia per quanto riguarda la riproduzione dell'ambientazione esterna, è volta a rappresentare condizioni più favorevoli al verificarsi di episodi di sonnolenza, sulla base delle rilevazioni statistiche esposte nel paragrafo precedente. Di norma, i soggetti reclutati per l'esperimento sono in buona parte di sesso maschile, giovani o al massimo di mezza età e svolgono la professione di autista o lavorano, almeno saltuariamente, durante la notte.
Contestualmente, è d'uso svolgere gli esperimenti durante le prime ore del mattino, in un lasso di tempo non lungo e riproducendo un'ambientazione che raffiguri strade larghe e con scarso traffico.
Per valutare le performances di guida del soggetto coinvolto nell'esperimento, è consuetudine prendere in considerazione parametri oggettivi relativi alla conduzione del veicolo quali:

- indicatori che caratterizzino lo stile di guida del soggetto. Ad esempio la frequenza delle sterzate in un certo intervallo di tempo [18] o la loro ampiezza misurata in gradi [18]

- indicatori che misurino l'incidentalità. Ad esempio la frequenza di episodi in cui le ruote fuoriescono dalla carreggiata [19]

Da esperimenti di simulazioni di guida effettuati da diversi gruppi di ricercatori emergono alcuni dati interessanti. In Thiffault, Bergeron [18] si riscontra un incremento statisticamente significativo dell'ampiezza media delle sterzate compiute nel corso dell'esperimento, che nei primi 20 minuti passa da 4,6 a 5,2 gradi. Ciò suggerisce un aumento delle sterzate di maggior ampiezza imputabile ad una crescente stanchezza che si manifesta già nei primi minuti di guida. Questo risultato è consistente con quanto esposto precedentemente e cioè con il fatto che gran parte dei colpi di sonno avvengono nella fase iniziale del viaggio.
Un altro risultato significativo lo si trova in Akerstedt et al. [19] , in cui un gruppo di volontari viene sottoposto a due analoghe sessioni di guida al simulatore, la prima dopo una notte regolare di sonno e la seconda dopo una notte completamente insonne poichè occupata da un turno di lavoro. Come previsto dalle ipotesi iniziali, emerge un netto peggioramento delle performances dei soggetti nel secondo caso. Tale peggioramento si manifesta sia attraverso un importante aumento del tempo in cui le palpebre si mantengono chiuse, sia attraverso un incremento del numero di episodi in cui il soggetto non riesce a mantenere tutte le ruote del veicolo all'interno della carreggiata.

Quanto precedentemente esposto ha l'obiettivo di sviluppare la consapevolezza delle problematiche relative ad uno scarso riposo ed in particolar modo dei rischi derivanti dall'addormentamento alla guida. Il passo seguente consiste in un'analisi più dettagliata dei parametri fisiologici che sono stati utilizzati in letteratura per caratterizzare il passaggio dalla fase di veglia alla fase di sonno.

1.3 Monitoraggio di parametri fisiologici

La raccolta di dati statistici è utile per inquadrare il fenomeno dell'addormentamento alla guida e per acquisire consapevolezza delle sue conseguenze sociali. In questa sezione, invece, si definisce la *polisonnografia*, allo scopo di introdurre gli strumenti utili ad un'analisi in tempo reale dei parametri fisiologici che caratterizzano un soggetto alla guida.
La polisonnografia consiste nella registrazione contemporanea di diversi parametri fisiologici di un soggetto dormiente e nella loro analisi. Di norma, il suo obiettivo è quello di diagnosticare eventuali disturbi del sonno. I segnali fisiologici che possono essere presi in considerazione sono molteplici. Un elenco dei più importanti contiene l'elettroencefalogramma (EEG), l'elettrocardiogramma (ECG), la fotopletismografia (PPG), la frequenza respiratoria e la pressione sanguigna [21].
Un ulteriore utilizzo che si può fare della polisonnografia consiste nell'evidenziare le differenze fisiologiche tra un soggetto dormiente ed uno in stato di veglia. Questo aspetto può essere molto interessante nello studio del fenomeno dell'addormentamento alla guida. Infatti, la possibilità di classificare un guidatore come *a rischio* sulla base di parametri oggettivi, sarebbe di grande aiuto nella prevenzione del fenomeno in esame.
In letteratura sono presenti alcuni studi che approfondiscono il tema mediante misurazioni effettuate durante simulazioni di guida oppure durante vere e proprie sessioni di guida effettuate su strada in condizioni di sicurezza garantite dalla supervisione di personale medico specializzato.
Ad esempio, si può analizzare il tracciato di un elettrocardiogramma, come quello rappresentato in Figura 1.2, in cui sono presenti diversi elementi di interesse. In particolare, viene studiata la sequenza RR, ovvero la distanza tra due picchi R. L'analisi di questa sequenza prende il nome di HRV *(Heart Rate Variability)* e le frequenze presenti tra i due picchi R vengono suddivise in [22]:

- LF *(Low Frequencies)*: in corrispondenza di queste si localizza il picco dell'attività del sistema nervoso simpatico, che abbassa la frequenza cardiaca.

- HF *(High Frequencies)*: corrispondono al picco dell'attività del sistema parasimpatico, che aumenta la frequenza cardiaca.

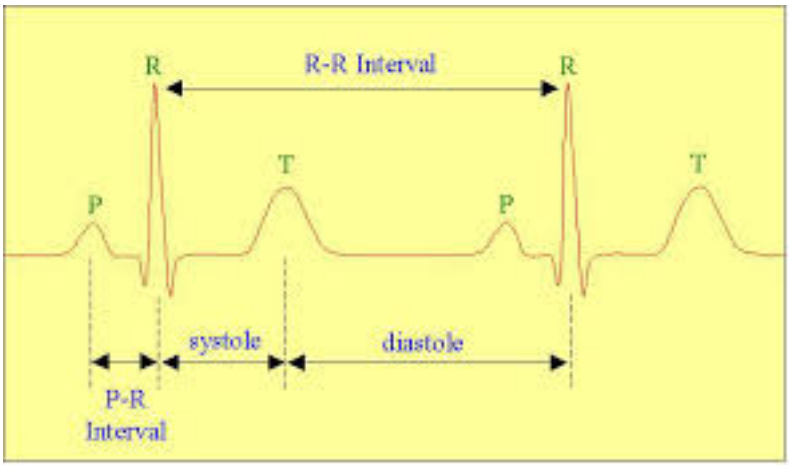


Figura 1.2: Esempio di tracciato di un ECG con sequenza RR [22]

Le componenti LF, HF ed il rapporto tra esse sono elementi che sono stati proposti come indicatori dello stato di veglia di un soggetto. Infatti si è osservato che a bassi valori di LF, che corrispondono ad una minore stimolazione da parte del sistema nervoso simpatico, si registra un incremento degli errori di guida. Ciò è conseguenza di un calo dell'attenzione e di una maggiore difficoltà del soggetto di mantenersi vigile [23].
Tali osservazioni possono essere utilizzate come punto di partenza per la progettazione di tecnologie innovative. Una possibile direzione di sviluppo è un sistema che allerti l'autista del rischio di addormentamento sulla base della variazione dell'andamento dei parametri in esame. In alternativa, il sistema può analizzare i dati relativi ai primi minuti di guida e catalogare il guidatore come idoneo alla guida o meno [24].
Nell'implementazione pratica di queste idee, occorre tuttavia tenere conto di due fondamentali trade-off:

- la precisione deve essere massimizzata senza rendere il sistema troppo invasivo.
- i falsi positivi devono essere minimizzati, senza perdere di efficienza nella segnalazione di situazioni di pericolo.

Relativamente ai falsi allarmi, infatti, un elevato numero di questi potrebbe indurre il soggetto ad ignorare gli avvertimenti del sistema, rendendolo di fatto inutile. Ad esempio, nello studio precedentemente citato [24], il sistema messo a punto ha come risultato un esiguo numero di falsi positivi (4%), ma a fronte di una percentuale di riconoscimento di episodi di sonnolenza non pienamente soddisfacente (59%).
In conclusione, in questo capitolo si è voluto evidenziare come il fenomeno dell'addormentamento alla guida può essere analizzato da diversi punti di vista. Uno dei più promettenti consiste nell'analisi di segnali fisiologici dei soggetti in esame. Per questo motivo nel prossimo capitolo si tratteranno alcuni importanti aspetti teorici inerenti allo studio di segnali.

Capitolo 2

Cenni di analisi dei segnali

2.1 Concetti di base

In generale, si definisce come *segnale* una funzione del tempo $x(t)$ che rappresenta l'evoluzione di un determinato fenomeno fisico osservabile.
I segnali possono essere classificati secondo differenti criteri, un segnale può infatti essere [39]:

- continuo o discreto, se è definito per tutti i valori di t in un certo intervallo o solo per un sottoinsieme di valori discreto
- reale o complesso, se assume soltanto valori reali o può assumere valori complessi
- scalare o vettoriale, se è descritto da uno o più valori ad ogni istante
- analogico o digitale, se può assumere qualsiasi valore in un certo intervallo o solo valori contenuti in un determinato insieme discreto
- periodico, se esiste un periodo τ tale che, per ogni t per cui il segnale è definito:

$$x(t) = x(t + \tau) \tag{2.1}$$

- deterministico o stocastico, se può essere conosciuto a priori o meno

Nell'ambito di questo lavoro verranno analizzati segnali continui, analogici e periodici.
La maggior parte dei segnali ottenuti dall'osservazione di fenomeni fisici reali sono di tipo continuo. Analizzare un segnale discreto può essere tuttavia più semplice. Per questo motivo può essere utile lavorare con un'approssimazione discreta di un segnale continuo. Per ottenere un segnale discreto da un segnale continuo si utilizza un procedimento detto *campionamento*, ovvero si estraggono i valori del segnale continuo soltanto in determinati istanti $t_1, t_2, ..., t_N$. I valori del segnale in corrispondenza di questi istanti sono denominati *campioni*.
Se la differenza:

$$\Delta(t) = t_i - t_{i-1} \tag{2.2}$$

è costante per tutti i valori di i, allora si parla di campionamento *uniforme*. Il numero di campioni che vengono presi nell'unità di tempo di riferimento è detto *frequenza di campionamento*. Al suo crescere l'approssimazione del segnale continuo originale con quello discretizzato sarà migliore.

Sistemi Si definisce *sistema* un operatore T che elabora uno o più segnali in input e restituisce uno o più segnali come output [40]. I sistemi vengono classificati nel modo seguente [39]:

- continui o discreti, a seconda della tipologia di segnale eleborato
- statici o dinamici, se l'output dipende solo dai valori correnti in input o se i valori agli istanti precedenti hanno un effetto sull'output
- causali, se l'uscita $y(t)$ dipende dall'ingresso $x(t)$ solo per valori temporali τ non superiori a t
- lineari, se ad una combinazione lineare di differenti ingressi corrisponde in uscita la stessa combinazione lineare delle relative uscite
- tempo-invarianti, se il loro comportamento si mantiene uguale nel tempo

Tra le caratteristiche elencate, una delle più importanti è la *causalità*. Essa infatti discrimina i filtri che possono essere realizzati praticamente da quelli che possono essere soltanto presi come riferimento teorico. Tale suddivisione sarà approfondita in seguito. Formalmente, per garantire la causalità di un sistema è necessario che valga la seguente condizione necessaria:

$$h(t) = 0, \forall t < 0 \tag{2.3}$$

Ovvero, la risposta impulsiva del sistema deve essere nulla per tutti i valori negativi di t [46].
Inoltre, sono di particolare interesse i sistemi contemporaneamente lineari e tempo-invarianti (LTI). Una importante proprietà di questi sistemi è quella di poter calcolare la loro risposta a qualsiasi segnale in ingresso, una volta che è nota la *risposta all'impulso del sistema*. Quest'ultima si definisce come:

$$h(t) = T(\delta(t)) \tag{2.4}$$

dove $\delta(t)$ è il *delta di Dirac* e T l'operatore che rappresenta il sistema [41].
Nel caso in cui il segnale in esame sia analogico vale la seguente equazione:

$$T(x(t)) = \int_{-\infty}^{\infty} h(y)x(t-y)dy \tag{2.5}$$

che formalizza come la risposta di un sistema LTI ad un certo segnale analogico in input si ottiene con la convoluzione del segnale stesso con la risposta del sistema all'impulso.

La trasformata di Fourier L'operazione che corrisponde all'equazione 2.5 può essere formalmente ridotta grazie all'utilizzo della *trasformata di Fourier* [39]. Grazie ad essa è possibile spostare l'analisi del segnale in oggetto dal dominio del tempo al dominio delle frequenze.
In generale, la trasformata di Fourier di una funzione si definisce come:

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt \tag{2.6}$$

Dove ω viene detta *pulsazione* ed è definita nel modo seguente:

$$\omega = 2\pi\nu \tag{2.7}$$

La pulsazione dipende da ν, che rappresenta la *frequenza del segnale*, ovvero, per un segnale periodico, il numero di ripetizioni del periodo T nell'unità di tempo di riferimento [41]:

$$\nu = \frac{1}{T} \tag{2.8}$$

L'unità di misura della frequenza è l'*Hertz* (Hz). Un Hertz corrisponde a sec^{-1}.
Una proprietà interessante della trasformata di Fourier è quella di semplificare un'operazione come la convoluzione che è computazionalmente complessa. Infatti la convoluzione tra due segnali è equivalente al prodotto tra le rispettive trasformate di Fourier. Formalmente si scrive che:

$$\int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau \leftrightarrow X_1(\omega)X_2(\omega) \tag{2.9}$$

In particolare, la trasformata di Fourier della risposta impulsiva del sistema viene detta *funzione di trasferimento* [46]. L'utilizzo della funzione di trasferimento e della trasformata di Fourier sono utili nel calcolare la risposta di un sistema LTI ad un dato segnale in input.

Rappresentazione spettrale dei segnali Avendo introdotto il concetto di frequenza, si può passare ad un nuovo tipo di rappresentazione del segnale, utilizzando il suo *spettro*. Per comprendere questo tipo di rappresentazione è necessario introdurre i segnali sinusoidali.
In generale, si definisce *sinusoidale* un caso particolare di segnale periodico che obbedisce ad una legge del seguente tipo:

$$x(t) = A\sin(2\pi\frac{t}{T} + \theta) \tag{2.10}$$

Nella definizione precedente entrano in gioco i seguenti parametri:

- il *periodo* T, ovvero l'intervallo di tempo che intercorre tra 2 valori uguali del segnale
- l'*ampiezza* A, ovvero il valore massimo (in valore assoluto) che può assumere il segnale
- la *fase iniziale* θ, ovvero la fase in $t = 0$

Utilizzando la *relazione di Eulero* (2.11), si possono esprimere segnali sinusoidali mediante esponenziali immaginari. Ciò può essere utile perchè è più semplice trattare algebricamente funzioni esponenziali piuttosto che funzioni trigonometriche [41].

$$e^{i\theta} = cos(\theta) + i\sin(\theta) \tag{2.11}$$

Nell'ambito della rappresentazione di un generico segnale, fissato un insieme di frequenze $\nu_1, \nu_2, ..., \nu_m$, a queste si fa corrispondere un insieme di segnali sinusoidali $e^{2i\pi\nu_1 t}, e^{2i\pi\nu_2 t}, ..., e^{2i\pi\nu_m t}$. Un qualsiasi segnale $x(t)$ può essere ottenuto combinando linearmente i precedenti segnali sinusoidali, utilizzando un insieme di coefficienti $F(\nu_1)F(\nu_2), ..., F(\nu_m)$. Il risultato è il seguente:

$$x(t) = \sum_{n=1}^{m} F(\nu_n)e^{2i\pi\nu_m t} \tag{2.12}$$

Lo *spettro* si definisce come l'insieme dei coefficienti $F(\nu_1), F(\nu_2), ..., F(\nu_m)$ [41]. La funzione 2.12 è una funzione a valori complessi, per la sua rappresentazione grafica sono quindi necessari sia il grafico del modulo sia il grafico della fase.
L'utilizzo della rappresentazione spettrale dei segnali e lo studio del comportamento dei sistemi LTI hanno un'importante applicazione pratica. Infatti, i sistemi LTI vengono anche definiti *filtri* perchè si possono utilizzare per effettuare una selezione delle frequenze del segnale in esame.

2.2 Filtri

L'uso dei filtri può essere utile per isolare le frequenze di un segnale, ad esempio per isolare le frequenze corrispondenti ad elementi di disturbo. In generale, l'intervallo delle frequenze isolate

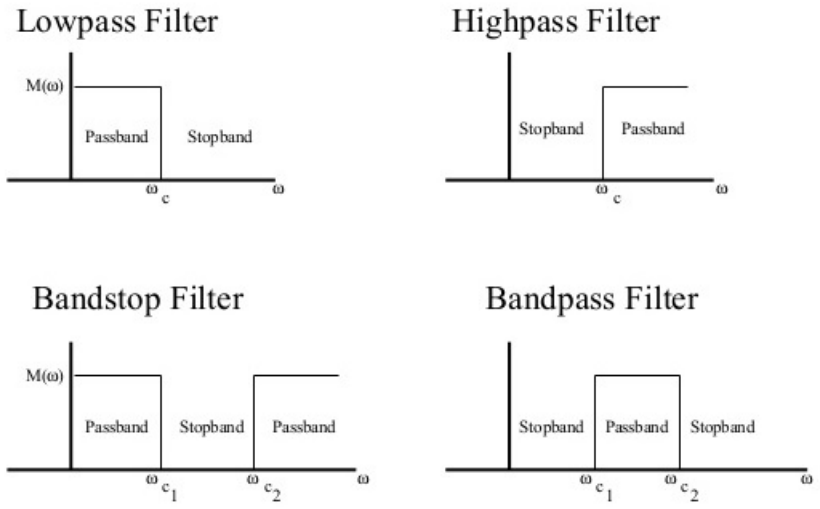


Figura 2.1: Rappresentazione banda passante e banda proibita dei filtri ideali [42]

da un filtro viene definito *banda passante*, mentre l'intervallo che viene tagliato *banda proibita*.
I cosiddetti filtri *ideali* consentono di distinguere completamente le frequenze di disturbo dal segnale pulito. Una rappresentazione del loro effetto si può trovare nella Figura 2.1. I filtri ideali si classificano in [45]:

- filtro *passa-basso*: definita una *frequenza di taglio* ν_c, isola le frequenze inferiori a ν_c e taglia le rimanenti
- filtro *passa-alto*: definita una *frequenza di taglio* ν_c, isola le frequenze superiori a ν_c e taglia le rimanenti
- filtro *passa-banda*: definite due frequenze di taglio ν_a e ν_b, isola le frequenze comprese nell'intervallo $[\nu_a, \nu_b]$ e taglia le frequenze al di fuori dell'intervallo
- filtro *elimina-banda*: definite due frequenze di taglio ν_a e ν_b, isola le frequenze al di fuori dell'intervallo $[\nu_a, \nu_b]$ e taglia le frequenze comprese nell'intervallo

I filtri appena elencati, tuttavia, sono detti ideali perchè impossibili da realizzare nella pratica. La ragione è che tutti questi filtri non sono *causali*, di conseguenza non possono essere realizzati da un sistema fisico che rispetti il principio di causalità. Nella sezione precedente si era già enunciata la condizione necessaria che deve essere verificata affinchè un sistema sia causale (2.3). Ora si mostra come tale condizione non sia valida nel caso di un filtro passa-basso.
La risposta impulsiva di un filtro passa-basso è:

$$h(t) = 2\nu_c sinc(2\nu_c t) \tag{2.13}$$

Per verificare la condizione 2.3, l'equazione 2.13 dovrebbe essere nulla per tutti i valori negativi di t. Come vedremo, ciò non è vero. Infatti, il grafico della funzione *sinc* in un intorno di 0 è rappresentato nella figura 2.2. Per ottenere la condizione di causalità nel caso in esame, la funzione $sinc(2\nu_c t)$ dovrebbe essere nulla per tutti i valori negativi di t. Dal grafico in Figura 2.2 si vede immediatamente che ciò non è vero. Questa affermazione vale a prescindere dall'ipotetico segno della frequenza di taglio ν_c, infatti il seno cardinale presenta oscillazioni attorno all'origine sia per valori negativi che per valori positivi dell'argomento.

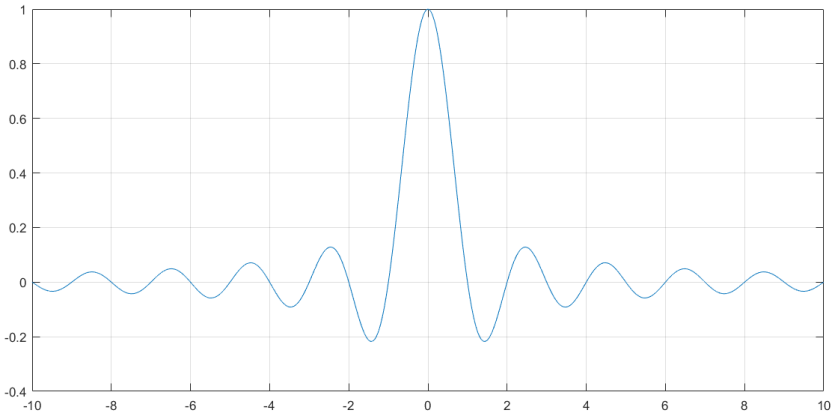


Figura 2.2: Rappresentazione grafica della funzione *sinc*

Ciò può essere verificato nella Figura 2.3, che fornisce una rappresentazione grafica dei valori di $h(t)$ in corrispondenza dei valori negativi di t. Per questo motivo la condizione 2.3 non è soddisfatta

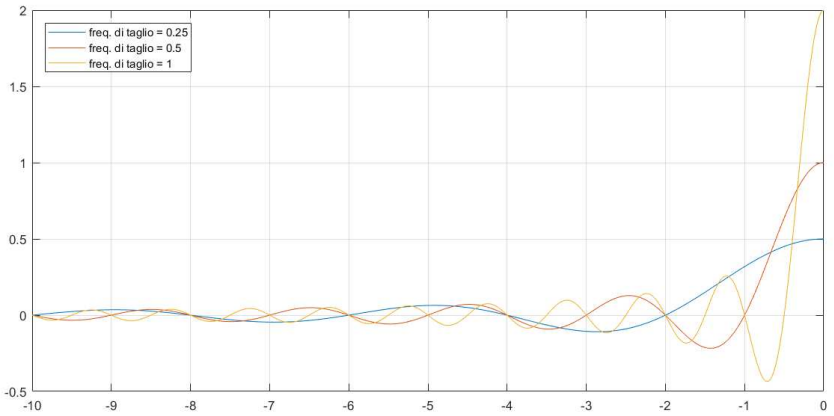


Figura 2.3: Rappresentazione grafica della risposta impulsiva di un filtro passa-basso per valori negativi di t, al variare della frequenza di taglio ν_c

e di conseguenza il filtro passa-basso non è un sistema causale. Lo stesso procedimento può essere replicato in modo analogo per gli altri filtri ideali, utilizzando le rispettive risposte impulsive.
I filtri ideali possono essere quindi solo approssimati da filtri *reali*.

Approssimazione di filtri ideali I filtri reali sono progettati per approssimare i filtri ideali con l'obiettivo di mantenere il più possibile alcune caratteristiche utili di quest'ultimi.
É quindi importante definire dei criteri che determinino la bontà dell'approssimazione ottenuta. In particolare, di norma si prendono in esame il modulo e la fase della funzione di trasferimento $H(\nu)$, ovvero della trasformata di Fourier della risposta impulsiva del sistema. La funzione di trasferimento di un filtro realizzabile, infatti, ha notevoli differenze rispetto alla funzione di trasferimento

di un filtro ideale.
A titolo di esempio si analizza la funzione di trasferimento di un filtro passa-basso realizzabile, le considerazioni che si faranno possono essere replicate analogamente per gli altri tipi di filtri ideali. Nella Figura 2.4 è rappresentato il modulo della tipica funzione di trasferimento di un filtro passa-basso reale. Rispetto al modulo della funzione di trasferimento di un filtro ideale si notano due

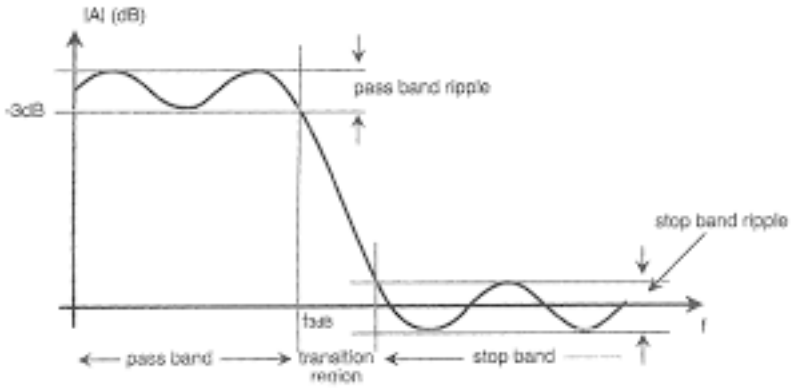


Figura 2.4: Rappresentazione grafica del modulo della tipica funzione di trasferimento $H(\nu)$ di un filtro passa-basso reale [43]

importanti differenze:

- la banda passante e la banda proibita sono separate dalla cosiddetta *banda di transizione*. In un filtro ideale non esiste la banda di transizione e banda passante e banda proibita sono separate dalla frequenza di taglio.
- la banda passante e la banda proibita presentano delle oscillazioni. In un filtro ideale la banda passante è costante ed ha valore 1 mentre la banda proibita è nulla.

Un'approssimazione è migliore tanto più si avvicina all'ideale, per cui un filtro reale è migliore tanto più:

- l'ampiezza della banda di transizione è ridotta
- l'ampiezza delle oscillazioni nella banda passante e nella banda proibita è ridotta

Per valutare un filtro reale sulla base dei parametri appena enunciati si parla di bontà nell'approssimazione del *guadagno*. Si utilizza questo termine per esprimere quanto l'approssimazione ottenuta si avvicina al filtro ideale.
Un altro punto di vista è quello che prende in esame la fase di $H(\nu)$. Nel caso di un filtro ideale, essa è lineare. Tale proprietà non vale per i filtri reali.
Una fase non lineare può creare alcuni problemi durante il filtraggio del segnale. In particolare, può creare ritardi differenti per le componenti del segnale di diversa frequenza, causando una distorsione del segnale stesso [41]. Per questo motivo è opportuno, nella progettazione di un filtro, cercare di riprodurre una fase che abbia un andamento il più possibile lineare.
Esistono diverse famiglie di filtri reali. Le principali sono elencate nella Tabella 2.1. In essa è anche riportata una valutazione qualitativa della bontà dell'approssimazione che si ottiene con ciascuna famiglia [41]. Come si può notare guardando la tabella 2.1, i filtri *ellittici* e i filtri di *Chebyshev* approssimano bene il modulo di un filtro ideale, ma sono carenti dal punto di vista della linearità

Famiglia	Bontà guadagno	Linearità fase
Bessel	cattiva	buona
Butterworth	media	media
Chebyshev	buona	cattiva
Ellittici	ottima	pessima

Tabella 2.1: Principali famiglie di filtri e loro caratteristiche qualitative

della fase. Per i filtri di *Bessel* è vero il contrario. I filtri di *Butterworth* conducono, invece, ad un'approssimazione accettabile sia per quanto riguarda il modulo sia per quanto riguarda la fase. Un'ulteriore termine di paragone è dato dalla Figura 2.5, in cui è rappresentata la banda passante di un filtro passa-basso appartenente a ciascuna delle famiglie precedentemente elencate. Coerentemente con quanto espresso nella Tabella 2.1, nel grafico si nota come i filtri di Bessel abbiano il risultato peggiore. I filtri ellittici e i filtri di Chebyshev, invece, ottengono i risultati migliori per quanto riguarda la riduzione della banda di transizione. Tuttavia, queste due ultime famiglie presentano consistenti oscillazioni attorno alla banda passante, a differenza dei filtri di Butterworth. Proprio i filtri di Butterworth presentano una riduzione della banda di transizione meno soddisfacente rispetto ai filtri ellittici e ai filtri di Chebyshev, ma decisamente migliore rispetto ai filtri di Bessel. Ciò avviene in maniera analoga per la banda proibita.

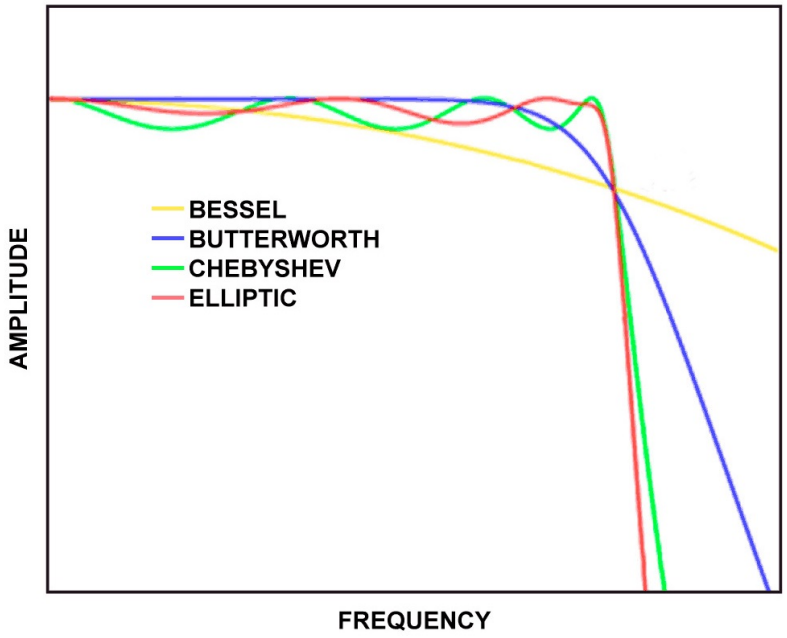


Figura 2.5: Confronto banda passante delle principali famiglie di filtri reali

In conclusione, si è visto come ci sono diversi motivi che motivano un ampio utilizzo in applicazioni reali dei filtri di Butterworth. Questi, infatti, hanno un'approssimazione accettabile sia per quanto riguarda il modulo che per la fase e si differenziano dalle altre famiglie per l'assenza di oscillazioni in corrispondenza della banda passante e della banda proibita. Di conseguenza, la famiglia dei filtri di Butterworth sarà oggetto di un approfondimento a parte nel paragrafo seguente.

Filtri Butterworth Come è stato detto precedentemente, i filtri di Butterworth trovano ampio utilizzo in diversi campi di applicazione perchè consentono di approssimare in modo soddisfacente sia la fase sia il modulo di un filtro ideale. In particolare, una loro caratteristica utile è che si comportano in modo tale da manifestare scarse oscillazioni nella banda passante. Inoltre, non va sottovalutato il fatto che si tratta di filtri di semplice implementazione.
In generale, il modulo della funzione di trasferimento di un filtro di Butterworth è rappresentato dalla seguente equazione:

$$|H(\nu)| = \frac{1}{\left|B_N(i\frac{\nu}{\nu_c})\right|} = \frac{1}{\sqrt{1+(\frac{\nu}{\nu_c})^{2N}}} \tag{2.14}$$

I due parametri che caratterizzano i filtri di Butterworth sono:

1. N, cioè l'ordine del polinomio di Butterworth
2. $\frac{\nu}{\nu_c}$, cioè la frequenza di taglio normalizzata

La formulazione di $B_N(s)$ varia a seconda del fatto che l'ordine sia pari o dispari:

- se N è pari, $B_N(s)$ è il prodotto di $\frac{N}{2}$ polinomi di tipo $s^2 + bs + s$
- se N è dispari, $B_N(s)$ è il prodotto di $\frac{N}{2}$ polinomi di tipo $s^2 + bs + s$ e del polinomio $s + 1$

Una proprietà importante è che tutti gli zeri dei polinomi di Butterworth appartengono al cerchio di raggio unitario [41].
La scelta dell'ordine di un filtro Butterworth è molto importante in fase di progettazione. Nella

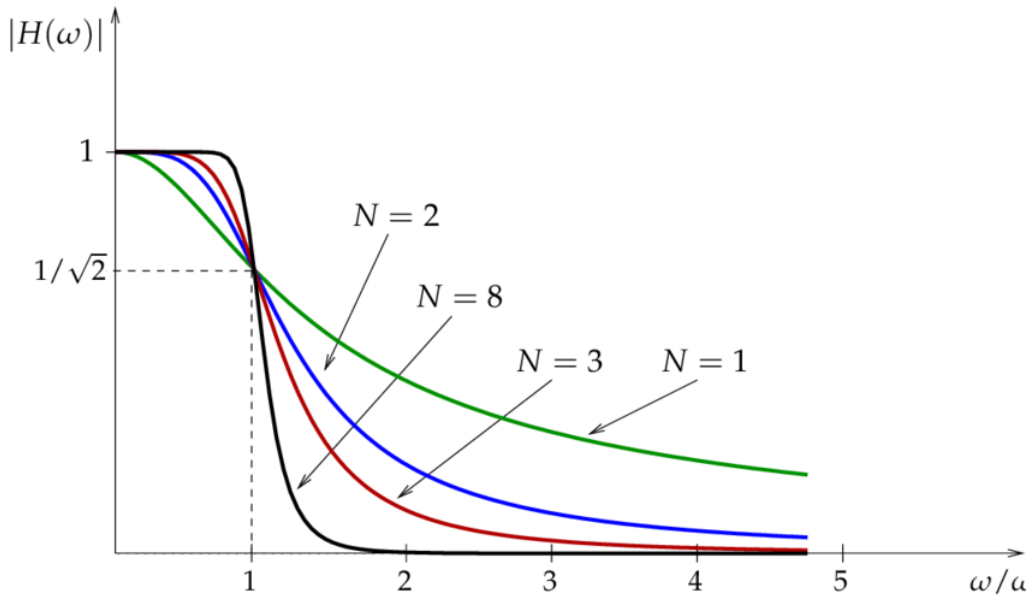


Figura 2.6: Risposta di filtri Butterworth al variare dell'ordine N [44]

Figura 2.6 è rappresentata la funzione di trasferimento di un filtro Butterworth al variare di N.
Si nota che la banda passante non presenta oscillazioni anche per i filtri di ordine basso. Tuttavia, all'aumentare di N migliora l'attenuazione della banda proibita, si registra anche un restringimento della banda di transizione e il guadagno in banda passante si mantiene costante.
La famiglia dei filtri Butterworth è quella che verrà usata successivamente nell'analisi dei segnali di interesse in questo lavoro.

Capitolo 3

Calcolo della SpO_2

3.1 La saturazione del sangue

Oltre ai parametri precedentemente illustrati, un'ulteriore ipotesi è che il riconoscimento dello stato di veglia possa basarsi sul calcolo dell'indice SpO_2, che misura la percentuale di saturazione di ossigeno del sangue. Questo indice può essere stimato attraverso uno strumento chiamato *pulsossimetro*. Si tratta di un piccolo apparecchio da applicare in corrispondenza di un dito della mano o di un lobo dell'orecchio che emette fasci di luce e registra il loro assorbimento. Il suo funzionamento e la sua applicazione saranno descritti nei dettagli in seguito.
Per spiegare meglio il concetto di saturazione del sangue occorre ricordare che l'ossigeno nel sangue si lega all'*emoglobina*, una proteina che ha il compito di trasportarlo dal cuore ai vasi sanguigni periferici. L'emoglobina si presenta nel corpo umano in 4 diverse forme [25]:

- ossiemoglobina (HbO_2): emoglobina satura di ossigeno
- deossiemoglobina (Hb): emoglobina che ha ceduto l'ossigeno
- carbossiemoglobina ($COHb$): emoglobina satura di monossido di carbonio
- metaemoglobina ($MetHb$): una forma di emoglobina alterata

Solo le prime 2 specie di emoglobina sono idonee per il trasporto dell'ossigeno e sono dette *funzionali*. La percentuale di emoglobina non funzionale presente nel sangue è molto ridotta, circa l'$1-2\%$ [25]. Si definisce saturazione del sangue arterioso (SaO_2), il rapporto tra l'ossiemoglobina e l'emoglobina totale presente nel sangue, come rappresentato dalla seguente formula:

$$SaO_2 = HbO_2/(HbO_2 + Hb + MetHb + COHb) \tag{3.1}$$

La misurazione diretta della SaO_2 può essere effettuata tramite un prelievo di sangue arterioso (ABG, Arterial Blood Gas test), che generalmente viene eseguito dall'arteria radiale.
La saturazione del sangue può anche essere stimata, in modo indiretto, attraverso un pulsossimetro, come accennato precedentemente. In questo caso l'indice misurato prende il nome di SpO_2 e nella formulazione si tiene conto solo dell'emoglobina funzionale [27]:

$$SpO_2 = HbO_2/(HbO_2 + Hb) \tag{3.2}$$

Entrambe le tecniche presentate, ABG e utilizzo del pulsossimetro, presentano vantaggi e svantaggi. Un confronto può essere fatto secondo diversi parametri [26]:

- precisione del risultato: è maggiore nel caso dell'ABG. Il pulsossimetro non tiene conto dell'emoglobina non funzionale.
- comfort del paziente: è maggiore nel caso del pulsossimetro. L'ABG può provocare dolore e tumefazioni lievi al paziente.
- autonomia del paziente: è maggiore nel caso del pulsossimetro. L'ABG deve essere eseguita da parte di personale medico specializzato all'interno di strutture adeguate.

Normalmente, i valori di SpO_2 in un soggetto sano si trovano in un range compreso tra il 97% e il 99%. Fumatori abituali possono avere una SpO_2 compresa tra 93% e 95%. Valori inferiori al 90% sono preoccupanti per la salute e devono indurre a test supplementari per confermare una situazione di ipossia e, nel caso, fornire artificialmente ossigeno al paziente. Se la SpO_2 scende al di sotto dell'85%, si parla di ipossia grave, che richiede un immediato intervento medico [27].

3.2 Il pulsossimetro

Come si è detto precedentemente, l'uso del pulsossimetro consente di stimare il valore della SpO_2 in modo non invasivo e senza necessità di supervisione medica.

Figura 3.1: Un pulsossimetro applicato su un dito della mano [29]

Un pulsossimetro è costituito essenzialmente da una sonda che si applica, allo stesso modo di una pinza, ad un dito della mano oppure al lobo di un orecchio. Al suo interno sono posti 2 emettitori che emettono raggi luminosi di diversa lunghezza d'onda, luce rossa (RED) e luce infrarossa (IR), e un rilevatore fotoelettrico. Questa sonda può essere collegata tramite un filo ad un'unità di calcolo che processa e visualizza i segnali raccolti, oppure un piccolo monitor può essere integrato direttamente su di essa.
Nella sezione precedente, si è detto che l'obiettivo del pulsossimetro è stimare la percentuale di ossiemoglobina presente nel sangue. Ciò avviene grazie al diverso comportamento di Hb e HbO_2 quando vengono attraversate da un fascio di luce.
La comprensione intuitiva del fenomeno è data dal grafico presente nella Figura 3.2. In esso,

sull'asse delle ascisse si trova la lunghezza d'onda della luce, espressa in nm, mentre su quello delle ordinate si trova una misura dell'assorbimento della luce. I segmenti paralleli all'asse y di colore rosso e viola rappresentano le lunghezze d'onda corrispondenti, rispettivamente, alla luce rossa e alla luce infrarossa. Le curve di colore rosso e grigio rappresentano l'assorbimento della luce in funzione della lunghezza d'onda da parte, rispettivamente, dell'ossiemoglobina (HbO_2) e della deossiemoglobina (Hb). In corrispondenza del primo segmento verticale, la curva relativa a Hb ha un valore più alto rispetto a quella relativa a HbO_2. In corrispondenza del secondo segmento avviene il contrario. In conclusione, si può notare come l'ossiemoglobina assorba maggiormente la luce infrarossa e, viceversa, la deossiemoglobina assorba maggiormente la luce rossa.

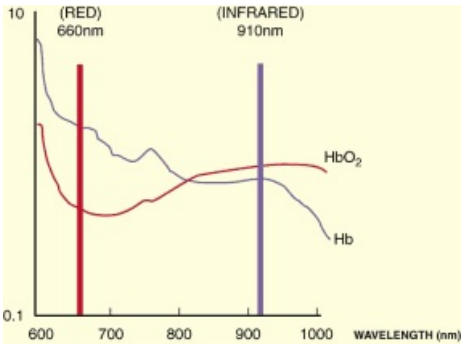


Figura 3.2: Assorbimento delle diverse lunghezze d'onda da parte di Hb e HbO_2 [28]

Volendo procedere con un approccio più analitico, la trasmissione della luce attraverso una soluzione è regolata dalla legge di Beer-Lambert, che qui si richiama:

$$I = I_0 * e^{(-\epsilon * [C] * l)} \tag{3.3}$$

Come si può notare, la trasmissione della luce è funzione di diversi elementi:

- l'intensità della luce incidente (I_0)
- il coefficiente di assorbimento (ϵ), che dipende sia dalla lunghezza d'onda della luce, sia dalla sostanza attraversata da essa.
- la concentrazione della sostanza ($[C]$)
- il cammino geometrico (l), ovvero lo spessore della soluzone attraversata dalla luce

Come è facile immaginare, Hb e HbO_2 presentano due diversi coefficienti di assorbimento, che sono a loro volta due diverse funzioni della lunghezza d'onda della luce incidente. Questa è la chiave che permette di stimare la SpO_2 tramite l'emissione di raggi luminosi [30].
Una volta compresi i principi fisici che sono alla base del funzionamento del pulsossimetro, si può passare alla descrizione del processo grazie a cui si ricava una stima della SpO_2. Quest'ultimo è composto da 3 passaggi:

- l'isolamento delle componenti continue ed alternate dei segnali
- la calibrazione del pulsossimetro
- l'utilizzo di una relazione che estragga il valore della SpO_2 in funzione dei valori misurati dal pulsossimetro

Prima di approfondire il processo schematizzato nel precedente elenco, si ritiene utile evidenziare alcune criticità che si possono incontrare adoperando un pulsossimetro.

Limitazioni del pulsossimetro La stima della SpO_2 che si ottiene tramite il pulsossimetro non è sempre affidabile e può essere influenzata da diversi elementi [27]:

- problematiche relative al soggetto:
  - anomalie fisiologiche dell'apparato cardiovascolare quali bassa pressione, vasocostrizione periferica, bradicardia o aritmie cardiache
  - anomalie nei livelli di emoglobina del paziente, comuni nei fumatori
  - patologie del paziente che causino tremori, come il morbo di Parkinson
- problematiche relative all'ambiente esterno:
  - interferenze di altre apparecchiature elettroniche
  - bassa temperatura
  - eccessiva luminosità
- problematiche relative allo strumento in uso:
  - manutenzione inadeguata, ad esempio scarsa pulizia dei sensori
  - utilizzo improprio, ad esempio in una posizione più alta rispetto al cuore

Queste problematiche hanno effetti diversi sulla misurazione finale. La bassa pressione e la vasocostrizione riducono la potenza del segnale, quindi lo strumento non è sempre in grado di distinguere il flusso sanguigno dal rumore. Aritmie cardiache e bradicardie provocano una forma inusuale dell'onda, inficiando l'accuratezza del pulsossimetro.
Nei fumatori abituali, il livello di emoglobina disfunzionale è di solito consistente. Di conseguenza la stima della SpO_2, che tiene conto solo dell'emoglobina funzionale, potrebbe essere notevolmente distante dal reale valore della SaO_2, che invece tiene conto di tutte le specie di emoglobina.
Tremori o movimenti del soggetto possono disturbare la lettura del segnale da parte del sensore. Lo stesso problema si può verificare a causa di interferenze provenienti dall'ambiente esterno, causate ad esempio da altre strumentazioni elettroniche.
Il pulsossimetro è in conclusione uno strumento con elevate potenzialità, specialmente per la sua praticità e la sua facilità di utilizzo. Necessita però di un uso accorto per ottenere una stima affidabile della SpO_2.

3.3 Processo di calcolo della SpO_2

Come si è detto nella precedente sezione, il processo di calcolo della SpO_2 si articola in diversi passaggi. Il primo consiste nella rielaborazione del segnale campionato dal pulsossimetro e, precisamente, nell'isolamento delle sue componenti.
Il segnale visualizzato dal pulsossimetro è infatti costituito da due componenti: la componente alternata (AC: *Alternating Current*) e la componente continua (DC: *Direct Current*). La prima componente è da attribuire alle pulsazioni del cuore, che pompa il sangue verso i tessuti periferici presso cui è posizionato il pulsossimetro. La componente DC, invece, è relativa al volume dei tessuti sottostanti, che varia più lentamente a seguito, ad esempio, dell'attività respiratoria [31].
Una volta estrapolate queste 2 componenti, di norma si calcola il loro rapporto. In letteratura si indica con la lettera R ciò che viene definito come *ratio dei ratio* e che è frutto di un'ulteriore elaborazione. La formulazione di R è la seguente [32]:

$$R = \frac{\frac{AC_{RED}}{DC_{RED}}}{\frac{AC_{IR}}{DC_{IR}}} \tag{3.4}$$

Essa combina il rapporto delle componenti dei 2 segnali che sono frutto dell'azione degli emettitori presenti nel pulsossimetro.
Il passaggio dal rapporto R all'indice SpO_2 non è nè immediato nè indipendente dal pulsossimetro che si sta utilizzando. É infatti necessario un passaggio intermedio che consiste nella calibrazione dello strumento. Si tratta di un'operazione che tipicamente viene eseguita su ogni modello di pulsossimetro prima della sua commercializzazione. Lo scopo di questa operazione è definire una relazione che faccia corrispondere uno specifico valore di SpO_2 ad ogni valore di R rilevato.
Ci sono diversi modi di calibrare un pulsossimetro, in particolare la calibrazione può essere:

- invasiva: vengono prelevati campioni di sangue da un gruppo di volontari
- non invasiva: non è necessario coinvolgere volontari

Nel primo caso, i soggetti selezionati vengono portati artificialmente in condizioni di ipossia, tramite la respirazione di appositi gas. Questo fa sì che la calibrazione avvenga solo per valori di SpO_2 maggiori di una certa soglia, generalmente il 70%. Infatti, sarebbe pericoloso forzare i volontari a sperimentare una condizione di maggior carenza di ossigeno [33].
La precisione del pulsossimetro per valori bassi di SpO_2 non è quindi sempre garantita. Inoltre, un'altra possibile limitazione è legata al fatto che l'assorbimento della luce è suscettibile di variazioni a seconda della diversa pigmentazione della pelle. Le misurazioni potrebbero quindi essere fuorvianti qualora il soggetto appartenesse ad una popolazione diversa da quella utilizzata per calibrare i parametri [32].
Per ovviare alle problematiche di sicurezza, di affidabilità e di costo relative alla calibrazione diretta dei pulsossimetri, in letteratura sono stati proposti alcuni metodi non invasivi di calibrazione [34]. Ad esempio, tramite l'utilizzo di un dito artificiale inondato da fasci di luce, oppure tramite un pulsossimetro di riferimento già calibrato.

Una volta che si hanno a disposizione le componenti continua ed alternata dei segnali RED e IR e un pulsossimetro correttamente calibrato, occorre esplicitare la relazione che lega il rapporto R, ottenuto tramite l'equazione 3.4, con il valore della SpO_2.
In letteratura si trovano numerose proposte utili a formalizzare tale relazione. Quella che tiene maggiormente conto degli aspetti fisici sottostanti è la seguente [38]:

$$SpO_2 = \frac{\epsilon_d - R(l_2/l_1)\epsilon_d}{R(l_2/l_1)(\epsilon_o - \epsilon_d) + (\epsilon_d - \epsilon_o)} \tag{3.5}$$

In essa, oltre a R, entrano in gioco diversi parametri:

- (ϵ_o): coefficiente di assorbimento della luce dell'emoglobina ossigenata
- (ϵ_d): coefficiente di assorbimento della luce dell'emoglobina deossigenata
- l_1 e l_2: lo spazio percorso dalla luce dalla sua sorgente alla sostanza che la assorbe

Questa formula ha il pregio di rispecchiare in maniera fedele la legge di Beer-Lambert (3.3), di cui si è parlato precedentemente. Tuttavia, introduce il problema del calcolo preciso dei coefficienti l_1 e l_2, che generalmente sono assunti costanti ed indipendenti dal soggetto in esame per semplicità. Una stima errata di questi parametri, però, può portare ad una misurazione della SpO_2 inadeguata [38]. Si può quindi valutare l'opportunità di utilizzare una relazione più semplice, ad esempio la seguente [38]:

$$SpO_2 = \frac{K_1 - K_2 * R}{K_3 - K_4 * R} \tag{3.6}$$

In essa sono presenti 4 parametri $(K_1, ..., K_4)$ che costituiscono il risultato del procedimento di calibrazione spiegato prima.

Un'ulteriore proposta è quella di considerare la relazione tra R e SpO_2 come lineare, come nella seguente equazione [32]:

$$SpO_2 = K_1 + K_2 * R \tag{3.7}$$

in cui sono presenti soltanto 2 parametri da calibrare.
Infine, la proposta più semplice è quella di considerare una relazione di proporzionalità diretta tra R e SpO_2 [37]:

$$SpO_2 = K * R \tag{3.8}$$

con K che è sempre un parametro da stabilire in seguito a calibrazione.
La relazione da utilizzare, quindi, deve essere scelta in modo da bilanciare la ricerca della precisione con la necessità di semplicità. Tipicamente, in letteratura viene utilizzata una relazione di tipo lineare, con un andamento che è sintetizzato dalla figura 3.3. Differenze tra un pulsossimetro ed un altro sono dovute alla calibrazione.

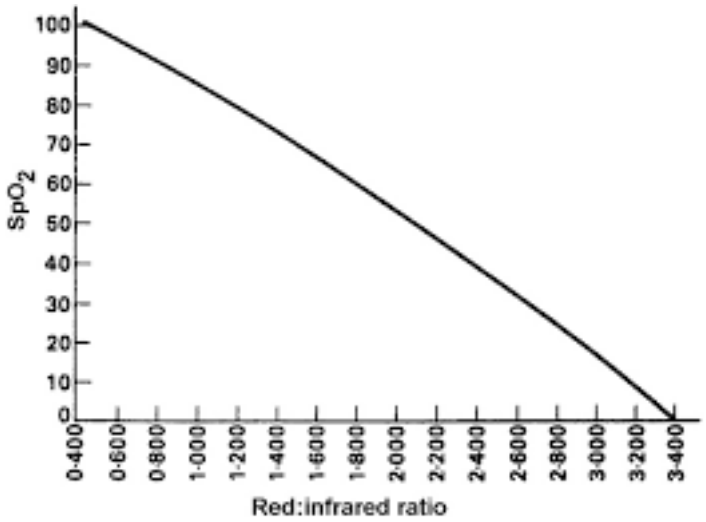


Figura 3.3: Relazione empirica tra R e SpO_2 [33]

A titolo di esempio e per rendere l'idea dell'ordine di grandezza e del segno dei parametri, nell'equazione seguente si sono sostituiti i parametri con dei valori numerici tratti da diversi importanti studi [25, 35, 36]:

$$SpO_2 = 110 - 25 * R \tag{3.9}$$

In conclusione, il processo di estrazione del valore della SpO_2 tramite l'uso del pulsossimetro non è banale. Richiede anzi grande attenzione in ogni suo passaggio e risulta fondamentale per ottenere risultati affidabili.

3.4 Implementazione pratica del calcolo della SpO_2

Seguendo i passaggi descritti nella precedente sezione, nell'ambito di questo lavoro di tesi è stato preparato un framework orientato al raggiungimento dell'obiettivo del calcolo dell'indice SpO_2 a partire dalla rilevazione di segnali fisiologici tramite pulsossimetro. La validazione di tale framework non è stata però possibile a causa della mancanza di dati idonei allo scopo. Nella presente sezione, verranno comunque discussi i passaggi salienti del procedimento implementato.

Come primo punto, non si può prescindere da una valutazione della regolarità del segnale a disposizione. La Figura 3.4 rappresenta l'andamento atteso, a titolo di esempio, del segnale $IRED$, caratterizzato da periodo regolare ed oscillazioni di ampiezza approssimativamente costante.

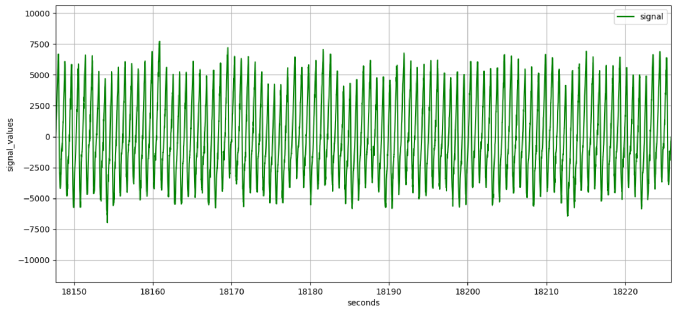


Figura 3.4: Andamento regolare del segnale $IRED$

Tuttavia, durante le operazioni di campionamento dei segnali possono verificarsi movimenti del soggetto oppure possono intervenire elementi di disturbo esterni. Il risultato è un segnale deformato come rappresentato in Figura 3.5. É evidente che un segnale di questo tipo non porterà al calcolo di un indice SpO_2 completamente affidabile.

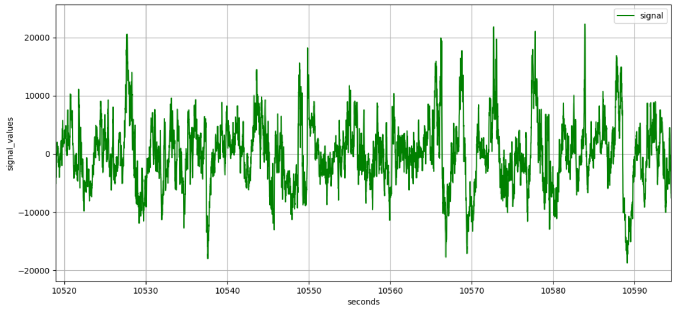


Figura 3.5: Andamento disturbato del segnale $IRED$

Successivamente, si procede all'identificazione delle componeneti continue ed alternate dei segnali RED e $IRED$. Infatti, come è stato esposto precedentemente, le formule per il calcolo della SpO_2 richiedono a monte il calcolo del rapporto R, che è stato definito nell'equazione 3.4. A sua volta, il calcolo di R è basato sui valori delle componenti alternata e continua dei segnali RED e $IRED$. Operativamente, per ottenere i valori di AC e DC sono stati presi in considerazione due approcci:

- Si considera AC come l'ampiezza dell'onda. Tutto ciò che non è AC si considera DC
- Si isolano AC e DC con l'utilizzo di appositi filtri

Per l'implementazione del primo approccio preliminarmente si procede all'individuazione dei picchi del segnale. Si è scelto di dividere il segnale in finestre di ampiezza costante e pari a 1 secondo, che corrisponde al periodo teorico del segnale. In queste finestre si cerca il massimo e il minimo locale. Nel caso in cui una finestra non contenga al suo interno massimo e minimo, a causa di qualsiasi tipo

di irregolarità locale del segnale, non viene calcolato l'indice SpO_2 in quella finestra. Un approccio del genere è efficace se il numero di finestre dal comportamento regolare è preponderante.
La Figura 3.6 rappresenta dal punto di vista grafico il primo approccio di isolamento delle componenti. In questo caso con 1 è stato indicato il valore del segnale in corrispondenza del suo minimo e con 2 il valore in corrispondenza del massimo. La differenza tra i due valori costituisce la componente alternata AC, mentre 1 corrisponde al valore della componente continua DC.

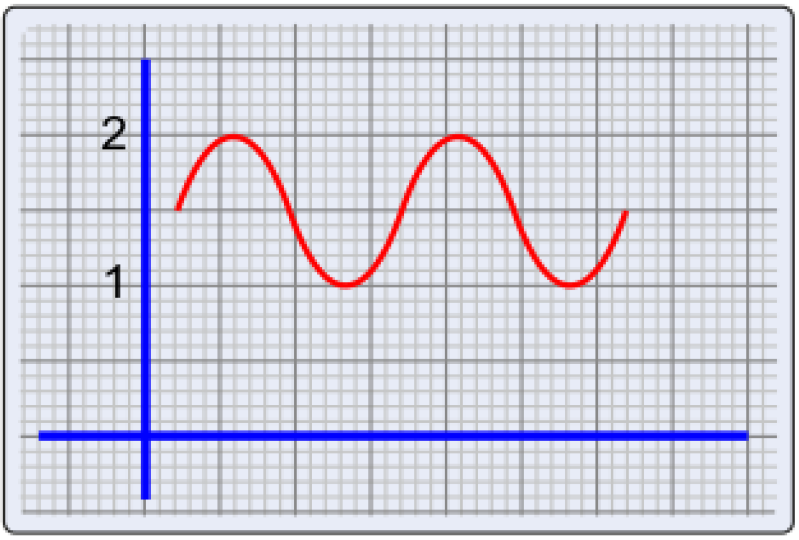


Figura 3.6: Primo approccio di isolamento delle componenti AC e DC

Il secondo approccio, in cui si prevede l'utilizzo dei filtri, è tuttavia maggiormente rigoroso. In particolare, è stato implementato un filtro passa-basso per isolare la componente DC ed un filtro passa-alto per isolare la componente AC. Questo perchè, come è stato descritto nel Capitolo 2, tramite l'applicazione del filtro passa-basso vengono eliminate tutte le componenti in frequenza superiori ad una certa soglia, isolando così la componente continua. L'opposto avviene applicando il passa-alto per isolare la componente alternata. A riprova di ciò, la Figura 3.7 e la Figura 3.8 rappresentano, rispettivamente, l'effetto dell'applicazione di un filtro passa-basso e di un filtro passa-alto su un segnale.

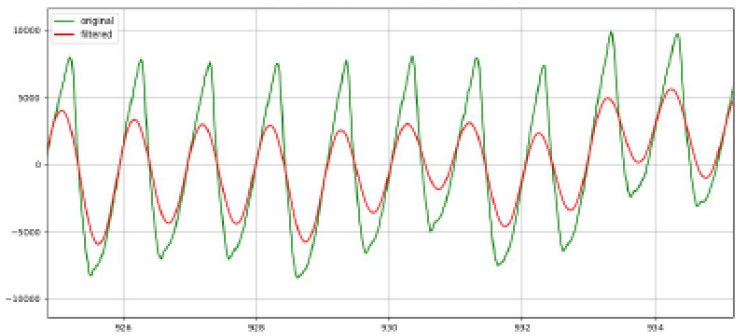


Figura 3.7: Effetto di un filtro passa-basso

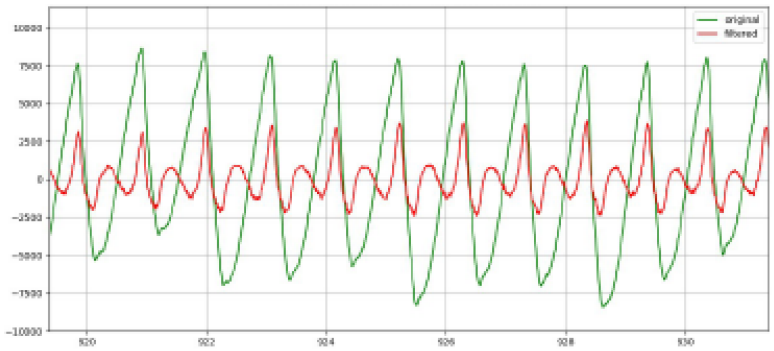


Figura 3.8: Effetto di un filtro passa-alto

Capitolo 4

Operazioni di rielaborazione dei segnali

4.1 Descrizione del dataset, degli strumenti utilizzati e del framework di analisi

I dati che sono stati analizzati nell'ambito di questo lavoro di tesi provengono da un campione eterogeneo di soggetti che sono stati sottoposti ad una sessione di analisi polisonnografica. Questa analisi è stata svolta secondo le metodologie che sono state illustrate nel Capitolo 1. L'acquisizione dei dati ha riguardato una vasta gamma di segnali fisiologici in modo tale da garantire, in un secondo momento, la possibilità di effettuare differenti analisi su differenti specifici segnali di interesse.
Il formato utilizzato per l'acquisizione dei dati è il formato *.edf* (European Data Format). Si tratta di un formato di cui viene fatto largo uso per la memorizzazione di segnali biomedici e che consente di memorizzare efficacemente immagini e metadati.
L'elaborazione del dataset ha richiesto l'impiego di diversi software, che sono intervenuti in differenti fasi del lavoro a seconda delle rispettive caratteristiche. Di seguito viene data una sintetica descrizione di ciascuno dei tool utilizzati.

EDFbrowser I file salvati in formato "edf", come quelli che compongono il dataset in esame, richiedono un apposito software per la visualizzazione. Tra tutti quelli esistenti si è scelto di utilizzare il software *EDFbrowser* [47].
Si tratta di un software open-source che offre molteplici possibilità di visualizzazione e di manipolazione dei segnali. Una schermata di un processo in esecuzione è mostrata in Figura 4.1. In particolare, molto efficaci risultano essere le funzionalità relative alla rappresentazione grafica dei segnali. EDFbrowser consente infatti di navigare lungo l'intera registrazione e di riscalarla sull'asse del tempo e sull'asse dell'ampiezza. É anche possibile visualizzare contemporaneamente diversi segnali e operare così un confronto tra di essi.
Inoltre, il programma offre anche funzioni che provvedono al filtraggio e ad altre operazioni sui segnali in esame.
La fase in cui si è prevalentemente fatto uso di questo software è la fase di analisi preliminare. Infatti, l'utilizzo di EDFbrowser è stato fondamentale per prendere confidenza con i segnali oggetto di studio. Da un lato è stato possibile osservare l'andamento regolare dei segnali, il loro periodo e la distribuzione dei loro picchi. Dall'altro, è stato possibile valutare la bontà e la qualità delle registrazioni in possesso e rilevare alcune criticità manifestate. Inoltre, EDFbrowser è stato

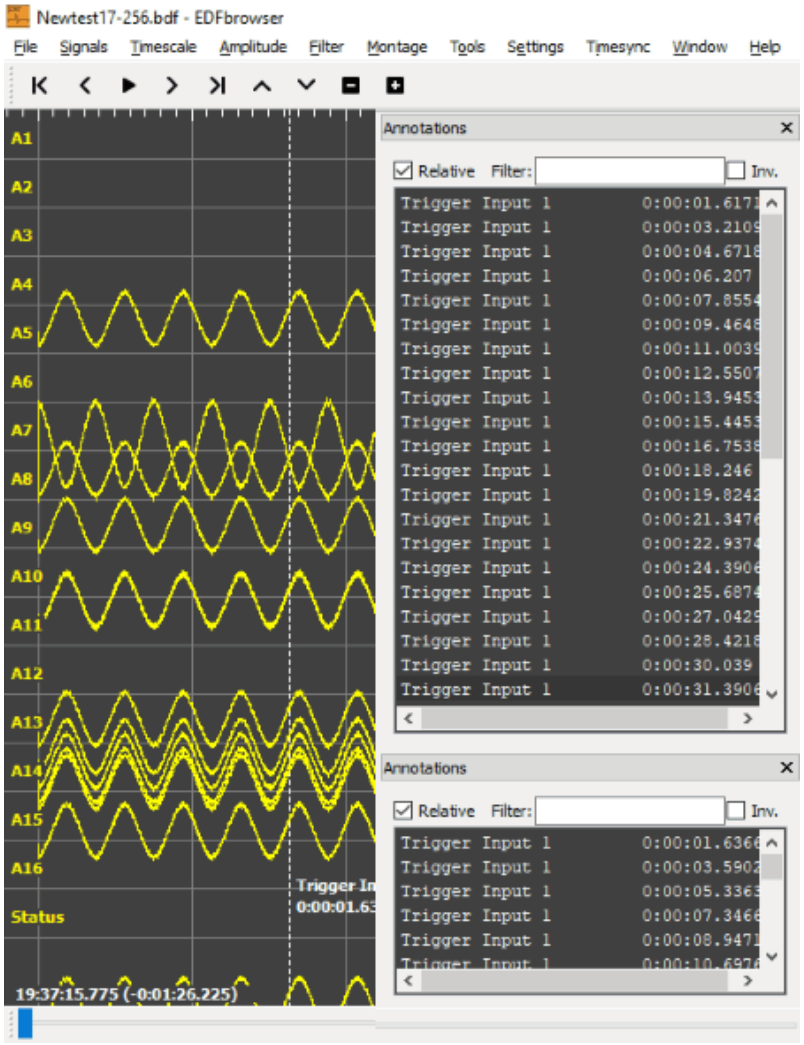


Figura 4.1: EDFbrowser

utilizzato anche successivamente, per ottenere un primo riscontro qualitativo circa le differenze nell'andamento del segnale tra le varie fasi.

Spyder A seguito della fase di analisi preliminare è stato necessario implementare del codice per svolgere tutte le operazioni necessarie di rielaborazione del segnale.
Il linguaggio di programmazione in cui è stato scritto il codice è *Python*. Si tratta di un linguaggio di programmazione ad alto livello ed orientato agli oggetti, caratterizzato da una licenza open-source che ne consente la libera modifica e redistribuzione.
Come ambiente di sviluppo per la produzione del codice si è scelto di lavorare tramite *Spyder* [48], un tool progettato appositamente per lavorare con Python e di cui in Figura 4.2 è mostrata una schermata a titolo di esempio.
Spyder possiede svariate caratteristiche che lo rendono un ambiente di sviluppo particolarmente adatto per applicazioni scientifiche. In esso sono implementate funzionalità che supportano lo sviluppo, l'analisi e il debug del codice. Inoltre, sono integrate anche funzionalità di visualizzazione

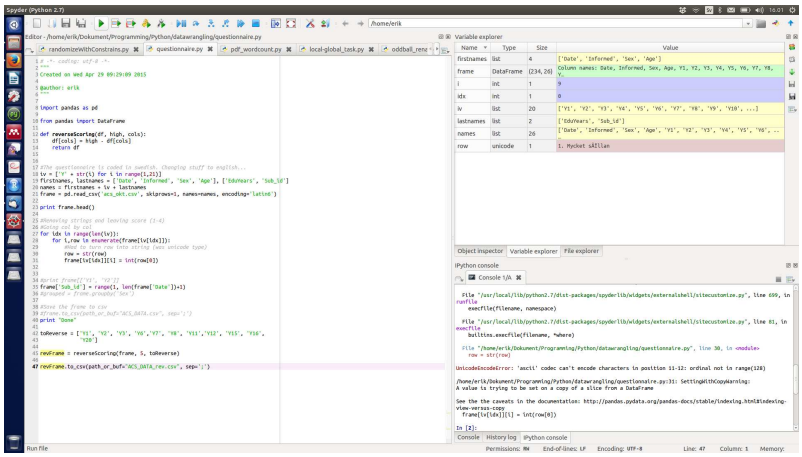


Figura 4.2: Spyder

e rappresentazione grafica.
In particolare, è presente un editor con funzioni di compilazione del codice, di analisi run-time e di navigazione all'interno del codice stesso, una console dotata di interfaccia grafica, un debugger eseguibile in maniera interattiva, oltre ad altre funzioni di visualizzazione e di interazione con le variabili e di accesso immediato alla documentazione di supporto.
La fase in cui si è fatto uso di Spyder è l'intera fase di rielaborazione del segnale, in cui il dataset a disposizione è stato processato e preparato alla successiva fase di implementazione di algoritmi di machine learning.

RapidMiner Nell'ambito del presente lavoro è stato necessario utilizzare anche software che consentono l'implementazione degli algoritmi di machine learning più adatti allo scopo prefissato. Uno dei programmi che sono stati impiegati è *RapidMiner* [49], di cui in Figura 4.3 viene mostrata la visualizzazione di un esempio di processo.
RapidMiner integra al suo interno diverse funzionalità che possono provvedere all'intero processo di elaborazione di un dataset, dalle operazioni di pre-processing alla costruzione e alla validazione di un modello statistico, fino alla rappresentazione e all'esportazione dei risultati ottenuti. Tutte queste funzioni sono presentate attraverso un'interfaccia grafica progettata per una immediata interazione con l'utente.
In particolare, tramite RapidMiner è possibile l'accesso diretto a file memorizzati nei formati più diffusi, come ad esempio formati testuali e tabulari. Sono presenti poi funzioni che consentono di effettuare operazioni preliminari sul dataset come gestione di dati mancanti, riscalamenti o normalizzazioni. Infine, si trova già implementata una vasta gamma di modelli accompagnata da metodi per la loro validazione. I risultati finali possono poi essere esportati in svariati formati di facile accesso.
Come accennato precedentemente, si è fatto uso di RapidMiner in modo particolare nella fase finale del lavoro, la fase in cui si è passati all'implementazione di modelli e all'analisi dei risultati ottenuti.

Weka RapidMiner non è l'unico software che è stato utilizzato per l'implementazione di modelli statistici. Un altro programma di cui si è fatto uso è *Weka* (Waikato Environment for Knowledge Analysis) [50], una sua schermata di esempio è visibile in Figura 4.4.
Similmente a RapidMiner, Weka è un tool che viene comunemente utilizzato per la progettazione

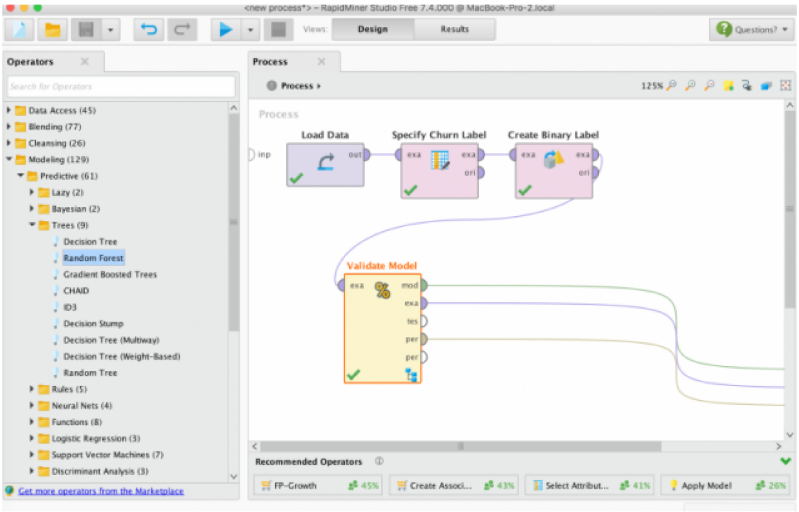


Figura 4.3: RapidMiner

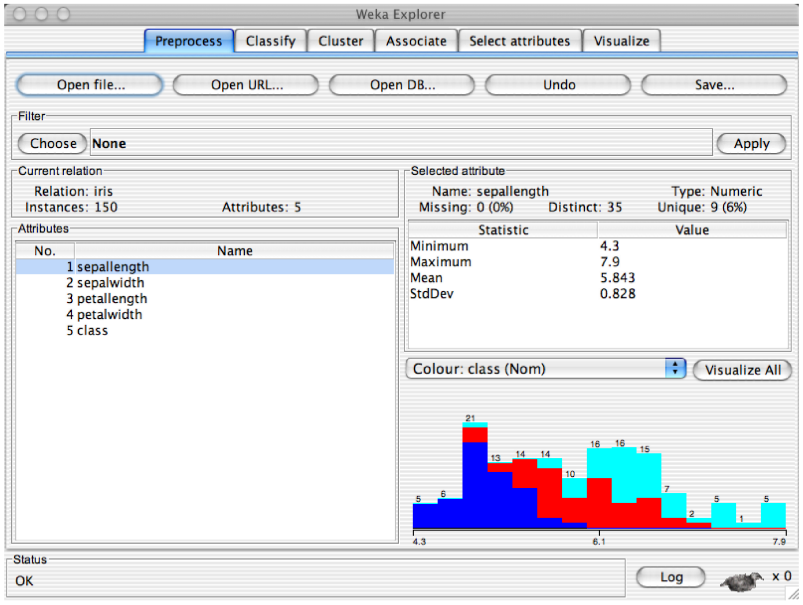


Figura 4.4: Weka

di vari algoritmi di statistica, ad esempio algoritmi di classificazione e di regressione o regole di associazione.
Anche Weka è dotato di funzionalità che riguardano la preparazione del dataset all'analisi vera e propria, di un'interfaccia grafica e di strumenti di visualizzazione grafica volti a semplificare l'interpretazione e l'estrapolazione di informazioni dal dataset in esame.
La fase in cui si è usufruito di questo software è sempre quella finale in cui ci si è dedicati all'estrazione di informazioni dal dataset tramite algoritmi di machine learning.
L'utilizzo di Weka è stato complementare a quello di RapidMiner, ovvero si è cercato di sfruttare al meglio le diverse opportunità e potenzialità offerte da questi due tool utilizzando di volta in

volta quello più idoneo allo specifico obiettivo che si intendeva raggiungere.

Descrizione del framework di analisi del dataset L'obiettivo che ci si prefigge nell'analisi del presente dataset è quello di giungere, tramite algoritmi di classificazione, al riconoscimento automatico della fase del segnale immediatamente precedente all'istante di addormentamento del paziente. Questa fase verrà definita in seguito come *fase di addormentamento*, in contrapposizione alla *fase di veglia*.

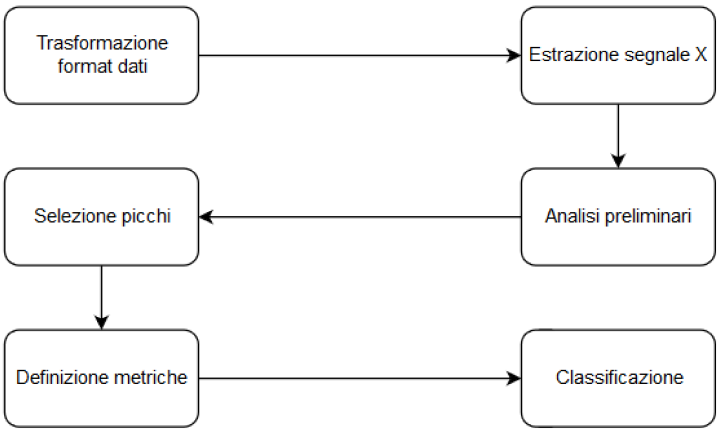


Figura 4.5: Rappresentazione del framework di elaborazione dei segnali

Il raggiungimento di questo obiettivo passa attraverso diversi step che possono essere schematizzati nella Figura 4.5. Ognuno dei blocchi sarà descritto approfonditamente nel prosieguo della trattazione, ma una prima sintetica descrizione è fornita nel presente paragrafo.
Per quanto riguarda il primo blocco, si è già detto precedentemente che i dati sono disponibili nel formato ".edf". Tale formato consente un'efficace visualizzazione grafica tramite EDFbrowser ma non è idoneo ad essere processato tramite Spyder.
Di conseguenza, si è dovuto procedere alla conversione dei file originali in file testuali, ovvero in formato *.txt*. Parallelamente, si sono separate in file distinti le informazioni relative ai diversi segnali registrati e anche i metadati che forniscono interessanti informazioni riguardo la composizione del dataset ma che non sono strettamente connessi alle registrazioni in sè.
Successivamente, si è proceduto ad isolare lo specifico segnale oggetto di interesse. Su di esso sono state condotte alcune analisi preliminari aventi come oggetto la regolarità e la qualità delle registrazioni a disposizione.
L'elaborazione dei segnali in esame ha come scopo prevalente la preparazione dei segnali alla caratterizzazione tramite le metriche. In particolare, gli sforzi sono stati volti all'implementazione di un algoritmo di selezione dei picchi del segnale. La corretta identificazione dei picchi è di grande importanza poichè su di essa sono basate le metriche che caratterizzano il segnale. Tuttavia, questa operazione è resa più complessa da alcune irregolarità del segnale di cui si discuterà nel dettaglio in seguito.
La definizione delle metriche deve essere sviluppata in modo tale da caratterizzare efficacemente il segnale e da mettere in luce le differenze tra la fase di veglia e la fase di addormentamento.
La base di partenza è costituita da osservazioni qualitative relative a differenze riscontrate nel segnale tra queste due fasi. In seguito, parametri importanti di queste metriche, come l'ampiezza

delle finestre su cui devono essere applicate, sono stati ottimizzati in seguito ad operazioni di fitting.
Infine, durante la fase di implementazione di algoritmi di classificazione, sono stati modellati e testati numerosi algoritmi variando i loro parametri con l'obiettivo di trovare le configurazioni con cui si ottengono i migliori risultati, facendo particolare attenzione ad aspetti teorici come la valutazione della bontà delle procedure proposte ed il rischio di *overfitting*.
Questa operazione è stata possibile tramite due dei software precedentemente citati, ovvero RapidMiner e Weka. L'analisi degli output forniti da questi programmi ha permesso di elaborare alcune considerazioni finali a coronamento del lavoro svolto.

4.2 Operazioni sul dataset originale

Come accennato precedentemente, prima di poter procedere con la fase di elaborazione dei segnali vera e propria è stato necessario effettuare alcune operazioni preliminari. La prima di queste consiste nel trasformare i file in input in un formato di più facile accesso.
Infatti, il formato "edf" è molto pratico per visualizzare graficamente i segnali, ma non lo è altrettanto per effettuare su di essi le elaborazioni necessarie. In particolare, per lavorare con Python, è molto utile utilizzare il formato testuale.
Inoltre, in un unico file "edf" sono contenute informazioni relative a tutti i segnali registrati durante l'analisi polisonnografica. Memorizzati nello stesso file sono anche presenti metadati che caratterizzano alcuni aspetti generali del dataset, in aggiunta ai valori dei segnali campionati.
Queste considerazioni hanno fatto sì che fosse necessario effettuare alcune operazioni distinte sul dataset originale. Esse sono elencate di seguito:

- conversione dei file ".edf" in file ".txt"
- separazione delle divere tipologie di informazione contenute in file distinti
- estrazione dei segnali di specifico interesse
- progettazione di un'adeguata struttura dati per la memorizzazione e l'utilizzo locale

Un file eseguibile *edftoascii.exe* [51] è stato utilizzato per compiere le prime due operazioni. Questo programma riceve in input un file ".edf" e lo converte in un file ".txt" facendo sì che tutte le informazioni contenute nel file originale vengano conservate.
Al termine dell'esecuzione di questo passaggio, non soltanto i dati relativi al soggetto in esame sono stati convertiti ma sono anche stati separati in quattro distinti file testuali con il criterio di accomunare informazioni simili per tipologia. Nel dettaglio, la suddivisione viene fatta come segue:

- *Annotations.txt*: contiene eventuali annotazioni relative alla registrazione in esame
- *Data.txt*: contiene, per ogni istante di tempo, i valori di tutti i segnali fisiologici raccolti
- *Header.txt*: contiene l'header della registrazione in esame
- *Signals.txt*: contiene informazioni relative ai segnali raccolti, ad esempio unità di misura e range in cui sono contenuti i valori

Il passaggio successivo prevede, a partire dal file di tipo *Data*, l'estrazione dei valori del segnale o dei segnali che si vogliono processare successivamente.
La struttura di questi file conclusivi prevede che ad ogni istante di campionamento del segnale corrisponda il valore in ampiezza del segnale stesso. Al termine di questa operazione sono disponibili per la fase successiva un numero di file pari al numero di segnali distinti che si intende analizzare

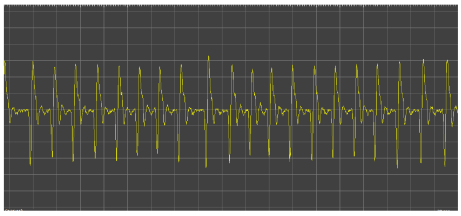
Nel momento del passaggio all'elaborazione dei segnali tramite Spyder, tale struttura dati viene mantenuta per un utilizzo locale dei segnali. Ciò è possibile grazie all'implementazione di una struttura di tipo chiave-valore in cui i dati sono indicizzati. Python mette ad esempio a disposizione una collezione definita come *Dictionary*.
A questo punto, una volta che i segnali di interesse sono stati resi facilmente accessibili si può procedere con le necessarie rielaborazioni per le quali è stato realizzato apposito codice.

4.3 Ricerca dei picchi del segnale

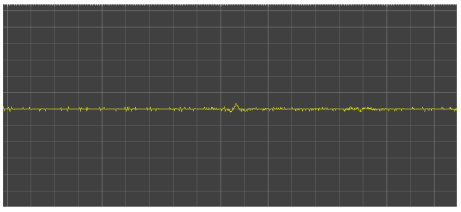
Per prima cosa, è necessario implementare un algoritmo che si occupi di selezionare i picchi del segnale in esame. Si tratta di un obiettivo fondamentale poichè le metriche che si definiranno successivamente sono basate proprio sui picchi.
La selezione automatica dei picchi di un segnale è un obiettivo il cui coefficiente di difficoltà dipende dalla forma standard dell'onda e dalla qualità di acquisizione del segnale stesso. Tanto più il segnale è regolare ed esente da interferenze e disturbi, tanto più è raro incorrere in errori di selezione. Nel caso in esame, il segnale presenta alcune irregolarità di cui sarà necessario tenere conto.
Fatte queste considerazioni si delinea il processo di selezione dei picchi, che si articola in diversi passaggi:

- Analisi preliminare del segnale in esame
- Scrematura preliminare dei picchi
- Miglioramento della selezione dei picchi tramite filtraggio del segnale
- Raffinamento della selezione effettuata

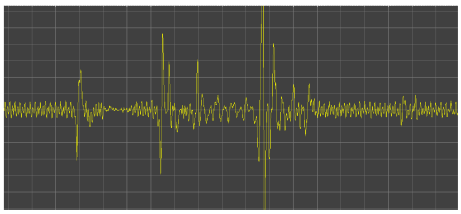
Per quanto riguarda il primo punto, l'andamento regolare del segnale in esame, che corrisponde a quanto si trova in letteratura, è rappresentato nella Figura 4.6a. Come si può notare, nell'intervallo di tempo rappresentato l'onda è caratterizzata da un periodo complessivamente regolare e da massimi e minimi locali che si differenziano notevolmente in ampiezza rispetto al resto del segnale.
Tuttavia, le registrazioni a disposizione non mantengono sempre il comportamento sperato. A titolo di esempio, si riportano alcuni intervalli del segnale, corrispondenti alle figure che vanno dalla Figura 4.6b alla Figura 4.6d, in cui sono rappresentate le più frequenti anomalie riscontrate. Come si può notare, le irregolarità che possono presentarsi si differenziano notevolmente tra loro e comportano differenti problemi nella selezione dei picchi. Ad esempio, nella figura 4.6b, il segnale si presenta estremamente piatto, a tal punto che diventa molto complicato affermare addirittura l'esistenza stessa di picchi.
Nella figura 4.6c, invece, il segnale assume valori nettamente sproporzionati rispetto al suo andamento naturale. Questi *outliers* sono ovviamente facilmente contrassegnabili come picchi ma, come si può notare nell'esempio, tendenzialmente alterano la normale alternanza e la normale distanza tra picchi consecutivi.
Infine, nella figura 4.6d, il segnale sembra perdere il suo caratteristico andamento e risulta complessa l'individuazione dei picchi, poichè si formano dei picchi intermedi che ne disturbano l'operazione di selezione.
Non è facile individuare la causa di questi cambiamenti nella forma del segnale. Si può ipotizzare che siano dovuti a movimenti del soggetto durante la fase di acquisizione dei segnali che possono disturbare la rilevazione del segnale o spostare gli strumenti di acquisizione dalla corretta posizione.
La presenza delle anomalie descritte precedentemente richiede di essere gestita con tecniche idonee. L'analisi qualitativa effettuata suggerisce l'utilizzo di un filtro passa-basso che, come descritto nel Capitolo 2, consentirebbe la rimozione dei disturbi da cui il segnale è affetto. Ciò si prevede abbia



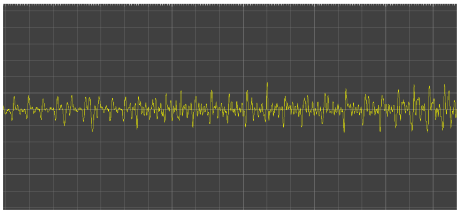
(a) Esempio di intervallo ad andamento regolare del segnale in esame



(b) Esempio di irregolarità nell'andamento del segnale in esame (1)



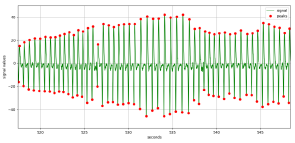
(c) Esempio di irregolarità nell'andamento del segnale in esame (2)



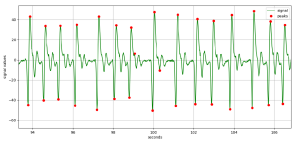
(d) Esempio di irregolarità nell'andamento del segnale in esame (3)

Figura 4.6: Intervalli di segnale regolari ed irregolari

un effetto positivo sulla selezione dei picchi soprattutto per quanto riguarda l'anomalia in Figura 4.6c e, in misura minore, per l'anomalia in Figura 4.6d.
In intervalli corrispondenti alla Figura 4.6b, invece, l'applicazione dei filtri si prevede che induca oscillazioni artificiali e non corrispondenti alla realtà. La presenza di quest'ultimo tipo di irregolarità è comunque quantitativamente limitata e non tale da inficiare la selezione dei picchi nel suo complesso.
Al termine dell'analisi preliminare, si può procedere con l'implementazione dell'algoritmo per la selezione dei picchi. Per ottenerne una prima scrematura si è ricorso ad alcune funzioni disponibili nelle librerie predefinite di Python. In particolare, la libreria *peakutils* di Python ne fornisce alcune che sono utili per raggiungere lo scopo prefissato. L'applicazione del codice basato su queste funzioni conduce a risultati generalmente accettabili in alcune zone del segnale, un esempio è raffigurato nella Figura 4.7a. Analizzando però il segnale nel suo complesso, il risultato ottenuto non risulta del tutto soddisfacente e necessita di una consistente rielaborazione. Le principali problematiche che si presentano vengono mostrate nelle figure successive. In particolare, nella Figura 4.7b si nota che sono stati contrassegnati come picchi alcuni picchi locali del segnale. Anche nella Figura 4.7c è presente un picco indesiderato oltre ai picchi correttamente selezionati. In questo caso, il picco in eccesso si trova molto vicino ad un picco reale.
Un primo miglioramento di questo risultato può essere ottenuto grazie al filtraggio del segnale. In particolare, viene applicato al segnale in esame un filtro *passa-basso* della famiglia dei filtri *Butterworth* che ha l'effetto di eliminare le frequenze superiori ad una certa frequenza di taglio, come descritto nel Capitolo 2. Il risultato è che la forma dell'onda del segnale si presenta più liscia ed affetta in misura minore da disturbi e rumore. Il confronto tra il segnale originale ed il segnale filtrato è visibile nella Figura 4.7d. Una volta che si ha a disposizione il segnale filtrato si possono cercare i picchi prendendo in esame quest'ultimo. La rilevazione dei picchi sul segnale filtrato permette di risolvere alcune problematiche riscontrate considerando esclusivamente il segnale originale. Questo avviene proprio grazie alla forma più liscia dell'onda, dovuta al fatto che, con l'azione del filtro, sono state rimosse componenti del segnale ad alta frequenza non tipiche del segnale in esame ma dovute ad alterazioni occorse in fase di acquisizione.

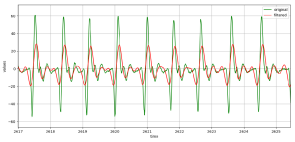


(a) Esempio di corretta classificazione dei picchi

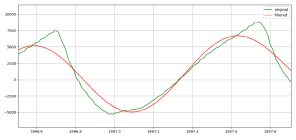


(b) Esempio di anomalia nella rilevazione dei picchi (I)

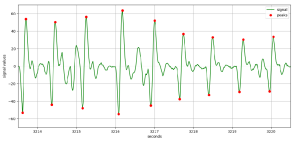
(c) Esempio di anomalia nella rilevazione dei picchi (II)



(d) Confronto tra il segnale originale ed il segnale filtrato con un filtro passa-basso



(e) Esempio di distorsione temporale dovuta all'azione del filtro passa-basso



(f) Selezione dei picchi al termine del processo di elaborazione dei picchi

Figura 4.7: Varie fasi del processo di selezione dei picchi

Successivamente si procede ad eseguire un'operazione di ricollocamento dei picchi trovati sul segnale originale. Infatti, l'azione del filtro provoca una distorsione del segnale sull'asse temporale, come rappresentato in Figura 4.7e.
L'implementazione di un'apposita funzione consente di completare questo passaggio e di valutare la bontà della nuova selezione dei picchi. Tale valutazione viene compiuta confrontando la prima selezione dei picchi con la nuova selezione in corrispondenza degli intervalli dove si rilevavano le maggiori criticità.
Il filtraggio del segnale si rivela efficace nella risoluzione del problema della tipologia mostrata nella Figura 4.7b.
Lo stesso procedimento però non è efficace nella risoluzione del problema rappresentato nella figura 4.7c. Ciò avviene perchè l'origine di quest'ultima anomalia è differente e non va ricercata in disturbi del segnale a cui si può porre rimedio tramite l'uso del filtro, bensì ad una perdita del segnale.
Per ovviare a quest'ulteriore problematica è stata implementata un'apposita funzione che ha il compito di effettuare un ulteriore raffinamento relativamente all'insieme dei picchi selezionato. Questa funzione si occupa di individuare le coppie di picchi ravvicinati dello stesso segno rilevate e di eliminare il più piccolo in valore assoluto tra i due picchi che le compongono. Il risultato dell'applicazione di quest'ultimo passaggio è visibile nella Figura 4.7f, in cui è rappresentato lo stesso intervallo temporale raffigurato nella Figura 4.7c. Si può notare come il picco indesiderato sia stato rimosso.
Al termine di quest'ultima operazione si considera conclusa la selezione dei picchi e si può procedere al salvataggio di essi in un apposita struttura dati in cui sono indicizzati tramite l'istante temporale di riferimento. Un ultimo controllo si effettua per sicurezza anche sulla parte di segnale in cui la selezione dei picchi era buona fin dall'inizio, per verificare che l'implementazione dei vari passaggi non abbia portato ad errori inaspettati.
L'insieme definitivo dei picchi viene ora utilizzato per la definizione di metriche che caratterizzino il segnale. In particolare, l'obiettivo è quello di individuare delle metriche che abbiano un andamento differente tra la fase di veglia e la fase di addormentamento. Preliminarmente però è necessario suddividere il segnale in intervalli di tempo, detti *finestre*, da prendere come unità di riferimento per il calcolo delle metriche.

Capitolo 5

Definizione delle metriche caratterizzanti il segnale

Le metriche che si definiranno successivamente sono degli indici numerici che hanno lo scopo di descrivere quantitativamente il comportamento del segnale, finestra per finestra. L'ampiezza delle finestre che devono essere considerate è un parametro fondamentale che deve essere ottimizzato per catturare nel miglior modo possibile un'evoluzione delle metriche stesse nel corso del tempo. Parallelamente e per lo stesso motivo, è da discutere anche l'opportunità di lavorare con finestre sovrapposte l'una all'altra o meno. In sintesi, è stata fatta un'analisi per ottimizzare la combinazione tra *l'ampiezza* delle finestre e lo *step* da interporre tra l'inizio di due finestre consecutive. Nei paragrafi successivi verranno mostrate alcune immagini che motivano le scelte finali adottate. Come riferimento è stata presa una selezione ristretta di metriche ma le considerazioni che verranno fatte possono essere riprodotte in modo analogo per l'intero set di metriche implementate, che verrà descritto nel dettaglio in seguito. Tutte le immagini fanno riferimento alla porzione di segnale che si conclude con l'istante di addormentamento del soggetto, poichè è questa la fase del segnale oggetto di studio nell'ambito della presente analisi.

5.1 Analisi dell'ampiezza delle finestre

Il parametro in esame in questo paragrafo è stato variato all'interno di un range che comprende valori racchiusi tra 30 secondi e 180 secondi.
Il rischio di ridurre eccessivamente l'ampiezza della finestra è quello di rendere più significativa l'influenza di eventuali irregolarità o disturbi del segnale. D'altra parte, lavorare su finestre troppo grandi riduce sensibilmente la cardinalità del campione di dati a disposizione, limitando di fatto le possibilità di analisi. Inoltre, facendo così si riduce anche l'effetto del cambiamento, dilatando i suoi tempi di rilevazione.
A titolo di esempio si consideri una certa metrica, che chiameremo *Metrica A*, che per un certo paziente abbia un andamento crescente nel passaggio dalla fase di veglia alla fase di addormentamento. L'effetto della variazione dell'ampiezza delle finestre si può valutare attraverso il confronto tra le figure 5.1 e 5.2.

In particolare, dalla figura 5.2 si nota che un ampiezza superiore al minuto delle finestre porta alla perdita della comprensione dell'andamento della metrica in esame.
Dalla figura 5.1, invece, si nota che l'evoluzione della metrica in oggetto è ben rappresentata per ampiezze delle finestre che non superino il minuto.
Si prende ora in esame un'altra metrica, detta *Metrica B*, e si rappresenta analogamente l'effetto

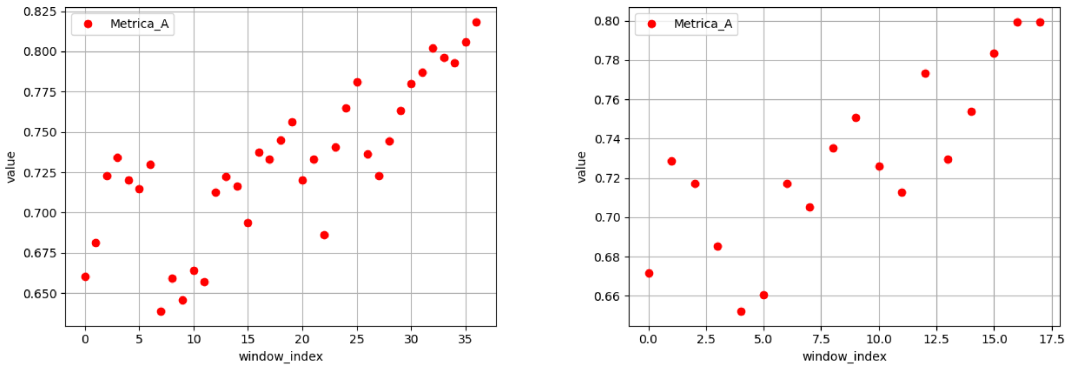


Figura 5.1: Confronto Metrica A finestre di ampiezza 30 e 60 secondi

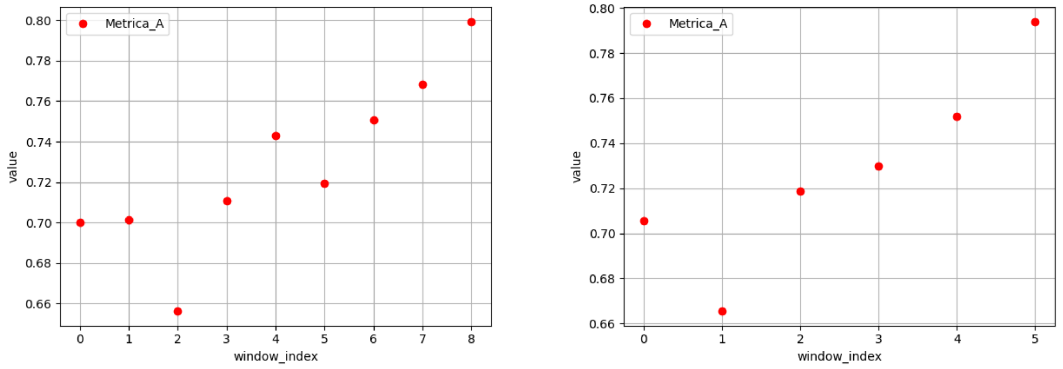


Figura 5.2: Confronto Metrica A finestre di ampiezza 120 e 180 secondi

della variazione dell'ampiezza delle finestre su di essa nelle figure 5.3 e 5.4.

Relativamente a questa seconda metrica viene confermata la tendenza, che si era già notata per

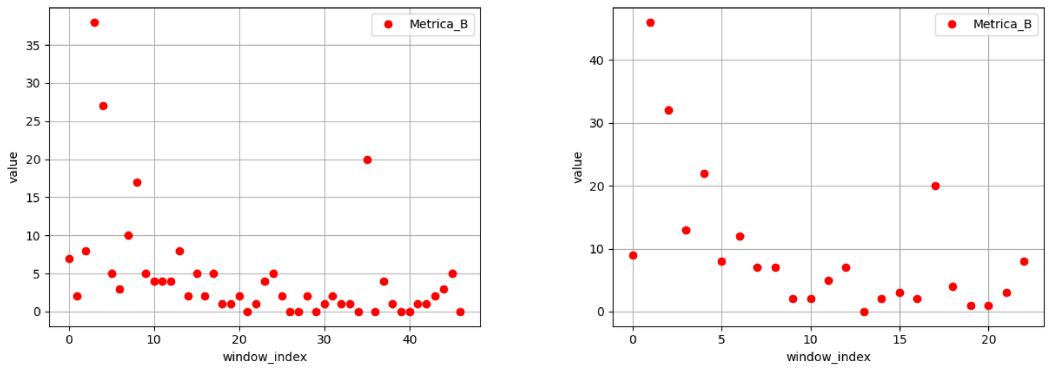


Figura 5.3: Confronto Metrica B finestre di ampiezza 30 e 60 secondi

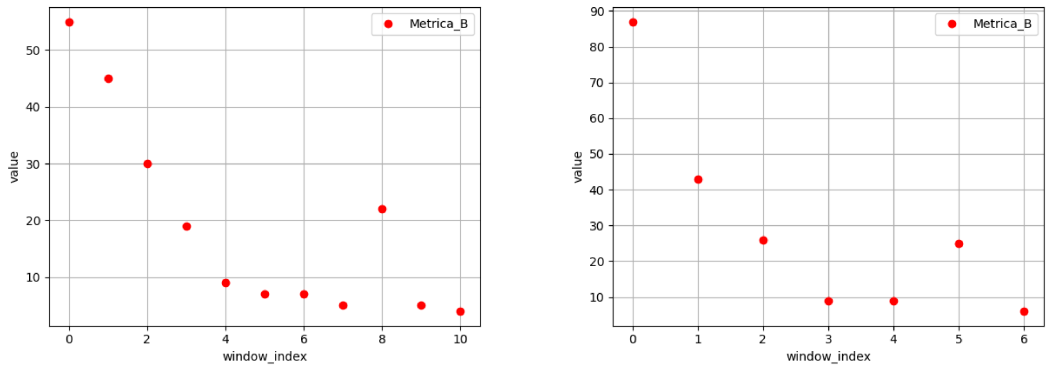


Figura 5.4: Confronto Metrica B finestre di ampiezza 120 e 180 secondi

la metrica A, secondo la quale un'ampiezza superiore al minuto delle finestre non cattura in modo soddisfacente l'andamento del segnale.
Inoltre, dalla figura 5.3, si nota che ad un'ampiezza della finestra di 30 secondi corrisponde un appiattimento della metrica su valori vicini allo zero. Questo dipende dal metodo con cui è stata calcolata la metrica B, che è differente da quello utilizzato per la precedente. In questo caso infatti la base di partenza è costituita da un conteggio di occorrenze per ogni singola finestra.
La conclusione di questa analisi qualitativa è che l'ampiezza ideale per le metriche con cui si è caratterizzato il segnale è di 60 secondi.

5.2 Analisi dello step intermedio tra finestre consecutive

L'altro parametro da valutare nella suddivisione del segnale in finestre è lo step che intercorre tra il punto di partenza di due finestre consecutive. Una soluzione consiste nel far partire una nuova finestra esattamente in corrispondenza dell'istante in cui termina la precedente. Questa soluzione corrisponde alla scelta di uno step pari all'ampiezza della finestra. In questo modo non si verificano mai sovrapposizioni tra una finestra e l'altra.
In alternativa, si può considerare uno step corrispondente ad una frazione dell'ampiezza della finestra. La conseguenza di questo modo di operare è che ogni piccolo sottointervallo del segnale viene considerato all'interno di più finestre diverse. Questo comporta che la cardinalità dell'insieme delle finestre disponibili per l'analisi aumenta. Inoltre, l'ipotesi è che la generazione del dataset effettuata in questo modo produca campioni a densità maggiore e con una migliore capacità di catturare il comportamento del segnale. Per la valutazione di questa ipotesi si è variato il parametro di step da un valore pari all'ampiezza della finestra di riferimento ad un valore pari a $\frac{1}{6}$ dell'ampiezza della finestra di riferimento.
Analogamente a quanto fatto per il parametro corrispondente all'ampiezza delle finestre, si procede tramite il confronto di diversi valori dello step per due metriche di riferimento, A e B, che sono le stesse che si sono prese in esame precedentemente.

In questo caso, sembra evidente dalle figure 5.5, 5.6, 5.7 e 5.8 che uno step più piccolo porta ad un'identificazione migliore dell'evoluzione di entrambe le metriche in esame. Per questo motivo, in conclusione, la combinazione di parametri prescelta per la valutazione delle metriche comprende un'ampiezza delle finestre di 60 secondi e uno step tra una finestra e l'altra di 10 secondi, ovvero di un sesto dell'ampiezza della finestra.

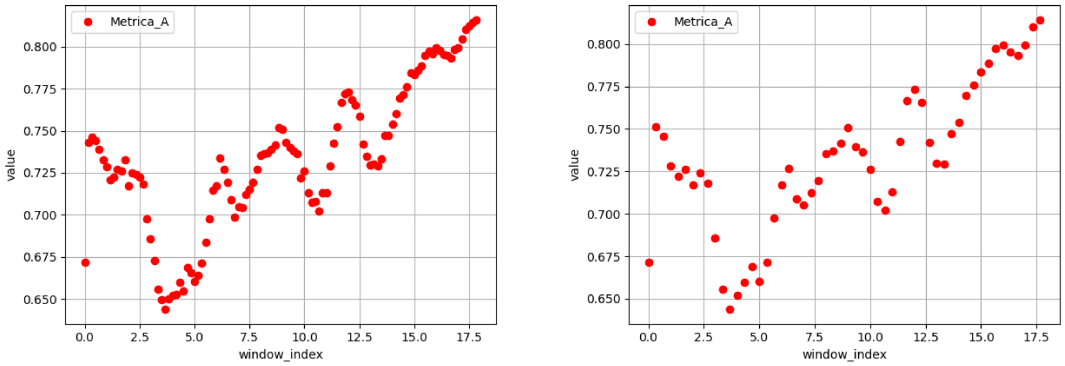


Figura 5.5: Confronto Metrica A step di un sesto e di un terzo dell'ampiezza della finestra

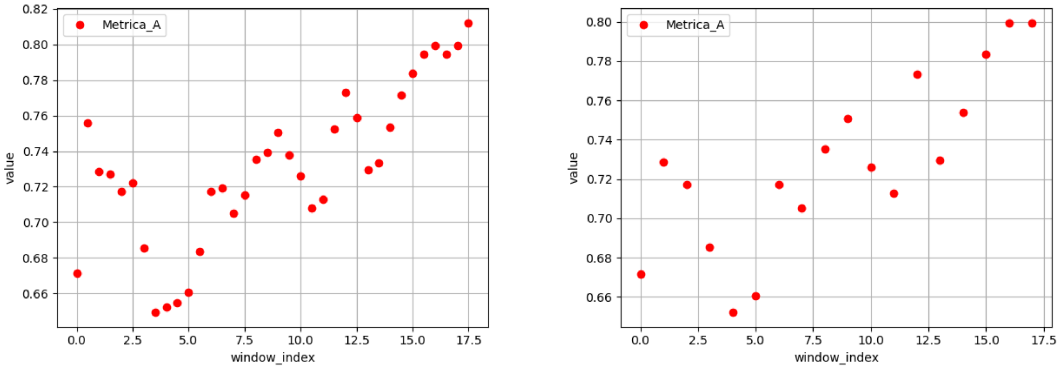


Figura 5.6: Confronto Metrica A step di un mezzo e pari all'ampiezza della finestra

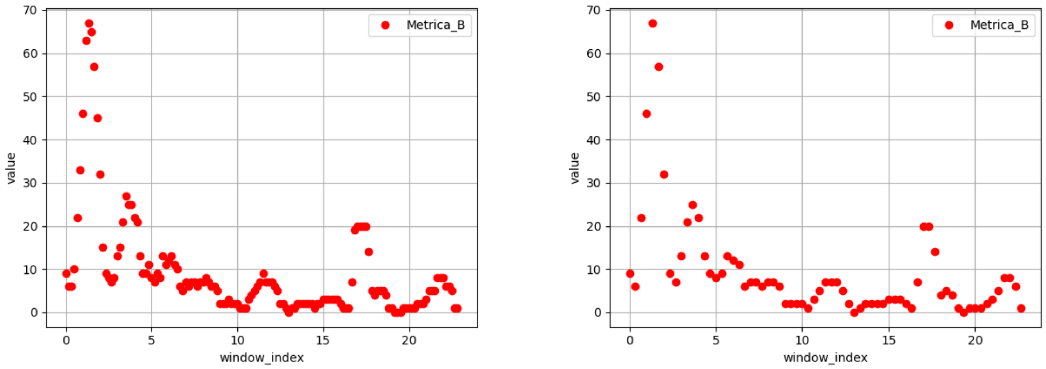


Figura 5.7: Confronto Metrica B step di un sesto e di un terzo dell'ampiezza della finestra

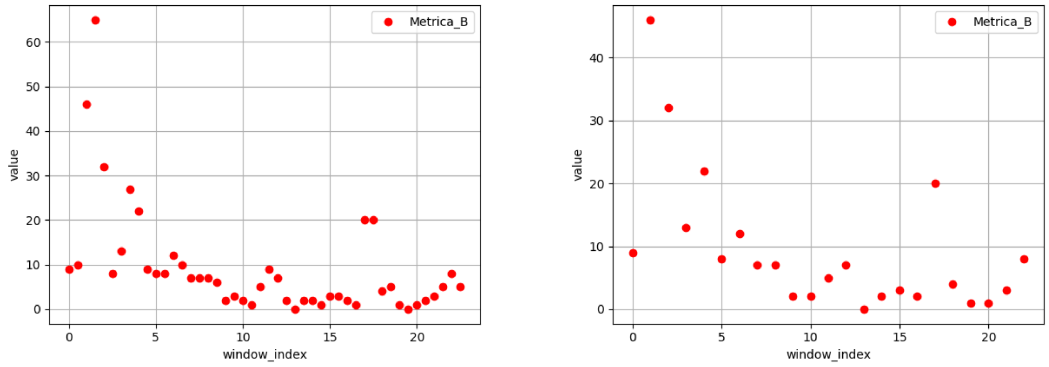


Figura 5.8: Confronto Metrica B step di un mezzo e pari all'ampiezza della finestra

5.3 Definizione delle metriche

L'obiettivo che ci si prefissa tramite la definizione delle metriche è quello di caratterizzare nel più efficace modo possibile l'evoluzione del segnale stesso nel passaggio dalla fase di veglia alla fase di addormentamento. Le metriche che sono state implementate cercano di catturare alcune differenze tra le due fasi che sono state riscontrate analizzando qualitativamente il segnale nel dominio del tempo. In generale, ciò che si è notato è che il segnale tende a diventare più regolare e lento quando il soggetto entra nella fase di addormentamento.
Nella definizione formale di tutte le metriche sono state utilizzate le seguenti notazioni in comune:

- N: numero totale di picchi presenti nella finestra di riferimento
- N_{max}: numero totale di massimi presenti nella finestra di riferimento
- N_{min}: numero totale di minimi presenti nella finestra di riferimento
- Ip: indice temporale in corrispondenza del quale si trova un picco
- Vp: valore del segnale in corrispondenza di un picco

Per tutte le metriche sono stati riportati alcuni grafici che ne raffigurano l'andamento nella fase di veglia e nella fase di addormentamento. Per fase di addormentamento si intendono i 15 minuti antecedenti all'istante di addormentamento del soggetto. Idealmente ogni metrica dovrebbe registrare un aumento o una diminuzione del proprio valore al variare della fase. Di fatto non sempre ciò si verifica, i grafici che sono stati riportati raffigurano sia situazioni in cui l'andamento corrisponde a quello sperato sia situazioni in cui è più difficile riscontrare un chiaro cambio di trend all'iniziare della fase di addormentamento.

Ampiezza Media Questa metrica misura la variazione dell'ampiezza dei picchi. Sono state calcolate tre varianti di essa: prendendo in considerazione solo i massimi, solo i minimi ed effettuando un'ulteriore media dei due precedenti valori. Le loro definizioni sono le seguenti:

$$AmpMediaMax = \frac{\sum_i^{N_{max}} Vp_i}{N_{max}} \tag{5.1}$$

In modo analogo si calcola l'ampiezza media dei minimi, *AmpMediaMin*. I due valori si possono poi ulteriormente mediare per il calcolo dell'ultima variante della metrica:

$$AmpMedia = \frac{AmpMediaMax + AmpMediaMin}{2} \tag{5.2}$$

L'andamento atteso corrisponde ad un aumento dell'ampiezza media dei picchi durante la fase di addormentamento, in pratica tale incremento non è sempre così evidente, come rappresentato in figura 5.9

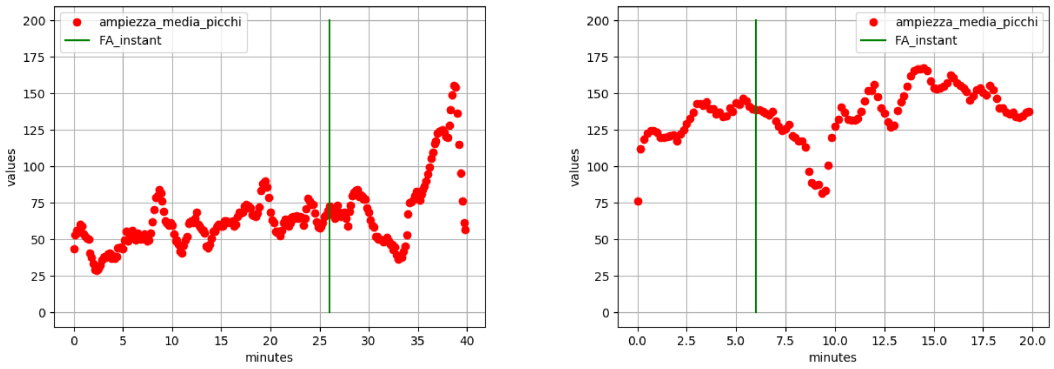


Figura 5.9: Andamento dell'ampiezza media dei picchi per due diversi pazienti

Correlazione tra massimi e minimi Misura *l'indice di correlazione di Pearson* calcolato tra il vettore contenente i massimi, indicato con Max, e il vettore contenente i minimi, indicato con Min:

$$CorrMaxMin = Corr(Max, Min) \tag{5.3}$$

Relativamente a questa metrica è opportuno fare una considerazione: a causa delle irregolarità che si possono presentare nei segnali non è detto che all'interno di una finestra massimi e minimi siano presenti in egual numero. Tuttavia, questa è una condizione necessaria per il calcolo dell'indice di correlazione. Per risolvere questo problema si considerano solo le coppie massimo-minimo regolari, escludendo i picchi che corrispondono a situazioni anomale.
Ci si aspetta che durante la fase di addormentamento il segnale sia più regolare e di conseguenza che si registri una maggiore correlazione per quanto riguarda le coppie massimo-minimo. La figura 5.10 mette a confronto una situazione in cui l'ipotesi è verificata con una in cui non è così.

Distanza euclidea tra minimi e massimi consecutivi Con questa metrica si vuole concentrare l'attenzione sulla stessa coppia minimo-massimo anzichè su picchi che appartengono a coppie adiacenti, come nel caso della 5.3 ma differentemente da tutte le rimanenti.
Infatti, sempre prendendo in considerazione come per la precedente metrica solo le coppie ad andamento regolare, si calcola la distanza euclidea tra il minimo e il massimo e al termine si fa la media di tutti i valori calcolati in una finestra. Formalmente, indicando con Z il numero di coppie ad andamento regolare presenti in una finestra e con h_i e l_i rispettivamente il massimo e il minimo appartenenti alla coppia i, si può scrivere:

$$EucDistMedia = \frac{\sum_{i=1}^{Z} \sqrt{(Ip_{h_i} - Ip_{l_i})^2 + (Vp_{h_i} + |Vp_{l_i}|)^2}}{Z} \tag{5.4}$$

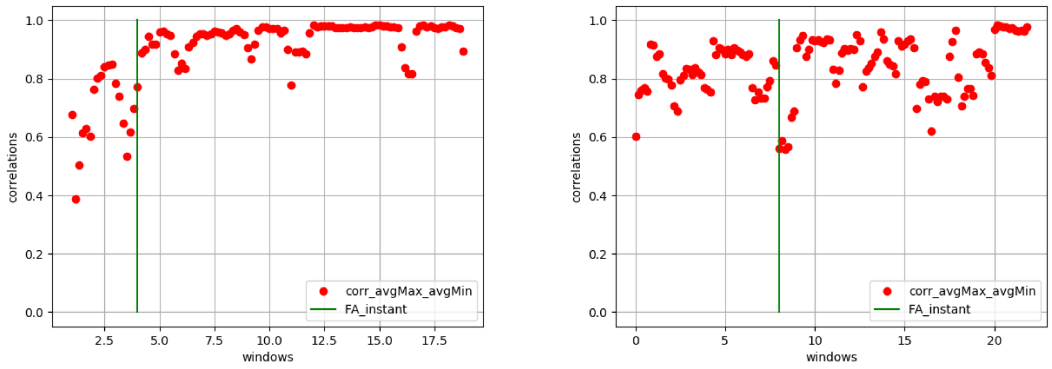


Figura 5.10: Andamento della correlazione tra le coppie massimo-minimo per due diversi pazienti

Anche per quanto riguarda questa metrica, ci si attendono valori più grandi in corrispondenza della fase di addormentamento. La figura 5.11 permette di confrontare due casi reali all'interno del dataset.

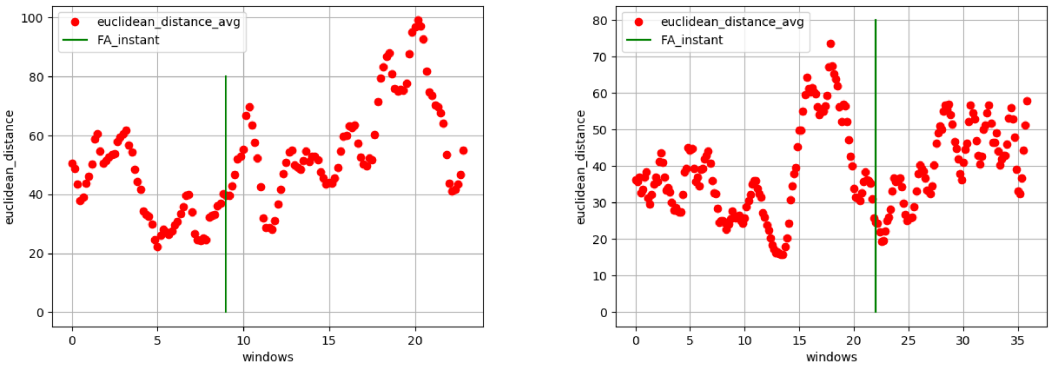


Figura 5.11: Andamento della distanza euclidea media delle coppie massimo-minimo per due diversi pazienti

Outliers Tramite questa metrica si conta il numero di massimi la cui ampiezza è significativamente superiore alla media dei massimi della finestra in esame. Formalmente si definisce *Outlier* un picco o se verifica la seguente condizione, con S che rappresenta una specifica soglia:

$$Vp_o > S\frac{\sum_{i=1}^{N_{max}} Vp_i}{N_{max}} \tag{5.5}$$

Relativamente alla scelta della soglia S sono state effettuate diverse prove, al termine delle quali si è deciso di adottare $S = 0.5$ perchè si tratta del valore che caratterizza meglio la differenza nell'andamento del segnale tra le due fasi. Le immagini seguenti mostrano l'effetto della variazione della soglia sull'andamento della metrica.
Ad esempio, la figura 5.12 mostra l'effetto dell'innalzamento, da sinistra verso destra, della soglia

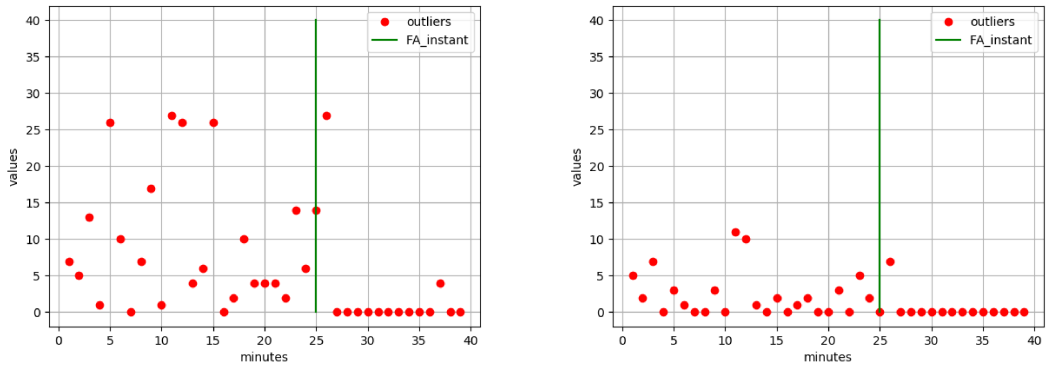


Figura 5.12: Effetto dell'innalzamento del parametro S nella definizione di outlier da 1.5 a 1.75

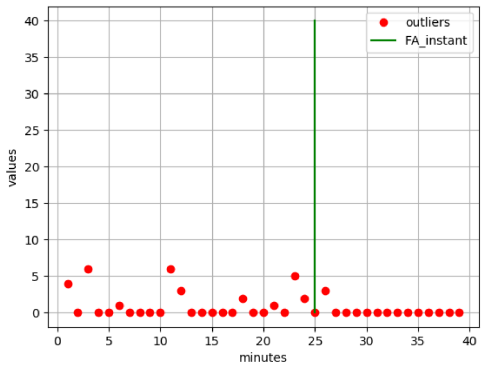


Figura 5.13: Effetto dell'innalzamento del parametro S a 2 nella definizione di outlier

da un valore di 1.5 a 1.75 e per finire a 2. Come si nota, l'innalzamento della soglia e il conseguente minor numero di outlier selezionati portano all'appiattimento della metrica su valori vicini allo zero.
D'altra parte, abbassare la soglia non porta comunque ad un miglioramento dei risultati. A titolo di esempio nella figura 5.14 è rappresentato l'effetto della riduzione di S ad un valore di 1.3: la rilevazione di un maggior numero di outlier si verifica in particolar modo durante la fase di addormentamento. Ciò comporta una maggiore difficoltà a riscontrare differenze tra le due fasi. Per quanto riguarda gli outliers, l'andamento atteso prevede una diminuzione di questi nella fase di addormentamento. La figura 5.15 riporta l'andamento di due pazienti, in uno dei quali è più evidente il verificarsi dell'ipotesi formulata.

Decentralizzazione L'andamento regolare del segnale prevede che esso sia centrato rispetto all'origine. Tuttavia, si è notato che questo aspetto non è sempre verificato e, in particolare nella fase di veglia, si assiste a traslazioni dell'asse verso l'alto o verso il basso. Questa metrica misura

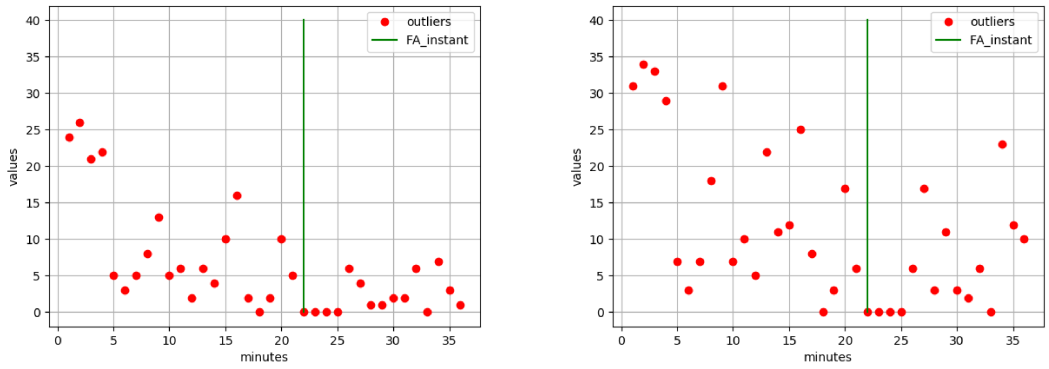


Figura 5.14: Effetto della riduzione del parametro S da 1.5 a 1.3 nella definizione di outlier

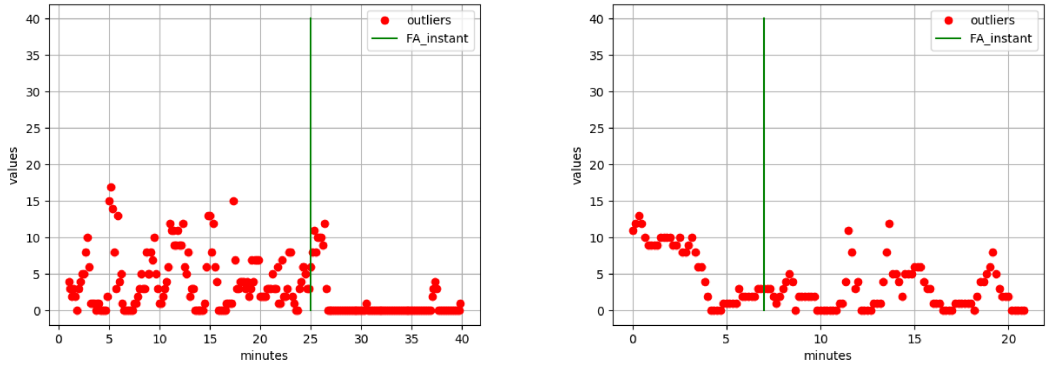


Figura 5.15: Andamento degli outliers per due diversi pazienti

la traslazione media in valore assoluto dell'asse all'interno di una finestra di segnale.

$$Dec = \frac{\left| \frac{\sum_{i=1}^{N_{max}} Vp_i}{N_{max}} + \frac{\sum_{j=1}^{N_{min}} Vp_j}{N_{min}} \right|}{2} \tag{5.6}$$

Il segnale dovrebbe essere maggiormente centrato rispetto all'origine durante la fase di addormentamento. Nella figura 5.16 si riporta il caso di due diversi soggetti.

Distanza tra picchi consecutivi Misura la distanza temporale che intercorre tra due massimi consecutivi. Per ridurre l'impatto di disturbi del segnale si considerano solo le coppie di massimi consecutivi che si trovano in intervalli di tempo in cui è rispettata l'alternanza tra picchi di segno opposto:

$$PD = \frac{\sum_{i=1}^{N_{max}-1} Ip_{i+1} - Ipi}{N_{max} - 1} \tag{5.7}$$

La figura 5.17 mette in luce la differenza tra un caso che rispecchia l'andamento atteso della metrica, ovvero un aumento del suo valore durante la fase di addormentamento, e un caso che si comporta in maniera differente.

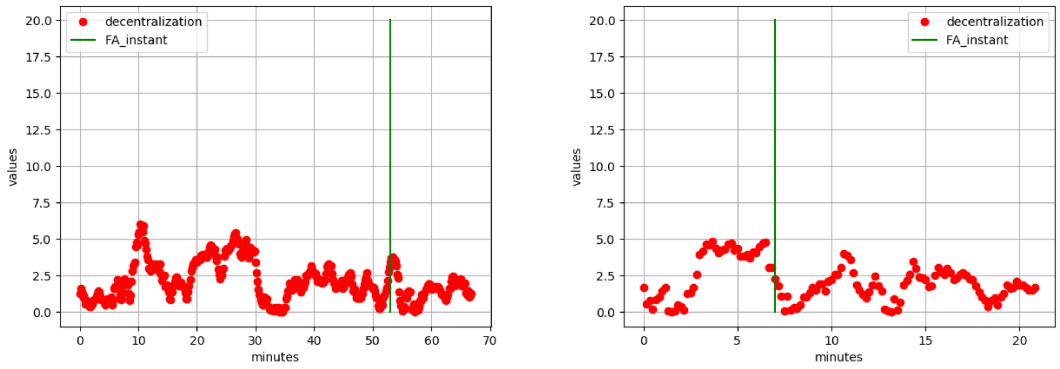


Figura 5.16: Andamento della decentralization per due diversi pazienti

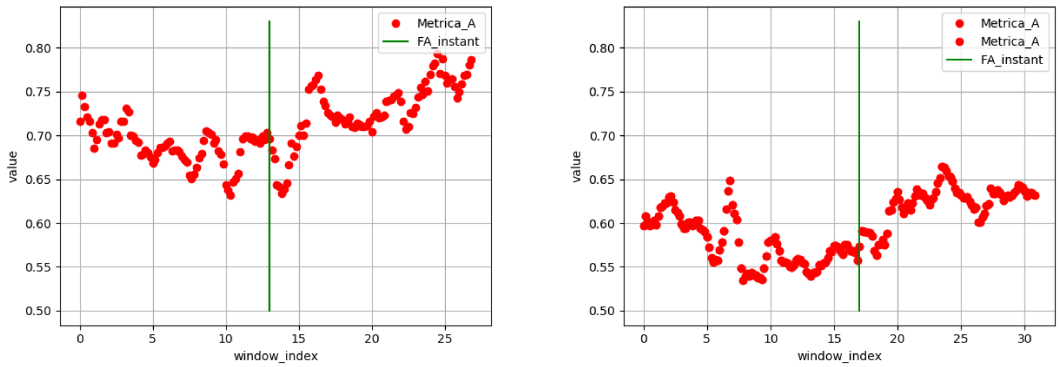


Figura 5.17: Andamento della distanza tra picchi consecutivi per due diversi pazienti

Numero di picchi eliminati Come è stato enunciato nella precedente sezione, in corrispondenza di intervalli di tempo in cui il segnale è irregolare la rilevazione dei picchi è più complessa perchè l'algoritmo tende a selezionare inizialmente un numero maggiore di picchi rispetto a quelli desiderati. Ci si aspetta quindi che nella fase di addormentamento, dove il segnale tende ad essere più regolare, il processo di rilevazione dei picchi sia più semplice e necessiti in misura minore delle correzioni che sono state approntate. Ovvero, ci si aspetta che l'insieme definitivo di picchi selezionato non si discosti molto dalla prima selezione effettuata. Per catturare questa tendenza si conta il numero di picchi eliminati nelle varie fasi del processo di selezione rispetto al primo insieme trovato. Si definisce quindi come $totPeaks_I$ il numero di picchi contenuti nel primo insieme selezionato e come $totPeaks_F$ il numero di picchi contenuti nell'insieme definitivo. La metrica calcolata è la differenza tra i due valori:

$$DelPeaks = totPeaks_I - totPeaks_F \tag{5.8}$$

Come detto, una maggiore regolarità del segnale, e quindi un minor numero di picchi cancellati nel processo di selezione, è ipotizzabile durante la fase di addormentamento. Ciò si verifica effettivamente in uno dei casi mostrati nella figura 5.18, mentre nell'altro caso il trend è decisamente meno evidente.

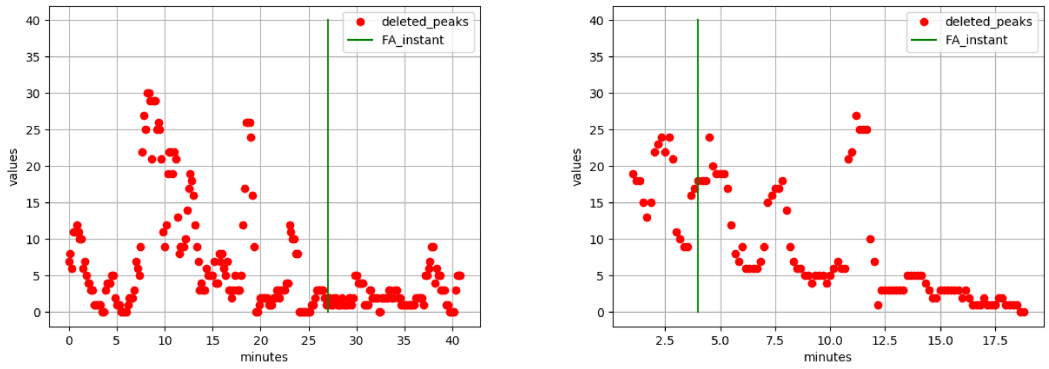


Figura 5.18: Andamento del numero di picchi cancellati per due diversi pazienti

Tutte queste metriche saranno utilizzate per implementare diversi algoritmi di classificazione volti al riconsocimento della fase di addormentamento, come descritto nel capitolo successivo.

Capitolo 6

Problemi di classificazione

6.1 Introduzione alla classificazione

Esistono diverse possibili definizioni per descrivere la *statistica*. Una di queste consiste nel definirla come la scienza che si occupa di imparare dai dati [52]. La statistica è quindi una disciplina che fornisce degli strumenti per effettuare analisi di dati con l'obiettivo di trarne utili conclusioni.
In generale, l'analisi dei dati può avere natura *descrittiva*, ovvero avere come obiettivo la descrizione dei dati raccolti mediante indici numerici. Oppure, l'analisi può avere natura *predittiva*, ovvero essere volta alla costruzione di *modelli* che, sulla base dei dati che descrivono il comportamento di un determinato fenomeno nel passato, ne prevedano l'evoluzione futura.
Un modello predittivo è generalmente composto da due diversi tipi di variabili:

- una variabile *risposta* Y
- un insieme di *predittori* $X_1, X_2, ..., X_n$

Lo scopo del modello è quello di definire una relazione che permetta di stimare le realizzazioni di Y sulla base del valore dei predittori. La variabile risposta può essere *quantitativa* o *qualitativa*. Nel primo caso può assumere valori numerici, continui o discreti, appartenenti all'insieme dei numeri reali o ad un determinato intervallo. Una variabile qualitativa o *categorica*, invece, può assumere solo un ristretto numero di valori appartenenti ad un determinato dominio. Problemi in cui la variabile risposta è qualitativa sono detti problemi di *classificazione* [53].
Infatti, in questo tipo di problemi l'obiettivo è quello di assegnare ogni osservazione ad una determinata *classe*. L'insieme delle classi possibili è rappresentato dal dominio di Y. Si parla di classificazione *binaria* qualora le classi siano due e, solitamente, si codificano con i valori 0 e 1. Il problema in esame nell'ambito del presente lavoro è un problema di classificazione e, precisamente, un problema di classificazone binaria.
Esistono moltissime tecniche utilizzabili per risolvere problemi di classificazione. Nel presente capitolo vengono presentate quelle che sono state impiegate nel problema in esame.
Un approccio comune nella costruzione di un classificatore consiste nel dividere il dataset a disposizione in due parti:

- un *Training set*, utilizzato durante la fase di *apprendimento* del modello
- un *Test set*, utilizzato durante la fase di *validazione* del modello

Il procedimento di costruire il modello prendendo in esame solo una parte dei dati e di testare le sue prestazioni sulla porzione del dataset rimasta fuori dal training set serve a verificare che il

modello abbia capacità predittive oltre che descrittive. Infatti, lo scopo di un classificatore non è classificare correttamente tutte le istanze di cui si conosce già la classe di appartenenza, ma classificare nuove istanze di classe sconosciuta. L'errore commesso dal classificatore sul training set si definisce *training error*, quello sul test set *test error* ed è quest'ultimo che è maggiormente interessante minimizzare. Quando un classificatore è caratterizzato da un training error vicino a 0 e da un test error decisamente più alto si parla di fenomeno dell'*overfitting* [54].
La Figura 6.1 riporta una rappresentazione grafica del fenomeno dell'overfitting nell'ambito della classificazione binaria. Le croci e i cerchi rappresentano osservazioni appartenenti a 2 diverse classi. Sugli assi cartesiani sono rappresentati i due predittori dello specifico problema.

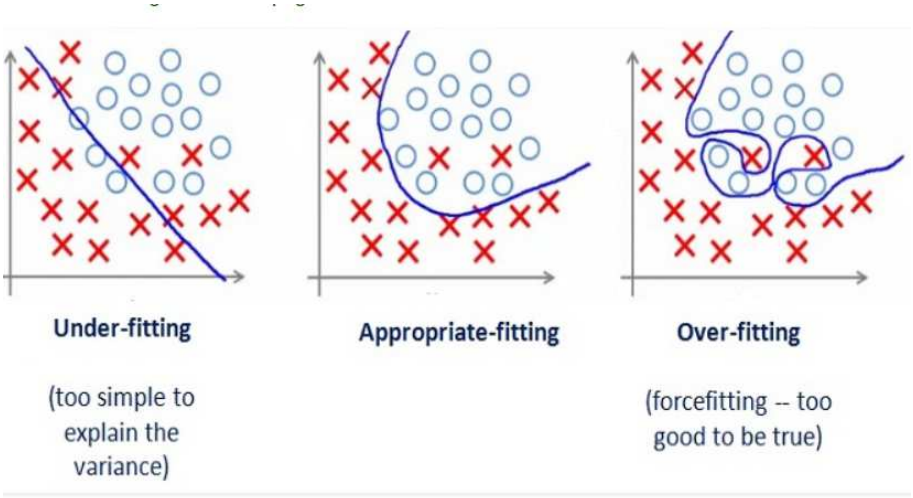


Figura 6.1: Confronto tra un modello inaccurato, un modello accurato e un modello affetto da overfitting [55]

Nel grafico a sinistra, i dati sono stati separati mediante un separatore lineare, ma risulta evidente che tale metodo non è idoneo alla classificazione dell'esempio perchè le classi non sono separabili linearmente. Il grafico centrale rappresenta la situazione maggiormente desiderabile, anche se è vero che si riscontrano alcuni errori di classificazione, ovvero 2 croci erroneamente classificate come cerchi. Infatti il grafico a destra, in cui tutte le osservazioni sono state classificate correttamente, rappresenta un classico esempio di overfitting. Questo perchè il separatore raffigurato in questo grafico è adatto a separare questa particolare configurazione delle osservazioni, ma non ha validità generale proprio perchè basato eccessivamente sui dati di training.
Delineati questi concetti generali, che sono comuni a tutti i classificatori, nelle seguenti sezioni si esporranno le caratteristiche di alcuni algoritmi di specifico interesse.

6.2 Regressione logistica

La regressione logistica è una variante della regressione semplice in cui come variabile risposta si considera la probabilità di accadimento di un evento. In questo ambito è comune l'espressione della probabilità di un evento tramite le sue *odds*, ovvero tramite il rapporto tra la probabilità p di accadimento e la probabilità di non accadimento $1 - p$:

$$ODDS = \frac{p}{1-p} \tag{6.1}$$

Nel caso della classificazione binaria, il corrispondente modello di regressione logistica si definisce con la formula seguente, dove $\beta_0, \beta_1, ..., \beta_n$ sono parametri da fittare durante la fase di apprendimento del modello [56]:

$$logit(y) = ln(ODDS) = ln(\frac{p}{1-p}) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n \tag{6.2}$$

Rielaborando l'equazione precedente, si può esprimere la probabilità condizionata di Y data una certa configurazione dei valori dei predittori:

$$p = P(Y = y | X_1 = x_1, ..., X_n = x_n) = \frac{e^{\beta_0+\beta_1 x_1+...+\beta_n x_n}}{1 + e^{\beta_0+\beta_1 x_1+...+\beta_n x_n}} \tag{6.3}$$

che è equivalente a scrivere:

$$p = \frac{1}{1 + e^{-(\beta_0+\beta_1 x_1+...+\beta_n x_n)}} \tag{6.4}$$

I parametri del modello vengono generalmente stimati con il metodo della *massima verosimiglianza*. Nel caso più semplice, ovvero un modello di regressione logistica con due soli parametri, l'intercetta e un parametro associato all'unico predittore presente, la *funzione di verosimiglianza* risulta essere la seguente [53]:

$$L(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'})) \tag{6.5}$$

Il concetto alla base del metodo della massima verosimiglianza è la massimizzazione del valore della funzione precedente. Per ottenere questo obiettivo, i parametri devono essere ottimizzati in modo tale che, per le istanze contenute nel training set, l'output del modello sia vicino a 1 per quelle della classe 1 e vicino a 0 per quelle della classe 0. Nel caso siano presenti più predittori, la funzione di verosimiglianza si costruisce analogamente.
L'output della regressione logistica è una probabilità che, per definizione, è un numero compreso nell'intervallo [0,1]. Per adattare la regressione logistica alle esigenze della classificazione binaria occorre stabilire una soglia. Ad esempio, si può stabilire che tutti i valori superiori a 0.5 appartengano ad una classe e i valori inferiori alla classe opposta. A seconda del problema si può anche scegliere di adottare una soglia di valore diverso, nel caso sussistano motivi particolari [53]. Ad esempio, in un modello utilizzato da una banca per classificare clienti che non rispettano i requisiti minimi di solidità finanziaria per l'accensione di un mutuo, è legittimo l'utilizzo di una soglia inferiore per mantenere un comportamento maggiormente conservativo.
La regressione logistica funziona bene nel caso in cui le classi siano separabili linearmente ed ha il vantaggio di mostrare in maniera immediata quali predittori sono più importanti di altri. Nel caso in cui lo spazio dei dati abbia una conformazione più complessa, è preferibile utilizzare un classificatore basato su un approccio radicalmente diverso, come il *KNN* [53].

6.3 KNN

Il classificatore KNN (K-Nearest Neighbors), diversamente dalla regressione logistica, è un algoritmo che affronta il problema della classificazione da un punto di vista *non parametrico*. Infatti, non sono presenti in questo modello dei parametri da stimare e le decisioni vengono prese basandosi su un approccio differente.
L'algoritmo del KNN prevede che per una nuova istanza x_0 si prenda in considerazione il vicinato N_0 che comprende le K istanze più vicine nello spazio dei dati, ovvero più simili nell'intero dataset.

A questo punto, si calcola la probabilità che l'istanza appartenga a ciascuna classe utilizzando l'equazione seguente [53]:

$$P(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j) \tag{6.6}$$

dove $I(x)$ è la funzione indicatrice, che assume valore 1 se la condizione in input è verificata e 0 altrimenti. La probabilità che x_0 appartenga alla classe j, quindi, è pari alla frazione di vicini appartenenti a N_0 di classe j. Dunque, la nuova istanza viene classificata come appartenente alla classe cui appartiene il maggior numero di suoi vicini.
In Figura 6.2 è possibile vedere un esempio di classificazione tramite KNN.

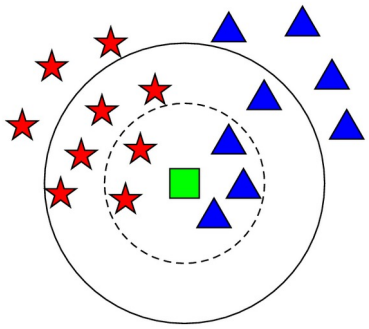


Figura 6.2: Esempio di classificazione tramite KNN al variare di K [58]

Le stelle e i triangoli rappresentano istanze di due differenti classi e il quadrato una nuova istanza da classificare. I cerchi concentrici delimitano il vicinato preso in considerazione. Nel caso del cerchio tratteggiato, è stato considerato il vicinato corrispondente a $K = 5$, che porta l'algoritmo a classificare la nuova istanza come appartenente alla classe dei triangoli. Nel caso del cerchio disegnato con la linea continua, invece, il vicinato è stato costruito utilizzando un valore di K corrispondente a 10. É interessante notare come in questo secondo caso la nuova istanza, differentemente dal caso precedente, venga classificata come appartenente alla classe delle stelle. Ciò è un esempio di come la scelta di K influenza pesantemente la classificazione.
In generale, all'aumentare di K diminuisce la flessibilità del modello, che produce separatori sempre più affini ad un separatore lineare. La scelta del corretto valore di K è fondamentale per una corretta classificazione. Infatti, all'aumentare della flessibilità del modello diminuisce senz'altro l'errore sul training set ma, come descritto precedentemente, aumenta il rischio di incorrere in overfitting [53].
Il vantaggio più importante del classificatore KNN è proprio la sua flessibilità, che permette di operare una buona classificazione anche in caso di classi non separabili linearmente. La lacuna maggiore di questo modello è a livello interpretativo: risulta difficile comprendere quali siano i predittori più influenti nel processo di classificazione.

6.4 Decision Tree

Un ulteriore approccio al problema della classificazione binaria è quello fornito dai cosiddetti *Alberi decisionali* o, in inglese, *Decision Trees*. Tramite questo metodo si cerca di partizionare lo spazio dei predittori in un determinato numero J di *regioni di decisione*. A tutte le osservazioni presenti in ciascuna regione di decisione si assegna la classe maggiormente presente. Il nome del modello è dovuto alla particolare forma che esso assume e che è rappresentata in Figura 6.3.

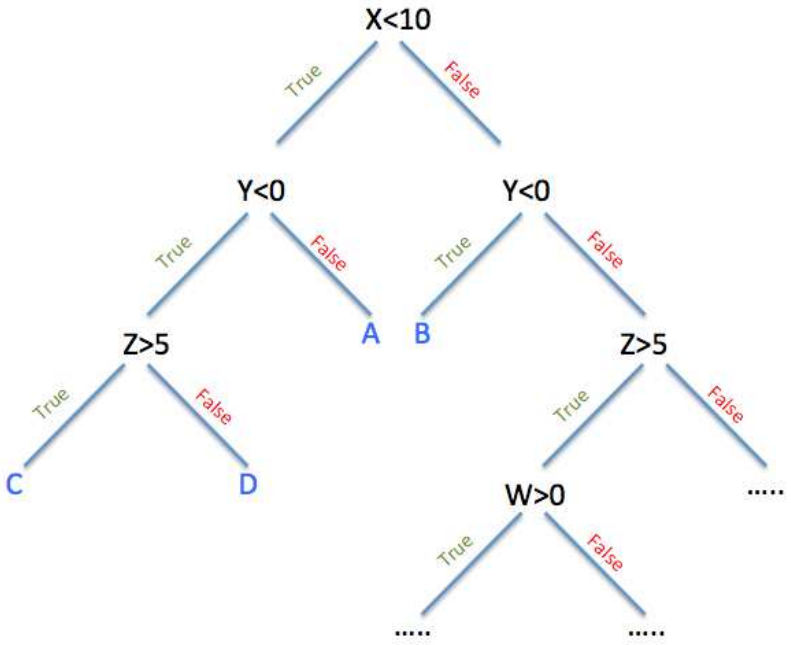


Figura 6.3: Esempio di albero decisionale [59]

Nella figura sopracitata X, Y, W, Z sono predittori, mentre A, B, C, D indicano regioni di decisione a cui è associata una determinata classe. Ad ogni ramificazione dell'albero corrisponde un partizionamento dello spazio dei predittori lungo un determinato predittore e per un determinato valore di taglio.
Le decisioni che riguardano i partizionamenti da effettuare e di conseguenza la costruzione delle regioni di decisione avvengono mediante un approccio euristico denominato *recursive binary splitting* [53], in cui ad ogni iterazione si divide una delle regioni derivate dall'iterazione precedente minimizzando un determinato criterio.
Un criterio semplice da utilizzare è il *classification error rate* [53], definito come segue:

$$E = 1 - \max_k(p_{mk}) \tag{6.7}$$

dove p_{mk} rappresenta la percentuale di osservazioni del training set della regione m che appartengono alla classe k. Come si può notare, E assume valori vicini a 0 quando la quasi totalità delle istanze della regione m appartiene alla classe k.
Esistono tuttavia criteri più efficienti rispetto al classification error rate [53], uno di questi è il *Gini index* che si definisce nel modo seguente:

$$G = \sum_{k=1}^{K} p_{mk}(1 - p_{mk}) \tag{6.8}$$

L'indice di Gini è una misura della purezza di una regione di decisione, infatti un suo valore vicino a 0 indica una netta predominanza di osservazioni di una singola classe.
In alternativa, si può anche utilizzare il criterio della minimizzazione della *cross entropy*, che viene

definita così:

$$D = -\sum_{k=1}^{K} p_{mk} log(p_{mk}) \tag{6.9}$$

I valori dell'indice di Gini e della cross entropy sono generalmente simili. Infatti, anche quest'ultimo indice assume valori vicini a 0 quando una delle p_{mk} è vicina a 1, mentre tutte le altre sono vicine a 0. L'utilizzo di uno di questi ultimi due criteri è di solito considerato equivalente.
I principali vantaggi dell'utilizzo di alberi decisionali sono i seguenti [57]:

- permettono di classificare in spazi dei predittori complessi, mediante l'unione di regioni locali semplici
- permettono un'immediata interpretazione del modello
- permettono di selezionare differenti sottoinsiemi ottimali dei predittori a diversi livelli dell'albero

D'altro canto, le prestazioni degli alberi decisionali in termini di accuratezza non sempre sono competitive nei confronti di altri tipi di classificatori.

6.5 Random Forest

La *Random Forest* è un modello che ha come obiettivo il miglioramento dell'accuratezza degli alberi decisionali. La tecnica utilizzata prevede la costruzione di B diversi alberi e successivamente la combinazione dei loro risultati.
Il primo passaggio consiste nell'applicazione del metodo del *Bootstrap*, un metodo che a partire da un dataset di cardinalità N, ne genera B della stessa cardinalità con la tecnica del campionamento casuale *con ripetizione*. Ciò significa che la stessa osservazione può essere estratta più di una volta [61].
Successivamente vengono costruiti B differenti alberi, uno per ogni dataset generato tramite il bootstrap, con le stesse modalità descritte nel paragrafo precedente. Ogni albero, però, prende in considerazione solo una parte dei predittori. In particolare, per ogni albero che viene costruito viene estratto un sottoinsieme casuale di n predittori e il modello viene fittato utilizzando solo questo sottoinsieme. In Figura 6.4 è rappresentato graficamente il procedimento in esame.

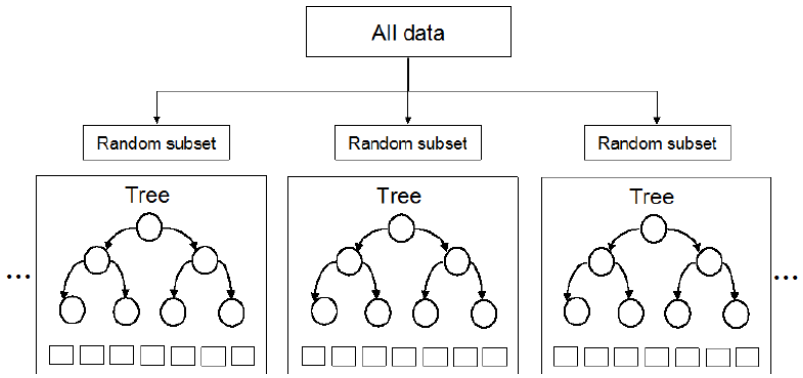


Figura 6.4: Rappresentazione grafica della Random Forest [62]

Il motivo per cui si sceglie di restringere in modo casuale di volta in volta il campo dei predittori è forzare ogni albero ad essere sensibilmente diverso dall'altro. Infatti, senza il campionamento di

un sottoinsieme di predittori, tutti gli alberi sarebbero approssimativamente uguali.
La combinazione dei risultati di alberi che sono molto diversi tra loro ha l'effetto di ridurre la varianza del risultato finale e di renderlo quindi maggiormente affidabile [53].
É importante anche discutere il fitting dei parametri B e n. Per quanto riguarda il numero di alberi B, il suo incremento non porta ad overfitting [53]. Per questo motivo, lo si può impostare grande a piacimento in modo tale che massimizzi la precisione del modello, senza però sottovalutare il fatto che un valore di B molto grande può aumentare notevolmente la complessità del modello. Non tanto dal punto di vista computazionale, in quanto il tempo computazionale della Random Forest è pari a [60]:

$$t = cB\sqrt{p}N \log N \tag{6.10}$$

dove p è il numero totale di predittori, N la cardinalità del dataset e c una costante. Come si può notare, il tempo computazionale è lineare in funzione del numero di alberi B. Tuttavia, si tratta di un modello che richiede una grande quantità di memoria, dovendo memorizzare una matrice di dimensioni NxT [60].
Il numero di predittori da selezionare, invece, richiede una maggiore attenzione. In generale, se p è il numero totale di predittori, una scelta tipica è quella di porre $n \approx \sqrt{p}$. Più si aumenta il valore di n, più ci si avvicina a p, perdendo il vantaggio della random forest, ovvero quello di forzare l'utilizzo dei predittori che meno influenzano la classificazione.
L'utilizzo della Random forest è generalmente utile a migliorare le performances del singolo albero decisionale, ma a scapito della facilità di interpretazione.

6.6 Validazione di un classificatore tramite Cross-Validation

Nella sezione precedente si è già detto dell'opportunità di valutare le performances di un classificatore su un dataset che non è stato coinvolto nella costruzione del modello. A questo proposito, nella presente sezione si approfondisce l'argomento discutendo il problema della selezione del test set rispetto al dataset completo e l'implementazione della tecnica della *Cross-Validation*.

Validation Set Approach Il metodo più semplice ed intuitivo per partizionare un dataset in training set e test set consiste nel campionamento casuale di metà delle osservazioni, le quali andranno così a costituire il test set. Questo metodo prende il nome di *Validation Set Approach* [53]. Tuttavia, questo metodo presenta due principali criticità:

- una marcata riduzione del training set può portare alla costruzione di un modello meno accurato ed alla sovrastima del test error
- è fortemente dipendente da quali osservazioni vengono estratte nel test set

In merito al secondo punto del precedente elenco si osservi la Figura 6.5. In essa sulle ascisse è rappresentato il grado di un modello di regressione polinomiale, sulle ordinate, con funzione di misura dell'errore, l'MSE (Mean Squared Error):

$$MSE = \frac{\sum_{i=1}^{n}(x_i - \hat{x}_i)^2}{n} \tag{6.11}$$

dove x_i e $\hat{x}_i$ sono il valore reale e stimato dell'i-sima osservazione del test set e n la cardinalità del test set. Ogni spezzata di colore diverso rappresenta il risultato che si ottiene fittando il modello con un diverso campionamento del test set. Si nota che l'MSE è influenzato sensibilmente dalla configurazione considerata. C'è quindi la necessità di implementare tecniche più avanzate che permettano di risolvere i problemi esposti precedentemente.

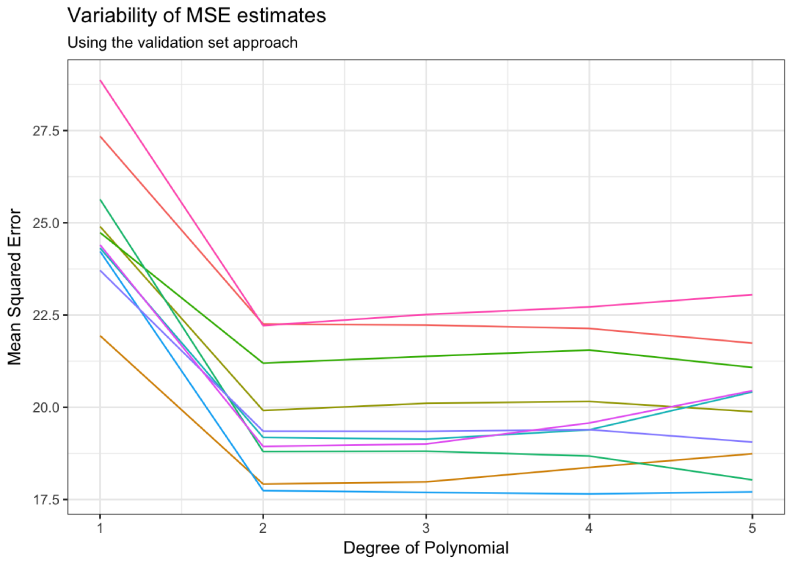


Figura 6.5: Variabilità del MSE in base alla composizione del test set [53]

Leave-One-Out Cross-Validation La *Leave-One-Out Cross-Validation* (*LOOCV*) è una tecnica che nasce con lo scopo di migliorare le performances del Validation Set Approach. Questo metodo, data n la cardinalità del dataset a disposizione, si compone di n iterazioni, in ognuna delle quali si compiono i seguenti passaggi:

- Si genera un test set composto solo dall'i-sima istanza
- Si costruisce il modello sul training set
- Si calcola l'errore applicando il modello sull'unica istanza del test set

In questo modo, ad ogni passaggio si ottiene una misura del test error. Di tutte queste misure alla fine del procedimento si calcola la media. Ad esempio, poichè questo capitolo si concentra su problemi di classificazione, come indice di errore si può considerare il numero di istanze classificate scorrettamente Err_i:

$$Err_i = I(x_i \neq \hat{x_i}) \tag{6.12}$$

dove $I(x)$ rappresenta la funzione indicatrice. La stima finale del test error corrisponde quindi a:

$$CV_{tot} = \frac{1}{n}\sum_{i=1}^{n} Err_i \tag{6.13}$$

L'utilizzo di questa tecnica, rispetto al Validation Set Approach presenta alcuni vantaggi. Prima di tutto, il modello viene fittato su un trainig set dalla cardinalità quasi identica a quella del dataset originale. Questo porta ad un modello più accurato e ad una stima del test error che non è per eccesso. Inoltre, il fatto di ripetere n volte il calcolo del test error diminuisce la variabilità della misura di CV_{tot}. Contemporaneamente, la *LOOCV* è molto dispendiosa da un punto di vista computazionale [64]. Infatti, l'operazione di fitting del modello deve essere ripetuta n volte, un numero pari alla cardinalità del dataset, che in alcune applicazioni reali potrebbe essere un numero molto grande, tale da rendere proibitiva l'operazione.

K-Fold Cross Validation La *K-Fold Cross Validation* è una procedura che generalizza la *LOOCV*, con l'obiettivo di costituire un buon compromesso in termini di fattibilità computazionale e di accuratezza nella stima del test error.
Il funzionamento della K-Fold Cross Validation è analogo a quello della *LOOCV*, con la differenza che il numero di iterazioni dell'algoritmo non è pari alla cardinalità del dataset originale ma è decisamente minore e pari ad un parametro k. In pratica, anzichè cosruire di volta in volta un test set contenente un'unica occorrenza, si partiziona il dataset in k sottoinsiemi di cardinalità costante. In ognuna delle iterazioni che si eseguono successivamente, uno dei sottoinsiemi così generati costituisce il test set, mentre l'unione dei restanti $k-1$ costituisce il training set. Al termine del procedimento si calcola la media dei test error. Ad esempio, considerando ancora Err_i come indice di errore [65]:

$$CV_k = \frac{1}{k}\sum_{i=1}^{k} Err_i \tag{6.14}$$

É evidente che la *LOOCV* costituisce un caso particolare della K-Fold Cross-Validation in cui $k = n$, con n cardinalità del dataset in esame. La Figura 6.6 rappresenta schematicamente le iterazioni compiute nell'ambito di una K-Fold Cross-Validation con $k = 5$. Questa tecnica riduce

Figura 6.6: Rappresentazione grafica delle iterazioni di una K-Fold Cross-Validation con $k = 5$ [66]

la complessità computazionale della *LOOCV*, fornendo così una soluzione alla sua più importante criticità. Contemporaneamente, il fatto che si basi su diverse iterazioni contribuisce a ridurre la variabilità della stima del test error, problema che caratterizza il Validation Set Approach. Dall'altro lato della bilancia, bisogna considerare quanto la riduzione delle iterazioni e di k impatta negativamente sulla precisione della stima del test error, che per la *LOOCV* è elevata. É stato dimostrato [53] empiricamente che l'utilizzo di un valore di k compreso tra 5 e 10 costituisce un buon compromesso dal punto di vista della precisione della stima del test error. Allo stesso tempo, un valore nel range sopracitato ha, rispetto alla *LOOCV*, il vantaggio di portare alla costruzione di training set meno correlati da un'iterazione all'altra. Questo aspetto è vantaggioso nella minimizzazione della varianza della stima del test error.

Capitolo 7

Metodi per la valutazione di un classificatore

Quando si ha a disposizione un algoritmo implementato e validato come descritto nel Capitolo 6, la fase successiva corrisponde alla valutazione del suo output. In questa fase vengono utilizzati appositi indici che esprimono la bontà della classificazione eseguita.

7.1 La matrice di confusione

Il primo output di un algoritmo di classificazione da prendere in considerazione è la cosiddetta *matrice di confusione* che, nell'ambito della classificazione binaria, si presenta come riportato nella Tabella 7.1 [67]. Il problema di classificazione è spesso ricondotto ad una classificazione binaria.

	True +	**True -**
Pred +	TP	FP
Pred -	FN	TN

Tabella 7.1: Matrice di confusione

Pertanto nella diagnostica medica è comune indicare le classi come classe dei positivi e classe dei negativi, riferendosi implicitamente alla presenza o meno, ad esempio, di una determinata patologia.
Nella matrice di confusione, le colonne rappresentano la classe reale delle istanze, mentre le righe la predizione effettuata dal classificatore. Nel blocco delle quattro caselle in Tabella 7.1 troviamo:

- TP, *True Positive*: il numero delle istanze della classe + classificate in modo corretto
- FP, *False Positive*: il numero di istanze della classe − classificate in modo errato
- TN, *True Negative*: il numero di istanze della classe − classificate in modo corretto
- FN, *False Negative*: il numero di istanze della classe + classificate in modo errato

I primi indici di bontà della classificazione prodotta possono essere estrapolati effettuando alcuni rapporti tra le quantità elencate. In generale per ogni classe possono essere calcolati la *precisione* e il *richiamo*. La prima si formula, a titolo di esempio per la classe +, nel modo seguente:

$$Prec_+ = \frac{TP}{TP + FP} \tag{7.1}$$

Mentre il richiamo, sempre a titolo di esempio per la classe +, ha la seguente espressione:

$$Rec_+ = \frac{TP}{TP + FN} \tag{7.2}$$

Queste due misure danno informazioni diverse e complementari. La precisione indica l'accuratezza delle predizioni del classificatore, ovvero misura una sorta di grado di affidabilità delle predizioni ottenute. Il richiamo dà invece una misura di quante istanze di ciascuna classse vengano riconosciute dall'algoritmo. L'ideale, ovviamente, è massimizzare entrambe le metriche finora descritte.
Un metodo alternativo di riferirsi al richiamo, utilizzato prevalentemente in ambito medico ma non solo, è quello di parlare di *sensitività*, *Sens* e *specificità*, *Spec*. In particolare, la sensitività corrisponde a Rec_+ e misura la capacità del test di riconoscere le istanze che presentano la caratteristica cercata. La specificità, al contrario, corrisponde a Rec_- e indica la capacità del test di riconoscere le istanze che non presentano la caratteristica di interesse. Da un punto di vista formale:

$$Sens = \frac{TP}{TP + FN} \tag{7.3}$$

$$Spec = \frac{TN}{TN + FP} \tag{7.4}$$

7.2 L'accuratezza

Uno dei criteri che vengono tradizionalmente considerati per la valutazione della bontà di un classificatore è quello dell'*accuratezza* (ACC) [69]. Di seguito se ne richiama la formula, in cui TP e TN sono veri positivi e veri negativi mentre FP e FN falsi positivi e falsi negativi:

$$ACC = \frac{TP + TN}{TP + FN + FP + TN} \tag{7.5}$$

La massimizzazione dell'accuratezza non è tuttavia un criterio sempre idoneo alla valutazione di un algoritmo di classificazione, soprattutto in presenza di dataset sbilanciati, ovvero in cui una delle due classi è numericamente preponderante. A dimostrazione di ciò, si consideri un dataset contenente 800 occorrenze negative e 200 occorrenze positive, le quali sono quelle interessanti da rilevare ai fini del problema specifico. Un algoritmo che produca come risultato la matrice di confusione riportata in Tabella 7.2, sebbene riporti un valore di accuratezza soddisfacente e pari a $(790 + 60)/(790 + 10 + 140 + 60) = 0.85$, non può essere considerato un buon classificatore.

	True +	**True -**
Pred +	60	10
Pred -	140	790

Tabella 7.2: Esempio numerico di cattiva classificazione di un dataset sbilanciato

Questo perchè la sensitività del test, ovvero la percentuale di positivi correttamente identificata sul totale, corrisponde ad un valore poco soddisfacente ed in particolare a $60/(60 + 140) = 0.3$.
In sintesi il classificatore, essendo influenzato dalla netta predominanza di istanze negative, contrassegna più facilmente come negativa una nuova istanza, con il risultato di ottenere una buona accuratezza ma a scapito del raggiungimento dell'obiettivo principale, ovvero la corretta identificazione dei positivi.
Sorge quindi la necessità di introdurre nuovi strumenti per valutare ed indirizzare la scelta del miglior classificatore da utilizzare, uno di questi è la curva *ROC* (Receiver Operating Characteristic) [78].

7.3 La curva ROC

La curva *ROC* è una curva parametrica. Per definirla si consideri che ogni test effettuato da un classificatore si basa su una soglia c che ha la funzione di discriminare le istanze che sono positive al test da quelle che risultano negative. Generalmente, le distribuzioni delle popolazioni delle istanze positive e delle istanze negative si sovrappongono tra loro. Un esempio, relativo ad un test diagnostico, è rappresentato in Figura 7.1.

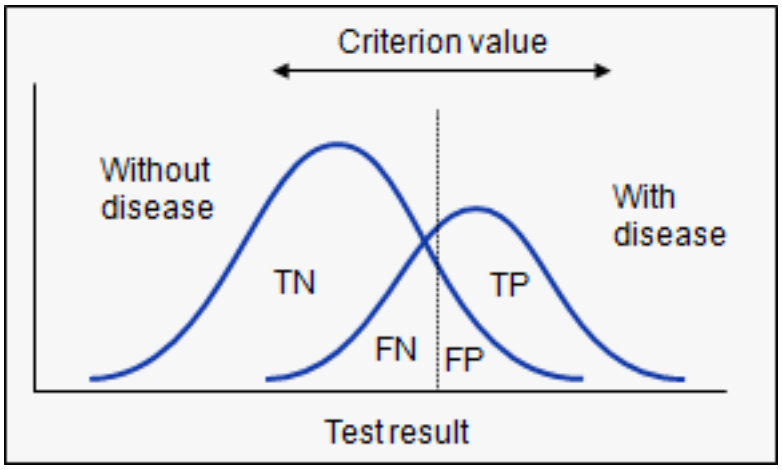


Figura 7.1: Esempio di popolazioni con distribuzioni che si intersecano [71]

Come si può notare, al variare della soglia c variano anche i falsi e i veri positivi (FP, TP) e i falsi e i veri negativi (FN, TN). Si può quindi affermare che specificità e sensitività dipendano anch'esse dal valore di c. A questo punto si può quindi definire la curva *ROC* nel modo seguente [79]:

$$curvaROC = \{(1 - Spec(c), Sens(c)); -\infty < c < +\infty\} \tag{7.6}$$

Graficamente, un esempio di curva *ROC* è rappresentato nella Figura 7.2. Sull'asse delle ascisse è rappresentato il FPR (False Positive Rate) e sull'asse delle ordinate il TPR (True Positive Rate), che corrispondono alle seguenti definizioni:

$$FPR = 1 - Spec(c) = \frac{FP}{FP + TN} \tag{7.7}$$

$$TPR = Sens(c) = \frac{TP}{TP + FN} \tag{7.8}$$

Questi due valori rappresentano, rispettivamente, la percentuale di istanze negative classificate erroneamente come positive e la percentuale di istanze positive correttamente rilevate. La bisettrice del grafico rappresenta una classificazione completamente casuale. La situazione ideale è rappresentata dall'angolo in alto a sinistra, punto corrispondente al 100% di diagnosi corrette e allo 0% di errori di classificazione sulla classe positiva. Ad una curva *ROC* passante per tale punto corrisponde una $AUROC$ (Area Under *ROC*) pari a 1.
Si tratta in ogni caso di una situazione teorica, difficilmente raggiungibile nella pratica. La $AUROC$ è comunque una buona misura di valutazione della bontà di un classificatore, senza dubbio migliore rispetto all'accuratezza. Un classificatore si considera tanto più preciso quanto più la sua $AUROC$ si avvicina a 1.

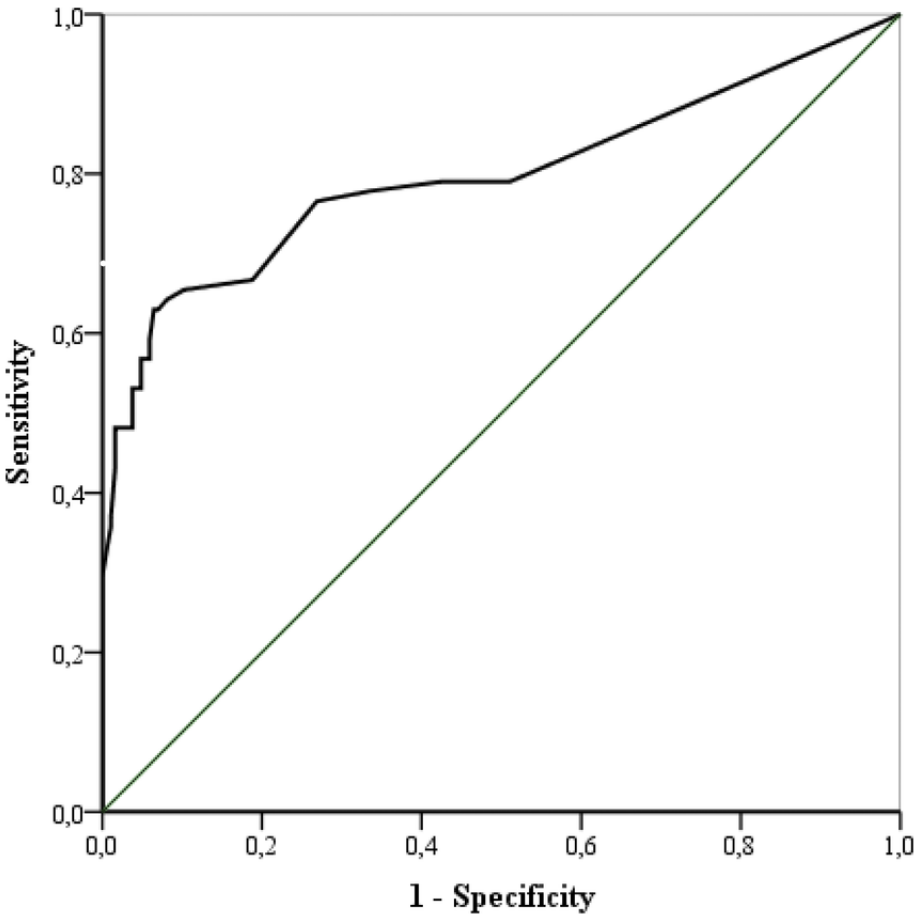


Figura 7.2: Esempio di curva *ROC* [72]

Una considerazione interessante che si può fare è che un modo per muoversi lungo la curva *ROC* è agire sulla classe maggioritaria variandone la cardinalità [78]. In particolare, la riduzione della cardinalità della classe maggioritaria tramite un undersampling produce uno spostamento lungo la curva *ROC* in direzione dell'angolo in alto a destra. Questo se si assume che la classe maggioritaria, come avviene di solito, corrisponda ai negativi. Infatti, un dataset fortemente sbilanciato a favore della classe positiva produrrebbe un TPR molto elevato ma a fronte di un FPR altrettanto elevato, perchè quasi tutte le istanze sarebbero classificate come positive. Viceversa, un dataset fortemente sbilanciato a favore della classe negativa produrrebbe TPR e FPR molto bassi. Ciò può essere un elemento a conferma del fatto che un dataset bilanciato conduce a decisioni di classificazione migliori. In relazione a quest'ultimo aspetto è presente una trattazione più approfondita nel Capitolo 8.

Capitolo 8

Classificazione di classi sbilanciate

8.1 Introduzione al problema dello sbilanciamento

Nell'ambito dei problemi di classificazione, è preferibile lavorare con dataset *bilanciati*, ovvero con dataset in cui la percentuale di osservazioni associate a ciascuna classe è uniforme. Circoscrivendo il campo alla classificazione binaria, la situazione ideale corrisponde a lavorare con un dataset in cui le due classi sono presenti in egual numero. Se il dataset è *sbilanciato* si può registrare un calo delle performances da parte di molti algoritmi di classificazione [68]. Di conseguenza, sono stati elaborati alcuni approcci che consentono di gestire dataset sbilanciati limitando i problemi e massimizzando le performances dei classificatori.
Il motivo per cui in generale è così importante lo studio della gestione di dataset sbilanciati è che sono molto comuni nella pratica. In situazioni reali, infatti, accade spesso che i dati a disposizione costituiscano un dataset sbilanciato a causa di caratteristiche intrinseche del problema in esame. Spesso, inoltre, la classe minoritaria è quella più importante dal punto di vista della classificazione ed è l'obiettivo dell'analisi.
Si possono portare alcuni esempi a sostegno di questa argomentazione:

- Rilevazione di transazioni fraudolente: le transazioni regolari sono in numero decisamente maggiore rispetto a quelle fraudolente. Eppure, è di queste ultime che è interessante l'identificazione.
- Test diagnostici: nel caso di malattie rare, sono molti di più i soggetti sani che i soggetti che ne sono affetti. Eppure lo scopo di un test diagnostico è la rilevazione della patologia.

L'effetto dello sbilanciamento di un dataset sulle performances degli algoritmi di classificazione dipende dal rapporto tra il numero di occorrenze della classe maggioritaria ($N_{majority}$) e il numero di occorrenze della classe minoritaria ($N_{minority}$). Tale rapporto viene denominato *Imbalance Ratio* ed indicato con IR:

$$IR = \frac{N_{majority}}{N_{minority}} \tag{8.1}$$

In letteratura, si può trovare la seguente classificazione del livello di sbilanciamento di un dataset in base al valore di IR [80]:

- $1 < IR < 10$: dataset moderatamente sbilanciato
- $10 \leq IR \leq 100$: dataset sbilanciato
- $IR > 100$: dataset estremamente sbilanciato

Tra lo sbilanciamento di un dataset e l'accuratezza della classificazione esiste un rapporto di proporzionalità inversa: più IR è alto, più influenza in modo negativo le performances dei classificatori. L'impatto dello sbilanciamento sulla classificazione è tuttavia già consistente per bassi valori del rapporto tra la numerosità delle due classi. Questo fenomeno viene descritto in diversi studi ed in particolare in [80], da cui è tratta la Figura 8.1. In essa sono rappresentati i risultati di uno studio condotto testando numerosi algoritmi di classificazione su diversi dataset che avevano in comune la caratteristica di essere sbilanciati, seppur con differenti IR. Sull'asse delle ascisse si trova IR.

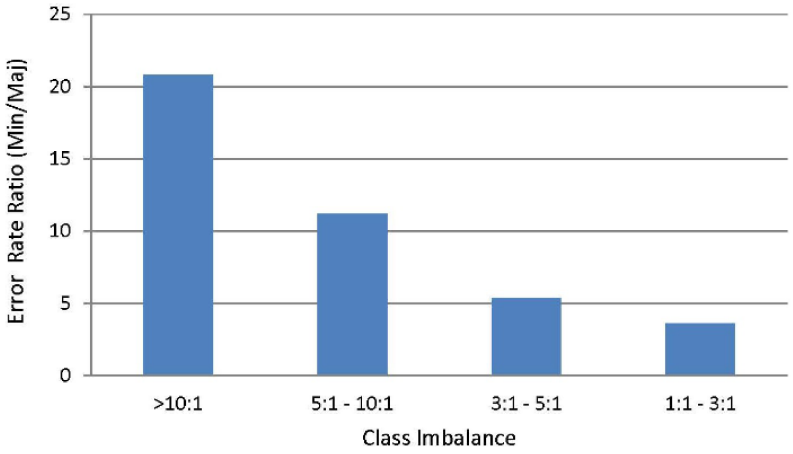


Figura 8.1: Impatto dello sbilanciamento di un dataset sulle performances della classificazione [80]

Sull'asse delle ordinate è presente il rapporto tra il tasso di errore associato alla classificazione della classe minoritaria e il tasso di errore associato alla classificazione della classe maggioritaria, detto *Error Rate Ratio* (ERR). Tale rapporto è stato definito nel modo seguente:

$$ERR = \frac{\frac{PredError_{Minority}}{N_{minority}}}{\frac{PredError_{Majority}}{N_{majority}}} \tag{8.2}$$

I diversi dataset presi in esame nello studio sono stati suddivisi in 4 classi a seconda del valore di IR. Gli istogrammi rappresentano il valore medio di ERR per i dataset appartenenti a ciascuna classe. Come si può notare, il valore medio di ERR cresce all'aumentare di IR e arriva ad essere addirittura superiore a 20 per i dataset in cui la classe maggioritaria è numericamente 10 volte superiore alla classe minoritaria. É da sottolineare però che, anche in presenza di uno sbilanciamento molto limitato, ovvero in casi in cui le osservazioni appartenenti alla classe più numerosa sono meno del triplo rispetto all'altra, il valor medio di ERR è di poco inferiore a 5. La conclusione è che è sufficiente uno sbilanciamento lieve per influenzare notevolmente ed in maniera negativa la classificazione.
Per far fronte alla problematica dello sbilanciamento dei dataset esistono in letteratura diversi approcci, che sono stati riepilogati nella sezione successiva.

8.2 Approcci per la gestione di dataset sbilanciati

Per implementare algoritmi di classificazione in situazioni con dataset sbilanciati si può procedere utilizzando differenti approcci, che appartengono a 3 categorie [69]:

- Approcci *Algorithm-level*, che adattano i classificatori a lavorare con dataset sbilanciati
- Approcci *Data-level*, che modificano il bilanciamento del dataset
- Approcci *Cost-sensitive*, che introducono il costo relativo ad un errore di classificazione

Approcci Algorithm-level Esistono molte strategie che appartengono a questa tipologia. Alcune di esse sono molto specifiche e adatte ad operare solo con un preciso algoritmo di classificazione. Ad esempio, per quanto riguarda il classificatore K-Nearest Neighbours, una soluzione consiste nell'introdurre un peso, dipendente dalle classi, che vada a compensare lo sbilanciamento del dataset. In particolare, definendo un peso maggiore per la classe maggioritaria in fase di training, si favorisce l'algoritmo a trovare un vicino nella classe minoritaria per le istanze utilizzate nella fase di test. [70].
Esistono anche tecniche più generali. Una di queste si definisce *Recognition-Based learning* e consiste nel far apprendere l'algoritmo esclusivamente utilizzando la classe minoritaria. Questo metodo è efficiente nel migliorare il riconoscimento delle istanze appartenenti a quest'ultima classe, ma non è implementabile per tutti i classificatori [73].
Un'ulteriore possibilità consiste nel combinare differenti classificatori. Le tecniche che prevedono questo tipo di approcci sono dette *Ensemble-based Methods*. Le più note sono il *boosting* e il *bagging* [73]. Il Boosting è un procedimento composto da numerose iterazioni. Ad ogni iterazione si prende in esame un diverso classificatore. Esso riceve in input sempre lo stesso training set, ma con la differenza che alle istanze che sono state classificate in maniera sbagliata nelle precedenti iterazioni viene attribuito un peso maggiore [74]. Con questo sistema ad ogni iterazione si focalizza l'attenzione sulle istanze più problematiche per la classificazione, migliorando le prestazioni. Per quanto riguarda il Bagging, noto anche come *Bootstrap Aggregating*, invece, ciò che varia ad ogni iterazione è la composizione del training set. Infatti, ad ogni iterazione un nuovo training set è generato tramite un'estrazione casuale effettuata sul dataset originale [74].

Approcci Data-level: lo SMOTE Il principale vantaggio dell'utilizzo di questo tipo di strategie rispetto a quelle Algorithm-level consiste nel fatto che sono indipendenti dal tipo di classificatore utilizzato.
Sostanzialmente, agire sul dataset originario significa effettuare un *resampling* dei dati a disposizione, con l'obiettivo di migliorare il bilanciamento del dataset originale. Esistono differenti possibilità:

- *Undersampling*: si eliminano alcune istanze della classe maggioritaria
- *Oversampling*: si costruiscono nuove istanze della classe minoritaria o si replicano alcune istanze esistenti
- Metodi ibridi che combinano le due tecniche precedenti

Un primo modo di procedere può essere quello di praticare un *undersampling* o un *oversampling* casuale. Tuttavia, questo approccio non è molto efficiente. Nel caso dell'undersampling casuale, perchè può portare all'eliminazione di dati utili. Nel caso dell'oversampling casuale, perchè aumenta la probabilità di overfitting [69]. Un'altra tecnica che non porta ad un sensibile aumento delle prestazioni dei classificatori è quella dell'oversampling tramite clonazione di alcune istanze. Infatti, è stato dimostrato [75] che questo procedimento genera regioni di decisione più piccole rispetto a quelle del dataset originale. Si tratta dell'effetto opposto a quello desiderato.
Una delle tecniche di resampling che si sono maggiormente affermate in tempi recenti è lo *SMOTE* (Synthetic Minority Over-sampling TEchnique). In questa tecnica, anzichè duplicare alcune istanze, ne vengono generate di nuove, definite *sintetiche*, operando nello spazio degli attributi del

dataset.
L'algoritmo di SMOTE riceve in input la cardinalità della classe minoritaria T, l'ammontare dell'oversampling desiderato per la classe minoritaria N espresso in termini percentuali e un parametro k che indica il numero di istanze da considerare nella costruzione delle nuove osservazioni.
La prima operazione consiste nell'identificare i k elementi più vicini per ogni occorrenza i della classe minoritaria e di salvarli in un vettore denominato K_i. In seguito, per ogni istanza i, si costruisce una nuova istanza h applicando i seguenti passaggi [75]:

- viene estratto casualmente un elemento j appartenente a K_i
- per ogni attributo x che caratterizza le istanze i e j, si calcola il valore che esso assume per h nel modo seguente:
  - si calcola la differenza $d = x_j - x_i$
  - si estrae un numero casuale r compreso tra 0 e 1
  - $x_h = x_i + d * r$

É da sottolineare che il calcolo di x_h corrisponde all'estrazione di un punto a caso posizionato sulla retta che congiunge x_i e x_j. Inoltre, l'operazione di generazione di nuove istanze potrebbe essere ripetuta più volte per ogni occorrenza della classe minoritaria. Questo accade se N è superiore al 100%, se invece N è inferiore a questa soglia solo un sottoinsieme di istanze della classe minoritaria viene utilizzato per la creazione di osservazioni sintetiche.
La Figura 8.2 dà una rappresentazione grafica della generazione di istanze sintetiche nello SMOTE. Dati $n_1, ..., n_5$ gli elementi più vicini a x, si nota che per generare le istanze sintetiche s_1, s_2, s_3 sono stati estratti rispettivamente gli elementi n_1, n_2, n_4. Da notare anche che le istanze sintetiche si trovano a diverse altezze lungo il segmento che collega le due istanze da cui discendono, proprio come descritto precedentemente.

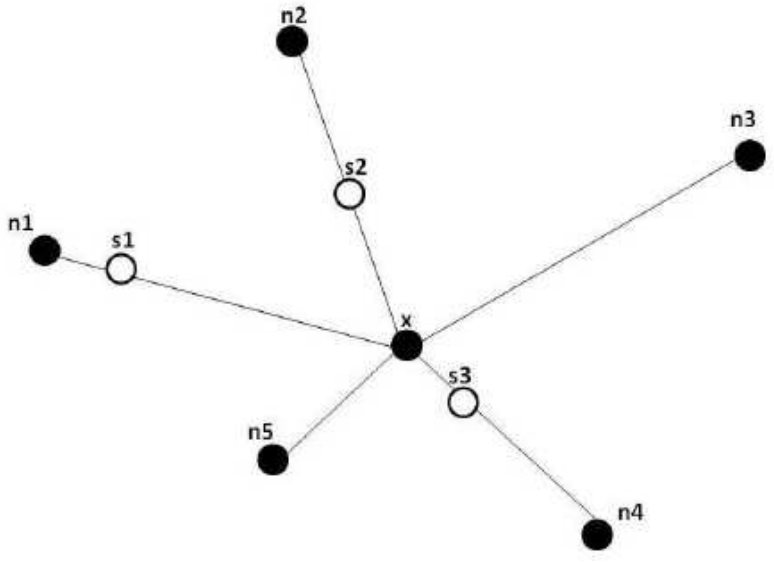


Figura 8.2: Rappresentazione grafica della generazione di nuove istanze nello SMOTE [76]

Uno dei punti di forza di questo approccio è il forte impatto della casualità nella generazione delle nuove istanze [75]. Si tratta di un aspetto che rende lo SMOTE preferibile ad altre tecniche di

oversampling come la clonazione delle istanze. A questo proposito, si può fare riferimento alla Figura 8.3, che riporta i risultati relativi ad un esperimento effettuato su un dataset relativo ad un test diagnostico per rilevare un tumore al seno [75].

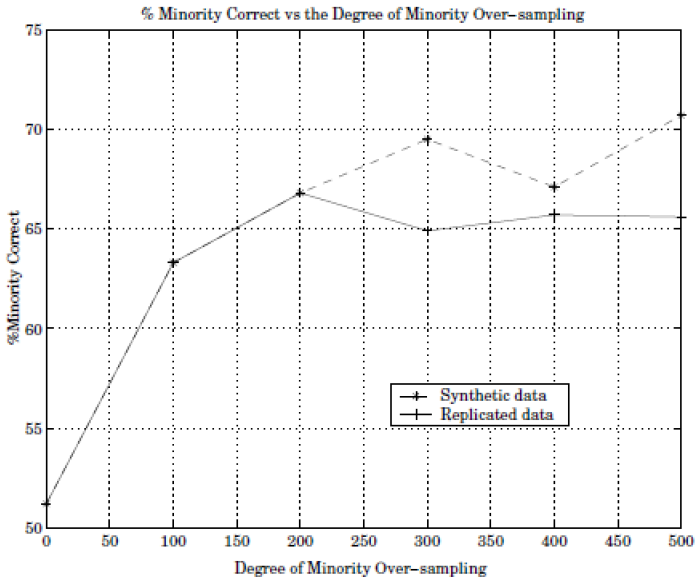


Figura 8.3: Classificazioni della classe minoritaria corrette per diverse tecniche di oversampling [75]

Sulle ascisse è riportato il tasso di oversampling della classe minoritaria, ovvero il parametro N dell'algoritmo dello SMOTE, sulle ordinate la percentuale di istanze della classe minoritaria correttamente classificate. La linea tratteggiata corrisponde allo SMOTE, mentre la linea continua ad un algoritmo di oversampling basato sulla clonazione delle istanze. Il risultato è una dominanza da parte dello SMOTE per valori di N superiori al 200% ed una sostanziale equivalenza dei due algoritmi per valori inferiori a tale soglia.
In generale è stato evidenziato [69] che lo SMOTE migliora le prestazioni degli algoritmi di classificazione più di altre tecniche di oversampling. Ciò ha portato ad una capillare diffusione di questo algoritmo e all'implementazione di sue numerose varianti.

Approcci Cost-sensitive L'idea alla base di questo tipo di strategie è il fatto che, in alcune situazioni pratiche, il costo di un errore di classificazione non è uguale per tutte le classsi.
A titolo di esempio, nel caso di un test diagnostico di una malattia grave, è preferibile classificare erroneamente un individuo sano come malato piuttosto che il contrario. Nel primo caso a ulteriori accertamenti permetteranno la soluzione del problema, nel caso di un individuo malato classificato come malato le conseguenze potrebbero essere fatali [77].
Ci sono diversi modi di implementare questo concetto nella gestione di dataset sbilanciati [69]:

- Metodi diretti, che utilizzano direttamente una variabile di costo all'interno dell'algoritmo di classificazione
- Metodi di meta-apprendimento, che non modificano il classificatore ma che agiscono tramite operazioni di pre-processing sul dataset o di post-processing sull'output

In entrambi i casi, si prospetta il problema della determinazione del costo di un'errata classificazione.

Capitolo 9

Analisi dei risultati

9.1 Principi generali

A coronamento del lavoro svolto, nell'analisi descritta nel presente capitolo ci si è posto l'obiettivo di valutare l'efficacia delle metriche definite nel Capitolo 5, tramite l'implementazione degli algoritmi di classificazione descritti nel Capitolo 6. La scelta dei parametri con i quali eseguire la finestratura per questa analisi conclusiva è stata eseguita in maniera coerente alle analisi sviluppate nel Capitolo 5. In particolare, si è scelta una ampiezza delle finestre di 60 secondi e uno step tra finestre consecutive di 10 secondi.
Per raggiungere gli obiettivi prefissati si è scelto di dividere in due differenti sottofasi la fase di veglia del soggetto:

- la fase di veglia vera e propria, che da qui in avanti si indicherà con la lettera A (Awake)
- la fase immediatamente precedente all'istante di addormentamento, che da qui in avanti si indicherà con la sigla FA (Falling Asleep)

Sulla lunghezza di quest'ultima fase, ovvero su quale sia il lasso di tempo precedente all'addormentamento in cui iniziano a manifestarsi i sintomi fisiologici del sonno, non si riscontra consenso unanime da parte del mondo biomedico. La lunghezza della fase FA deve essere sufficientemente ampia da consentire il riconoscimento dei sintomi della sonnolenza con sufficiente preavviso rispetto al verificarsi dell'addormentamento. Allo stesso tempo, per garantire un riconoscimento efficace, non deve essere troppo ampia. Fatte queste considerazioni si è deciso, sentito anche il parere di esperti del settore, di valutare due differenti valori per la lunghezza della fase FA, pari a 10 minuti e a 15 minuti. Il risultato della configurazione dei parametri di finestratura e della suddivisione delle fasi che sono state appena descritte è un dataset composto da 2388 finestre:

- nel caso di FA della durata di 15 minuti, 1452 della fase A e 936 della fase FA
- nel caso di FA della durata di 10 minuti, 1822 della fase A e 566 della fase FA

Ognuna di queste finestre è stata indicizzata con un ID costruito combinando il soggetto della registrazione e l'indice temporale dell'inizio della finestra.
Gli algoritmi di classificazione che sono stati implementati sono quelli di cui si è discusso nel Capitolo 6, ovvero Regressione Logistica, Alberi decisionali, Random Forest e K-Nearest Neighbors. Nell'analisi degli output di questi algoritmi, si è analizzata la bontà dei risultati ottenuti con l'aiuto degli indici di qualità esposti nel corso del Capitolo 7. Un focus particolare è stato diretto sulla matrice di confusione ed in special modo sulla sensitività degli algoritmi, ovvero sulla loro capacità

di classificare correttamente buona parte delle finestre appartenenti alla fase FA. Infatti, lo scopo principale di questo lavoro è la rilevazione di questa fase, più che della fase di veglia.
Nel Capitolo 6, si è anche parlato dell'importanza di dividere il dataset in un training set su cui costruire il modello e in un test set su cui validarlo. Da questo punto di vista, la validazione è stata eseguita tramite la K-Fold Cross-Validation. La ripartizione del dataset in K sottoinsiemi è stata effettuata con la tecnica dell'impostazione di un attributo di *batch*.
Questo metodo ha lo scopo di non affidare al caso la generazione del test set, ma di controllarla seguendo una logica precisa. Infatti, dato K il numero di valori distinti $v_1, v_2, ..., v_k$ assunti dall'attributo di batch, il dataset originario viene diviso in K sottoinsiemi, ognuno dei quali contiene tutte le osservazioni caratterizzate dallo stesso valore v_i dell'attributo di batch. Poi, seguendo il principio della Cross-Validation, ad ogni iterazione si costruisce il modello su $K-1$ sottoinsiemi del dataset e lo si valida sul K-simo, che funge da test set. In questo caso, si è assegnato il ruolo di attributo di batch all'attributo *Soggetto*. Questo perchè così facendo si vuole simulare la costruzione del modello su un training set costituito da un certo campione di soggetti e la sua validazione su un nuovo soggetto esterno al training set.

Set di metriche utilizzato Nel Capitolo 5 si è proposto un insieme di metriche utili alla descrizione del segnale. Non tutte sono state considerate per l'implementazione dei classificatori. Infatti, inizialmente è stata calcolata la *matrice di correlazione* che ha fatto emergere una forte correlazione tra l'ampiezza media dei picchi di una finestra, $AmpMedia$, e la media tra le distanze euclidee delle coppie massimo-minimo, $EucDistMedia$. L'indice di correlazione per questa coppia di attributi ammonta a 0.989. Inoltre, $EucDistMedia$ presenta una notevole correlazione, 0.678, con la traslazione media dell'asse del segnale rispetto all'origine, Dec. Infine, come era prevedibile, le differenti varianti dell'ampiezza media dei picchi, $AmpMedia$, $AmpMediaMax$ e $AmpMediaMin$, sono tra loro correlate per valori che superano tutti 0.95. Tenendo conto di queste considerazioni, si è deciso di scartare $EucDistMedia$ e di considerare esclusivamente $AmpMedia$ e non le sue varianti. L'insieme delle metriche rimanenti, che da qui in avanti verrà indicato con M, ha quindi una cardinalità pari a 6 ed è composto dalle seguenti metriche: $AmpMedia$, $CorrMaxMin$, $Outliers$, Dec, $PeaksDist$, $DelPeaks$.
Parallelamente all'analisi del set di metriche M, è stato effettuato un confronto con un set di metriche precedentemente implementato, indicato con $M2$, di cardinalità 8. L'obiettivo è quello di ottenere un miglioramento, che può sussistere dal punto di vista dei risultati della classificazione o dal punto di vista di una ridotta complessità computazionale ed interpretativa del modello.

9.2 Analisi dell'ampiezza della fase di addormentamento

Come è stato anticipato nella precedente sezione, un aspetto fondamentale dell'analisi in esame è la corretta identificazione dell'ampiezza della fase FA. I pareri in merito degli esperti del settore sono talvolta discordanti. Per questo motivo, sono stati testati 2 diversi valori dell'ampiezza della fase FA e, precisamente, 10 minuti e 15 minuti, con il set di metriche M. A titolo di esempio, nella Tabella 9.1 vengono riportati i risultati ottenuti con un K-Nearest Neighbours. Il classificatore è pensato per rilevare gli episodi di sonnolenza, quindi per sensitività si considera la capacità di individuare la fase FA e per specificità la capacità di individuare la fase A.
La stessa analisi è stata effettuata implementando altri algoritmi di classificazione. NellaTabella 9.2 vengono riportati i risultati ottenuti con una Regressione Logistica.
Dai valori che sono stati riportati, sembra più promettente la configurazione in cui la fase FA ha un'ampiezza di 15 minuti. Questo vale soprattutto per quanto riguarda la sensitività e la precisione per la classe FA, che sono i valori di maggiore interesse per l'analisi in oggetto. Inoltre, il guadagno dal punto di vista della specificità e della precisione della classe A non è sufficiente a compensare la

	FA 10 **minuti**	FA 15 **minuti**
Sensitività	23,92%	45,33%
Specificità	73,82%	60,60%
Precisione FA	30,85%	44,78%
Precisione A	67,81%	61,13%

Tabella 9.1: Confronto risultati KNN per diversi valori di ampiezza della FA

	FA 10 **minuti**	FA 15 **minuti**
Sensitività	35,34%	40,42%
Specificità	59,36%	57,42%
Precisione FA	25,14%	42,35%
Precisione A	69,95%	55,46%

Tabella 9.2: Confronto risultati Regressione Logistica per diversi valori di ampiezza della FA

perdita che si ottiene per gli altri indici. Queste considerazioni hanno fatto sì che nelle successive analisi si sia sempre considerata una durata della fase FA di 15 minuti.

9.3 Oversampling del dataset tramite SMOTE

Nel Capitolo 8 si sono descritte in modo approfondito le problematiche relative alla classificazione derivanti da una distribuzione non uniforme delle classi. Il problema in esame appartiene proprio alla tipologia dei problemi di classificazione con distribuzione sbilanciata delle classi. Per questo motivo si è deciso di applicare una delle tecniche descritte nel Capitolo 8. In particolare, è stato praticato un oversampling dei dati a disposizione con la tecnica dello SMOTE. Ciò è stato fatto seguendo due approcci differenti, da cui si sono ottenuti i seguenti dataset:

- un dataset da 5000 osservazioni: 1960 della fase FA e 3040 della fase A
- un dataset da 5000 osservazioni: 2500 della fase FA e 2500 della fase A

Nel primo caso, l'obiettivo è quello di aumentare la cardinalità del training set e del test set per migliorare le prestazioni degli algoritmi di classificazione. Nel secondo caso, oltre ad allargare il campione di osservazioni cui attingere nella fase di costruzione dei modelli, si intende ottenere un bilanciamento perfetto delle due classi. Ci si aspetta che questa operazione migliori ulteriormente le prestazioni dei classificatori.
Come prima, si riportano la Tabella 9.3 e la Tabella 9.4 con lo scopo di illustrare alcuni esempi dei risultati ottenuti relativamente a diversi algoritmi di classificazione.

	Dataset originale	**SMOTE sbilanciato**	**SMOTE bilanciato**
Sensitività	45,33%	53,01%	58,68%
Specificità	60,60%	67,14%	60,20%
Precisione FA	44,78%	50,98%	59,59%
Precisione A	61,13%	68,91%	59,30%

Tabella 9.3: Confronto risultati KNN per dataset dalle differenti composizioni

L'analisi delle precedenti tabelle offre diversi spunti di riflessione. Innanzitutto, i risultati ottenuti con l'oversampling sono migliori da quasi tutti i punti di vista rispetto a quelli ottenuti con il

	Dataset originale	**SMOTE sbilanciato**	**SMOTE bilanciato**
Sensitività	27,58%	36,12%	72,40%
Specificità	53,88%	67,63%	38,88%
Precisione FA	29,65%	41,84%	54,22%
Precisione A	51,35%	62,15%	58,48%

Tabella 9.4: Confronto risultati Random Forest per dataset dalle differenti composizioni

dataset originale. Questo è indice del fatto che un maggiore bilanciamento delle classi oppure un allargamento del campione sono operazioni utili a risolvere alcune problematiche da cui il datset originale è affetto.
Il confronto tra i due dataset ottenuti tramite lo SMOTE introduce invece una sorta di trade-off. Infatti, il dataset bilanciato è caratterizzato da un netto miglioramento dal punto di vista delle misure che riguardano la fase FA, ovvero la sensitività e la precisione FA. Allo stesso tempo, però, peggiorano le misure che riguardano la fase A, ovvero la specificità e la precisione A. Ciò è dovuto al fatto che un miglior bilanciamento del dataset consente al classificatore di essere più preciso nel riconoscimento della classe originariamente minoritaria. Questo può avvenire a scapito della classe originariamente maggioritaria, poichè un'alta specificità ed un'alta precisione A possono essere dovute ad una tendenza dell'algoritmo a classificare con più facilità un'occorrenza come A per la prevalenza di quest'ultima classe. Il bilanciamento del dataset dovrebbe servire proprio ad eliminare questa tendenza. La decisione su quale dei due dataset sia meglio utilizzare dipende dagli obiettivi che si vogliono raggiungere. Dal punto di vista del presente lavoro, può essere più utile preferire un riconoscimento più preciso della fase FA e, di conseguenza, il dataset bilanciato.

9.4 Confronto con un set alternativo di metriche

Come è stato anticipato nella Sezione 9.1, il set di metriche M implementato nell'ambito del presente lavoro è stato confrontato con un altro set di metriche, $M2$, che era stato implementato in un precedente lavoro. Da un punto di vista della complessità computazionale ed interpretativa, il nuovo set di metriche rappresenta un passo in avanti poichè la sua cardinalità è minore, essendo costituito da 6 elementi anzichè da 8. Un confronto dal punto di vista dei risultati si può elaborare prendendo in esame le seguenti tabelle, che riportano a titolo di esempio alcuni degli algoritmi di classificazione implementati.

	set di metriche M	**set di metriche $M2$**
Sensitività	58,68%	50,80%
Specificità	60,20%	49,72%
Precisione FA	59,59%	50,26%
Precisione A	59,30%	50,26%

Tabella 9.5: Confronto risultati KNN per diversi set di metriche

I valori riportati nelle Tabelle 9.5, 9.6 e 9.7 evidenziano come i risultati ottenuti con il set di metriche M siano in larga parte migliori rispetto a quelli ottenuti con il set di metriche $M2$. Questo vale soprattutto per il classificatore K-Nearest Neighbors e per la regressione logistica, mentre per l'albero decisionale si registrano risultati moderatamente migliori con il set $M2$.
É stata valutatata anche una possibile integrazione tra i due set di metriche. Tuttavia, la bontà dei risultati ottenuti non è stata tale da giustificare il notevole incremento nella complessità del modello. Un esempio concreto è mostrato nella Tabella 9.8.

	set di metriche M	**set di metriche** $M2$
Sensitività	68,44%	75,16%
Specificità	42,20%	40,32%
Precisione FA	54,21%	55,74%
Precisione A	57,21%	61,88%

Tabella 9.6: Confronto risultati Decision Tree per diversi set di metriche

	set di metriche M	**set di metriche** $M2$
Sensitività	49,00%	40,04%
Specificità	64,04%	55,56%
Precisione FA	57,67%	47,40%
Precisione A	55,67%	48,10%

Tabella 9.7: Confronto risultati Regressione Logistica per diversi set di metriche

	set di metriche M	**set di metriche** $M2 + M$
Sensitività	58,68%	57,04%
Specificità	60,20%	62,24%
Precisione FA	59,59%	60,17%
Precisione A	59,30%	59,16%

Tabella 9.8: Confronto risultati KNN per diversi set di metriche

Come si può notare, tutto ciò che si ottiene è un lieve miglioramento per alcune delle misure, in particolare specificità e precisione FA, una sostanziale equivalenza per la precisione A e addirittura un leggero peggioramento per quanto riguarda la sensitività.

9.5 Confronto tra diversi classificatori

Dall'analisi descritta nelle precedenti sezioni si evince che l'approccio migliore per procedere al riconoscimento della fase FA è quello di considerare un dataset su cui sono state eseguite operazioni di oversampling e di bilanciamento.
Il riconoscimento della fase FA è stato condotto tramite l'implementazione di diversi algoritmi di classificazione. Per ognuno è stata ricercata la configurazione ottimale di parametri in grado di generare i migliori risultati. Nella presente sezione si riportano i risultati ottenuti con i diversi classificatori nelle loro configurazioni ottimali, in modo da evidenziare quali algoritmi risultano essere maggiormente promettenti. La Tabella 9.9 sintetizza i risultati ottenuti.
La prima colonna corrisponde al classificatore K-Nearest Neighbors. Questo algoritmo presenta

	KNN	**Dec. Tree**	**R. Forest**	**Reg. Log.**
Sensitività	58,68%	58,60%	72,40%	49,00%
Specificità	60,20%	62,64%	38,88%	64,04%
Precisione FA	59,59%	61,07%	54,22%	57,67%
Precisione A	59,30%	60,21%	58,48%	55,67%

Tabella 9.9: Risultati ottenuti con le migliori configurazioni dei differenti algoritmi

risultati accettabili sotto tutti i punti di vista, senza picchi di eccellenza e senza valori fortemente

negativi.
La seconda colonna è relativa al Decision Tree. I suoi risultati sono analoghi a quelli ottenuti per il KNN, con tutti gli indici riportati che si aggirano intorno al 60%, senza nè valori particolarmente soddisfacenti nè particolari criticità.
Per quanto riguarda la Random Forest, i cui risultati sono riepilogati nella terza colonna, la situazione non è esattamente la stessa. Infatti, essa presenta una sensitività decisamente più alta rispetto ai valori precedenti, che supera il 70%. D'altra parte, però, la specificità scende sotto al 40%, un valore che non si può considerare soddisfacente.
Infine, come riportato nella quarta colonna, è stato provato anche l'algoritmo della regressione logistica. In questo caso, si ottiene un valore moderatamente più basso rispetto ai precedenti per quanto riguarda la sensitività, sotto al 50%. Gli altri valori si mantengono su livelli analoghi ai precedenti.
In conclusione, gli algoritmi più promettenti sembrano essere il classificatore K-Nearest Neighbors e il Decision Tree, perchè ottengono risultati più omogenei rispetto alla Random Forest e perchè sono caratterizzati da una sensitività sensibilmente più alta rispetto alla regressione logistica.

Capitolo 10

Conclusioni

L'individuazione di una fase di addormentamento collocata in una posizione intermedia tra la veglia e il sonno e conseguentemente il suo riconoscimento automatico sono obiettivi di grande importanza. Il tema è di stretta attualità, considerato il notevole impatto sociale della sonnolenza, soprattutto per quanto riguarda gli incidenti stradali che si possono imputare ad essa.
Il riconoscimento di questa fase di addormentamento passa attraverso l'analisi di segnali fisiologici. In particolare, numerosi possono essere i segnali da analizzare a tale scopo. Nell'ambito del presente lavoro ne sono stati analizzati più di uno.
Una corposa parte del lavoro è stata dedicata alla definizione delle metriche caratterizzanti il segnale. Esse sono state lo strumento utilizzato per la differenziazione delle varie fasi del segnale. L'efficacia delle metriche proposte è stata testata tramite l'implementazione di alcuni algoritmi di classificazione, quali il K-Nearest Neighbor, la Random Forest, gli alberi decisionali e la regressione logistica. Tutti questi modelli sono stati validati tramite la K-Fold Cross-Validation. L'utilizzo di quest'ultima tecnica, con l'accortezza di impostare un attributo di batch corrispondente al soggetto da cui è stata estratta l'osservazione in esame, ha permesso di simulare il comportamento del classificatore in presenza di un nuovo soggetto al di fuori del training set.
I risultati hanno beneficiato dell'effettuazione di un'operazione di oversampling e di bilanciamento del dataset tramite la tecnica dello SMOTE. Gli algoritmi che si sono rivelati più promettenti per la rilevazione della sonnolenza sono il KNN e il Decision Tree.
Questo lavoro lascia aperti possibili spunti di approfondimento. Ad esempio, si potrebbe pensare ad una caratterizzazione del segnale nel dominio delle frequenze, in alternativa alla caratterizzazione sul dominio del tempo che è stata affrontata nel presente lavoro.

Bibliografia

[1] National Sleep Foundation https://www.sleepfoundation.org/press-release/national-sleep-foundation-recommends-new-sleep-times/page/0/1

[2] Khalighi, S., Sousa, T., Pires, G., Nunes, U., *Automatic Sleep Staging: A Computer Assisted Approach for Optimal Combination of Features and Polysomnographic Channels.*, Expert Systems with Applications, 40: 7046–7059

[3] http://healthysleep.med.harvard.edu/healthy/

[4] Beccuti, G., Pannain, S., *Sleep and obesity*, Curr. Opin. Clin. Nutr. Metab. Care, 2011, 14: 402–412

[5] Magee, C.A., Caput, P., Iverson, D.C., *Is sleep duration associated with obesity in older Australian adults?*, J. Aging Health, 2010, 22: 1235–1255

[6] Ayas, N.T., White, D.P., Manson, J.E., Stampfer, M.J., Speizer, F.E., Malhotra, A., Hu, F.B., *A prospective study of sleep duration and coronary heart disease in women*, Arch. Intern. Med., 2003, 163: 205–209

[7] Hublin, C., Partinen, M., Koskenvuo, M., Kaprio, J., *Sleep and mortality: a population-based 22-year follow-up study*, Sleep, 2007, 30: 1245–1253

[8] Landrigan, C.P., Rothschild, J.M., Cronin, J.W., Kaushal, R., Burdick, E., Katz, J.T., Lilly, C.M., Stone, P.H., Lockley, S.W., Bates, D.W., Czeisler, C.A. *Effect of reducing interns' work hours on serious medical errors in intensive care units*, New England Journal of Medicine, 2004, 351: 1838–1848

[9] Tefft, B.C. *The Prevalence and Impact of Drowsy Driving*, AAA Foundation For Traffic Safety, 2010

[10] Lyznicki, J.M., Doege, T.C., Davis, R.M., Williams, M.A. *Sleepiness, driving, and motor vehicle crashes*, JAMA, 1998, 279: 1908–1913

[11] Goncalves, M., Amici, R., Lucas, R., Akerstedt, T., Cirignotta, F., Horne, J., Léger, D., Mc Nicholas, W.T., Partinen, M., Téran-Santos, J., Peigneux, P., Grote, L., *Sleepiness at the wheel across Europe: a survey of 19 countries*, J Sleep Res., 2015, 24: 242–253

[12] Owens, J. M., Dingus, T. A., Guo, F., Fang, Y., Perez, M., McClafferty, J., Tefft, B., *Prevalence of Drowsy Driving Crashes: Estimates from a Large-Scale Naturalistic Driving Study. (Research Brief.)*, Washington, D.C.: AAA Foundation for Traffic Safety. 2018

[13] Garbarino S., Pitidis A., Giustini M., Taggi F., Sanna A., *Motor vehicle accidents and obstructive sleep apnea syndrome: A methodology to calculate the related burden of injuries*, Chron Respir Dis. 2015; 13:1–9

[14] Hillman, D., Mitchell, S., Streatfield, J., Burns, C., Bruck, D., Pezzullo, L., *The economic cost of inadequate sleep*, Sleep J., 2018, 1–13

[15] Royal, D., *National Survey of Distracted and Drowsy Driving*, U.S. Department of Transportation, National Highway Traffic Safety Administration (NHTSA). 2003

[16] Sadeghniiat-Haghighi, K., Yazdi, Z., Kazemifar, A. M., *Sleep quality in long haul truck drivers: A study on Iranian national data*, Chinese Journal of Traumatology, 2016, 19: 225–228

[17] Cassenti, D.N., *Advances in Human Factors in Simulation and Modeling*, Springer, 2018

[18] Thiffault, P., Bergeron, J., *Monotony of road environment and driver fatigue: a simulator study*, Accident Analysis and Prevention. 2003, 35: 381–391

[19] Akerstedt, T., Peters, B., Anund, A., Kecklund, G., *Impaired alertness and performance driving home from the night shift: a driving simulator study*, Journal of Sleep Research. 2005, 14: 17–20

[20] Rogé, J., Pébayle, T., El Hannachi, S., Muzet, A., *Effect of sleep deprivation and driving duration on the useful visual field in younger and older subjects during simulator driving*, Vision Research. 2003, 43: 1465–1472

[21] Roebuck, A., Monasterio, V., Gederi, E., Osipov, M., Behar, J., Malhotra, A., Penzel, T., Clifford, G.D., *A review of signals in sleep analysis*, Physiol. Meas., 2014, 35: 1–57

[22] Cammarota, C., Rogora, E., *Alcune applicazioni della Matematica all'analisi dell'elettrocardiogramma*, Bollettino dell'Unione Matematica Italiana. 2007, 10: 537–562

[23] Michail, E., Kokonozi, A., Chouvarda, I., Maglaveras, N., *EEG and HRV markers of sleepiness and loss of control during car driving*, Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference. 2008.

[24] Vicente, J., Laguna, P., Bartra, A., Bailón, R., *Drowsiness detection using heart rate variability*, Medical and Biological Engineering and Computing. 2016, 54: 927-937

[25] Webster, J.G., *Designe of Pulse oximeters*, IOP Publishing, Bristol, UK, 2003

[26] Carruthers, D.M., Harrison, B.D.W., *Arterial blood gas analysis or oxygen saturation in the assessment of acute asthma?*, Thorax. 1995, 50: 186–188

[27] DeMeulenaere, S., *Pulse Oximetry: Uses and Limitations*, The Journal of nurse Practitioners. 2007, 3: 312–317

[28] www.oximetry.org

[29] www.dealry.it

[30] Boke, C., Shetty, A., Kadam, A., Jadhav, S., Pakhmode, S., Barahate, S., *Smart Band for Drowsiness Detection to Prevent Accidents*, International Journal of Innovative Research in Science, Engineering and Technology, 2016, 5.

[31] Allen, J., *Photoplethysmography and its application in clinical physiological measurement*, Physiological Measurement, 2007, 28: 1–39

[32] Komalla, A.R., *A method for estimation of oxygen saturation for pulse oximeter*, Journal of Engineering Technology, 2017, 6: 618–627

[33] Kyriacou, P.A., *Pulse oximetry in the oesophagus*, Physiological Measurement, 2005, 1

[34] Stubán, N., Masatsugu, N., *Non-invasive calibration method for pulse oximeters*, Electrical Engineering, 2008, 52: 91–94

[35] Shafique, M., Kyriacou, P.A., *Photoplethysmographic signals and blood oxygen saturation values during artificial hypothermia in healthy volunteers*, Physiological Measurement, 2012, 33: 2065–2078

[36] Oak, S.S., Aroul, P., *How to Design Peripheral Oxygen Saturation (SpO2) and Optical Heart Rate Monitoring (OHRM) Systems Using the AFE4403*, Texas Instruments, Appliacation Report, 2015

[37] Bagha, S., Shaw, L., *A Real Time Analysis of PPG Signal for Measurement of SpO2 and Pulse Rate*, International Journal of Computer Applications, 2011, 36

[38] Nitzan, M., Noach, S., Tobal, E., Adar, Y., Miller, Y., Shalom, E., Engelberg, S., *Calibration-Free Pulse Oximetry Based on Two Wavelengths in the Infrared - A Preliminary Study*, Sensors, 2014, 14: 7420–7434

[39] Frasca, S., *Analisi dei Segnali*, Università di Roma La Sapienza, Dipartimento di Fisica, 2006

[40] Scarpiniti, M., *Elementi di Teoria dei Segnali*, Università di Roma 1

[41] Bertoni, A., Campadelli, P., Grossi, G., *Introduzione all'elaborazione dei segnali*, Università degli Studi di Milano, 2010

[42] The Health Physics Society, www.hpssociety.info/news/ideal-low-pass-filter.html
[43] Miller, P., *Low-pass and high-pass filters*, University of California San Diego, 2006
[44] Oppenheim, A.V., Schafer, R.W., *Discrete-time signal processing*, Prentice Hall, 1989
[45] Verdoliva, L., *Analisi dei sistemi nel dominio della frequenza*, Università degli Studi di Napoli, 2010
[46] Lombardo, P., *Telecomunicazioni per l'Aerospazio*, Università di Roma La Sapienza
[47] https://www.teuniz.net/edfbrowser/
[48] https://www.spyder-ide.org/
[49] https://rapidminer.com/
[50] https://www.cs.waikato.ac.nz/ml/weka/
[51] https://www.teuniz.net/edf2ascii/
[52] Ross, S.M., *Introduction to Probability and Statistics for Engieers and Scientists*, Elsevier Academic Press, 2004
[53] James, G., Witten, D., Hastie, T., Tibshirani, R., *An Introduction to Statistical Learning*, Springer, 2014
[54] Subramanian, J., Simon, R., *Overfitting in prediction models - Is it a problem only in high dimensions?*, Contemporary Clinical Trials, 2013, 36:636–641
[55] www.vitalflux.com
[56] Park, H.A., An Introduction to Logistic Regression: from Basic Concepts to Interpretation with Particular Attention to Nursing Domain, J Korean Acad Nurse, 2013, 2:154–164
[57] Safavian, S.R., Landgrebe, D., *A Survey of Decision Tree Classifier Methodology*, IEEE Transactions on Systems, Man and Cybernetics, 1991, 3
[58] Wu, J., Cui, Z., Sheng, V., Shi, Y., Zhao, P., *Mixed Pattern Matching-Based Traffic Abnormal Behavior Recogition*, The Scientific World Journal, 2014, 1
[59] Cavaioni, M., *Machine Learning: Decision Tree classifier*, Medium Corporation, 2017
[60] Gislason, P.O., Benediktsson, J.A., Sveinsson, J.R., *Random Forests for land cover classification*, Pattern Recognitions Letters, 2006, 27:294–300
[61] Pal, M., *Random forest classifier for remote sensing classification*, International Journal of Remote Sensing, 2005, 1:217–222
[62] Isied, A., Tamimi, H., *Using Random Forest (RF) as a transfer learning classifier for detecting Error-Related Potential (ErrP) within the context of P300-Speller*, Bernstein Conference, 2015
[63] Soltoff, B., *Computing for the Social Sciences*, University of Chicago, 2016
[64] Moore, A.W., *Cross-validation for detecting and preventing overfitting*, Carnegie Mellon University, 2001, www.cs.cmu.edu/ awm
[65] Kohavi, R., *A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection*, Internationl Joint Conference of Artificial Intelligence, 1995
[66] https://my.oschina.net
[67] Visa, S., Ramsay, B., Ralescu, A.L., Van der Knaap, E., *Confusion Matrix-based feature selection*, MAICS, 2011:120–127
[68] Ling, C.X., Sheng, V.S., *Class Imbalance Problem*, in: Sammut, C., Webb, G.I., Encyclopedia of Machine Learning. Springer, Boston, 2011
[69] Lopez, V., Fernandez, A., Garcia, S., Palade, V., Herrera, F., *An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics*, Information Sciences, 2013, 250: 113–141
[70] Kotsiantis, S., Kanellopoulos, D., Pintelas, P., *Handling imbalanced datasets: a review*, GESTS International Transactions on Computer Science and Engineering, 2006, 30
[71] https://www.medcalc.org
[72] Hashemian, A.H., Beiranvand, B., Rezaei, M., Bardideh, A., Zandkarimi, E., *Comparison of Artificial Neural Networks and Cox Regression Models in Prediction of Kidney Transplant Survival*, International journal of Advanced Biological and Biomedical Research, 2013, 1

[73] Abd Elrahman, S.M., Abraham, A., *A Review of Class Imbalance Problem*, Journal of Network and Innovative Computing, 2013, 1:332–340

[74] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F., *a, A review on ensembles for class imbalance problem: bagging, boosting and hybrid based approaches*, IEEE Transactions on Systems, Man and Cybernetics, 2012, 42:463–484

[75] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., *SMOTE: Synthetic Minority Over-sampling Technique*, Journal of Artificial Intelligence Research, 2002, 16:321–357

[76] Cateni, S., Colla, V., *Fuzzy Inference System for Data Processing in Industrial Applications*, IntechOpen, 2012

[77] Zadrozny, B., Elkan, C., *Learning and Making Decisions when Costs and Probabilities are Both Unknown*, IEEE International Conference of Data Mining, 2003:435–442

[78] Maimon, O., Rokach, L., *Data Mining and Knowledge Discovery Handbook*, Springer, 2010

[79] Krzanowski, W.J, Hand, D.J., *ROC curves for continuous data*, Chapman and Hall/CRC, 2009

[80] Weiss, G.M., Provost, F., *Learning when Training Data are Costly: The Effect of Class Distribution on Tree-Induction*, Journal of Artificial Intelligence Research, 2003, 19:315–354