UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA DELL'INFORMAZIONE

SVILUPPO DI ALGORITMI DI MACHINE LEARNING PER BCI

Relatrice: Dott.ssa Giulia Cisotto

Laureando: Elia Nasato

ANNO ACCADEMICO 2021 - 2022

Data di laurea : 14/03/2022

Sommario

La tesi si propone di confrontare diverse strategie di sovra-campionamento e sotto-campionamento di un dataset misto EEG-EMG, a classi sbilanciate, tramite clustering.
Nello specifico, si è impegato k-means per valutare il campionamento di un dataset pubblico EEG-EMG ottenuto mentre 8 soggetti sani svolgevano due diversi movimenti della mano destra.
K-means è stato impiegato per suddividere in modo un-supervised i campioni di EEG-EMG nei due cluster relativi ai due diversi movimenti. Si sono poi calcolate le distanze tra i centroidi e la dimensione dei cluster per valutare le modifiche nella composizione del dataset sovra-campionato o sotto-campionato, rispettivamente. Inoltre, è stata applicata la normalizzazione, per soggetto, e ripetuta l'analisi sopra. I risultati principali hanno dimostrato un miglioramento nei cluster calcolati con i dati sovra-campionati e inoltre, la normalizzazione ha aumentato significativamente la distanza tra centroidi.
In futuro è necessario estendere lo studio tramite prove ripetute di sovra-campionamento e sotto-campionamento per aumentare la significatività statistica dei risultati trovati. Inoltre, tali risultati potrebbero essere replicati in svariati studi che coinvolgono l'uso di segnali EEG, EMG ed entrambi con dataset sbilanciati.

Indice

Introduzione **2**

Background **3**

1.1 Il sistema neuromuscolare 3

1.2 Elettroencefalogramma (EEG) 6

1.2.1 Acquisizione ed elaborazione EEG 7

1.3 Elettromiogramma (EMG) 9

1.4 Magnitude Squared Coherence (MSC) 11

1.5 Over- sampling con SMOTE 12

1.6 Algoritmo K-Means 15

1.7 Stato dell'arte 15

1.7.1 Dataset sbilanciati 16

1.8 Dataset EEG, EMG combinati 18

Metodi **19**

2.1 Dataset WAY-EEG-GAL 19

2.2 Descrizione esperimenti 23

2.3 Implementazione Matlab 24

Risultati e discussioni **26**

3.1 Analisi dei dati 26

Conclusioni **33**

Bibliografia **34**

Introduzione

Per monitorare il movimento del corpo umano esistono diverse tipologie di strumentazione e di analisi. Questa tesi si focalizza sull'uso congiunto di EEG, che acquisisce il segnale cerebrale, e di EMG che invece acquisisce il segnale muscolare. Studi recenti [[1], [2]]hanno analizzato questi due segnali e hanno provato che per migliorare la precisione e l'accuratezza della classificazione del movimento (task) è più efficace usare un sistema ibrido piuttosto che trattare i segnali singolarmente. Si consideri un esperimento dove si vogliono registrare 10 persone che compiono 10 movimenti diversi degli arti superiori. E' lecito pensare che ogni persona impieghi un tempo diverso sia alle altre sia alla tipologia di movimento; o ancora, pensare che ogni task interessi regioni diverse del sistema nervoso. Si vanno quindi a registrare dati che tra loro presentano delle diversità o squilibri non solo a livello di "ampiezze" ma anche a livello di quantità di campioni rilevati per esperimento.
Dataset sbilanciati possono compromettere la correttezza della classificazione dei task. Lo studio di questa tesi vuole porre l'attenzione a problemi di questo genere, ovvero di dataset sbilanciati e valutare la classificazione (basata su SVM) mediante un algoritmo di clustering applicato a dataset ricampionat in modo tale da equilibrare le classi. Nel capitolo 1 vengono introdotti i segnali EEG-EMG, viene spiegato come viene estratta la feature MSC, come funziona l'algoritmo di sovracampionamento SMOTE e l'algoritmo di clusterizzazione K-Means, inoltre viene anche spiegato in modo più dettagliato il problema di avere un dataset sbilanciato. Nel capitolo 2 viene presentato il dataset usato e viene descritto l'esperimento eseguito e la sua relativa implementazione in Matlab. Nel capitolo 3 si discutono i risultai ottenuti.

Background

1.1 Il sistema neuromuscolare

Il sistema Neuromuscolare è composto dal sistema neurale e dal sistema muscolare, esso rappresenta il collegamento tra il cervello e l'apparato muscolare.
Il sistema nervoso è l'insieme degli organi e delle strutture che permettono di trasmettere segnali tra le diverse parti del corpo, di coordinare le sue azioni e funzioni volontarie e involontarie, sia fisiche che psicologiche. Il sistema nervoso è composto dal cervello, dal midollo spinale, dagli organi di senso e dai nervi. Il sistema nervoso si può distinguere in due parti: il sistema nervoso centrale (SNC) e il sistema nervoso periferico (SNP).
Il SNC è formato dall'encefalo e dal midollo spinale, ha il compito di elaborare le informazioni che vengono dal SNP e di fornire le risposte che verranno redistribuite nell'organismo attraverso il sistema nervoso periferico.
Il SNC può essere diviso in due parti: il sistema nervoso autonomo e il sistema nervoso somatico. Il sistema nervoso autonomo è formato da neuroni presenti nel cervello e nel midollo spinale; a formarlo sono neuroni il cui corpo è localizzato nel cervello o nel midollo spinale i cui prolungamenti sono diretti verso strutture, detti gangli, a livello dei quali entrano in contatto con il corpo di altri neuroni. Il sistema nervoso somatico è formato da singoli neuroni che si frappongono tra il sistema nervoso centrale e l'organo cui deve essere connesso. Il corpo cellulare di questi neuroni può trovarsi nel cervello o nel midollo spinale; è possibile distinguerne di due tipi: i neuroni sensitivi, i cui prolungamenti formano le fibre nervose che inviano le informazioni provenienti dalla pelle e dagli organi di senso verso il sistema nervoso centrale, e i motoneuroni da cui partono le fibre nervose dirette verso i muscoli scheletrici, quelli che vengono mossi volontariamente.
L'unità funzionale del sistema nervoso è la cellula nervosa, o neurone. Il centro di controllo del neurone è il soma, il quale è provvisto di un nucleo e nucleolo. Attorno al soma si posso-

no vedere i dentriti, il cui compito è quello di ricevere segnale da altri neuroni connettendosi ad altri dentriti o ad assoni.[3]

Le attività nel SNC sono principalmente correlate al trasferimento di correnti sinaptiche tra giunzioni (chiamate sinapsi) di assoni e dentriti.

Un potenziale d'azione (AP) è un evento di breve durata in cui l'energia di una cellula aumenta rapidamente per poi scendere, può anche essere visto come l'informazione trasmessa da un nervo. I potenziali d'azione sono causati da uno scambio di ioni tra la membrana del neurone e sono un temporaneo cambiamento del potenziale il quale viaggia attraverso l'assone. Inizialmente la membrana si depolarizza, diventando positiva, e produce un picco per poi ritornare ad uno stato di quiete. Questi AP hanno velocità di trasmissione variabile da 1m/s a 100 m/s, in intervalli di tempo di circa $5ms$. Si può misurare un potenziale di $60-70mV$ con polarità negativa al livello della membrana del neurone variabile in base all'attività sinaptica.

Se un potenziale d'azione (AP) che viaggia lungo una fibra incontra una sinapsi eccitatoria, allora nel neurone che la segue verrà percepito un EPSP (Excitatory Post-Synaptic Potential). Se più di un AP viaggiano nella medesima fibra, si avrà una somma di tali segnali; se invece finisce in una inibitoria, allora si troverà un IPSP (Inhibitory Post-Synaptic Potential).

La generazione di un AP avviene quando la somma di tutti gli EPSP ed IPSP produce un

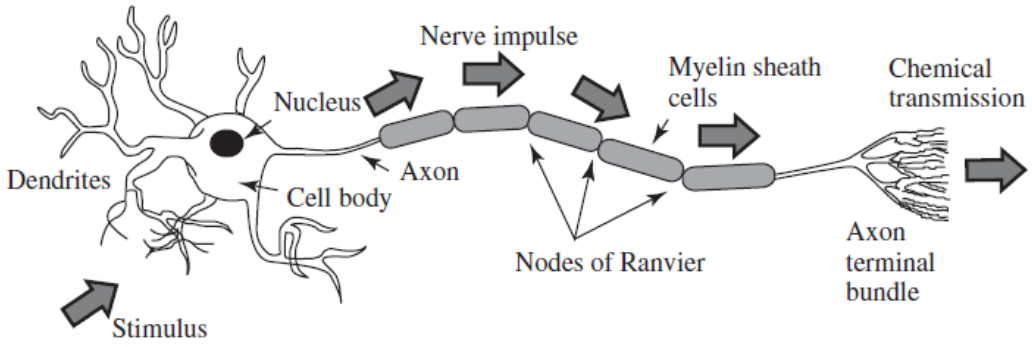

Figura 1.1: Struttura base di un neurone

potenziale superiore a una certa soglia. In questo caso si aprono i canali di sodio (Na^+) e lo ione si muove verso l'interno data la densità superiore esterna, incrementando il potenziale.

I canali di potassio (K^+) impiegano più tempo per aprirsi, la fase di depolarizzazione ha tempo di arrivare a dei valori di circa $10mV$. A questo punto gli ioni di potassio escono dalla membrana creando un effetto di ripolarizzazione verso il valore iniziale.

Questo processo continua fino a raggiungere un valore di circa $-90mv$. Questa fase è chia-

mata iperpolarizzazione e serve a prevenire che il neurone non venga sollecitato da un altro stimolo nello stesso intervallo temporale. La durata di questa fase è di circa 2*ms* ed è chiamata periodo refrattario.[4]

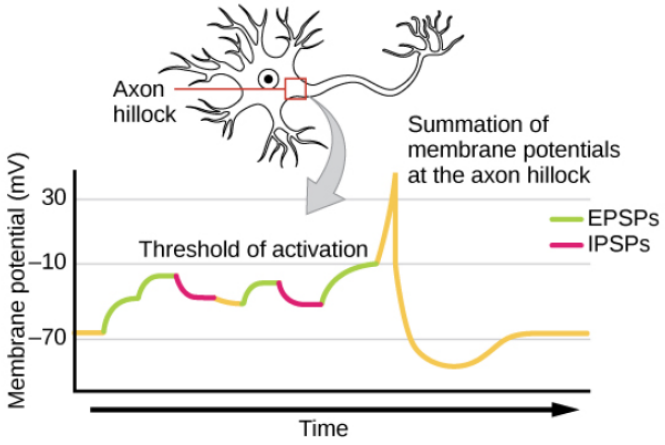

Figura 1.2: EPSP e IPSP che insieme accumulano potenziale su una fibra nervosa prima di raggiungere il valore di soglia

1.2 Elettroencefalogramma (EEG)

L'elettroencefalogramma è la misura di un potenziale che rispecchia l'attività elettrica del cervello umano. Il sistema EEG è largamente usato da scienziati per studiare le funzioni del cervello e per diagnosticare disfunzionamenti neurologici come epilessia, tumori al cervello o disturbi nel sonno. Il primo sistema di registrazione EEG è stato inventato dal neuropsichiatra Hens Berger nel 1929, il quale con il termine tedesco 'elektrenphalogramm' ha dato una rappresentazione grafica delle correnti elettriche generate all'interno del cervello.
In base a dove il segnale è preso: sullo scalpo o intracranico; esistono due tipi di registrazione EEG. Nel primo caso si applicano dei piccoli dischi, chiamati elettrodi, posizionati in posti differenti sulla superficie della testa. Nel secondo caso invece si applicano piccoli elettrodi all'interno della scatola cranica Ogni elettrodo è collegato ad un amplificatore e alla macchina di registrazione. Quando un neurone si attiva vengono prodotte delle correnti sinaptiche nei dentriti, le quali a loro volta generano dei campi elettrici che vengono rilevati dagli elettrodi e trasmessi amplificati alla macchina. Ogni misura contiene rumore interno dovuto all'attenuazione del cranio, rumore esterno o rumore del sistema EEG spesso associato alla corrente di alimentazione. Di conseguenza c'è bisogno di un gran numero di neuroni che lavorino in sincrono per per produrre corrente sufficiente da essere rilevata in superficie.

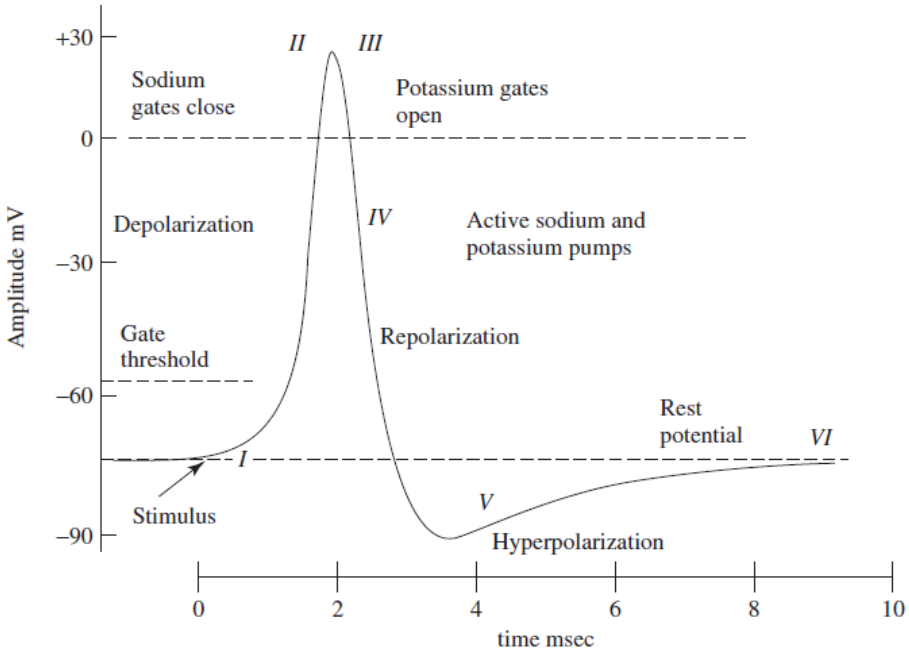

Figura 1.3: Depolarizzazione, ripolarizzazione e iperpolarizzazione di un neurone

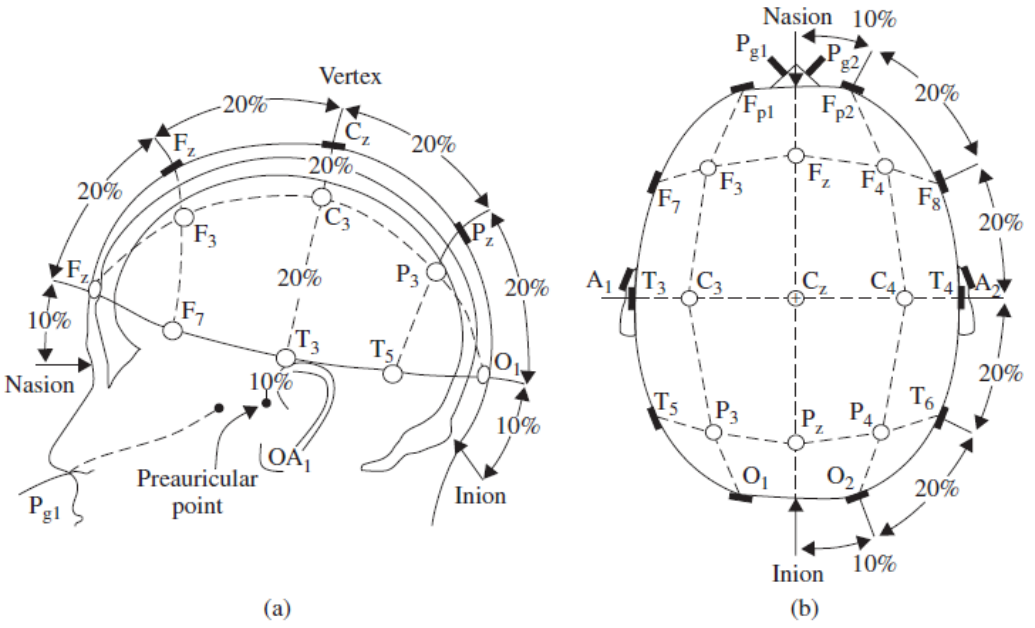

Figura 1.4: Posizionamento degli elettrodi secondo il Sistema Internazionale 10-20

1.2.1 Acquisizione ed elaborazione EEG

Il posizionamento degli elettrodi è importante perchè differenti lobi della corteccia cerebrale sono responsabili per il processo di differenti tipi di attività. Il metodo standard usato per il posizionamento degli elettrodi sulla superficie della testa è chiamato Sistema Internazionale 10-20, prevede apppunto una distanza tra elettrodi del 10% della lunghezza fronte-retro lungo l'asse longitudinale e una distanza del 20% della larghezza destra-sinistra lungo l'asse trasversale. Il posizionamento degli elettrodi è determinato da due punti: il nasion, ovvero il punto intermedio fra fronte e naso e l'inion, cioè la 'punta' ossea che si può sentire alla base del cranio, a metà del retro della testa. Ogni elettrodo è identificato da una lettera: F,T,C,P,O e da un numero pari se collocato nell'emisfero destro, dispari se in quello sinistro oppure da una 'z' per indicare la linea centrale.

Un importante analisi del segnale EEG compiuta da esperti clinici del settore rende in grado il riconoscimento dei vari ritmi nel cervello nei segnali EEG. Negli adulti in salute, l'ampiezza e la frequenza di questi segnali cambia da persona a persona e in stato di veglia o di sonno. Le caratteristiche di queste onde variano anche in base all'età. Ci sono cinque onde principali che variano in base al range di frequenza: alpha (α), theta (θ), beta (β), delta (δ)e gamma (γ). Il ritmo δ giace nel range $[0.5-4Hz]$, queste onde sono associate allo stato di sonno profondo e possono essere rilevate anche nello stato di veglia. Rappresenta le onde più lente e con ampiezza maggiore.

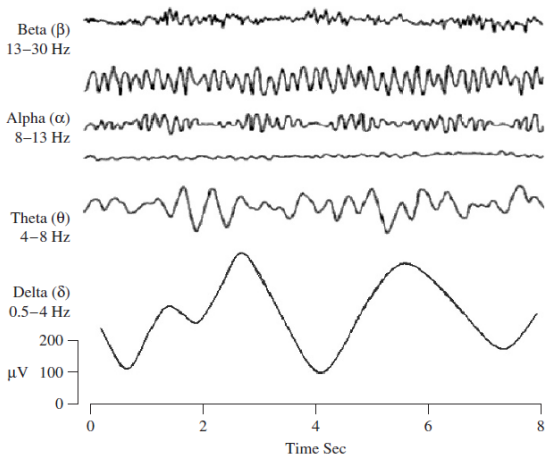

Figura 1.5: Esempio di diversi ritmi

Il ritmo θ comprende il range $[4-7.5Hz]$, queste onde sono state associate allo stato di ispirazione creativa e profonda meditazione. L'onda theta è spesso accompagnata da altre frequenza e sembra essere connessa al risveglio. La sua ampiezza può superare i $20\mu V$

Il ritmo α ha un range di frequenza di $[8-13Hz]$, spesso appaiono come un segnale di forma rotonda o sinosuidale, le onde alpha indicano uno stato di rilassamento, esse compaiono sopratutto nella regione occipitale sul retro della testa. La sua ampiezza varia fra i 30 e i $50\mu V$.

Il ritmo β si presenta a frequenza di $[13-30Hz]$, viene rilevato a livello dell'area frontale ed emerge quando il soggetto è in uno stato di concentrazione o è intento a risolvere problemi. Un livello alto di queste onde si può verificare quando la persona si trova in uno stato di panico.

Il ritmo γ ha frequenza maggiori dei $30Hz$ in su, queste onde vengono associate a funzioni motorie e cognitive.

In generale tutte le frequenza sopra gli $80Hz$ non vengono considerate.[4]

1.3 Elettromiogramma (EMG)

Il segnale EMG è un segnale biomedico, quindi è funzione del tempo; può essere descritto in base la sua ampiezza, frequenza e fase. Esso misura la corrente elettrica generata dalla contrazione muscolare durante attività neuromuscolari. Il sistema nervoso controlla le attività muscolari (contrazione e rilassamento), quindi il segnale EMG è un segnale complicato perchè è controllato dal sistema nervoso e dipende dall'anatomia e dalle proprietà fisiologiche del muscolo. Viaggiando attraverso i vari tessuti acquisisce un rumore aggiuntivo. Il tessuto muscolare conduce potenziali elettrici simili a quelli nei nervi, per cui questi segnali elettrici prendono il nome di potenziali d'azione muscolari. Quando viene rilevato e registrato un segnale EMG, ci sono due problemi principali per quanto riguarda l'integrità del segnale. Il primo è il rapporto segnale-rumore, che fa riferimento all'energia del segnale e del rumore. Il secondo problema è invece la distorsione del segnale, ovvero quanto il contributo relativo di ogni componente in frequenza nel segnale non viene alterato.

Esistono due tipi di elettrodi per registrare un segnale muscolare: uno invasivo e uno non invasivo. Quando un EMG è acquisito da elettrodi montati direttamente sulla pelle, il segnale è un composito di tutti i potenziali di azione delle fibre muscolari che si trovano nei muscoli sottostanti la pelle. Questi AP si verificano ad intervalli non regolari. Quindi, in ogni momento, il segnale EMG può avere un potenziale positivo o negativo. I potenziali di azione delle fibre muscolari possono essere presi usando un elettrodo a filo o ad ago posizionato direttamente nel muscolo. La combinazione dei potenziali d'azione delle fibre muscolari di una singola unità motoria viene chiamata Motor Unit Action Potential (MUAP) Qui sottostante una equazione che esprime un semplice modello di segnale EMG:

$$x(n) = \sum_{r=0}^{N-1} h(r)e(n-r) + w(n) \tag{1.1}$$

$x(n)$ rappresenta il modello del segnale EMG, è una combinazione lineare di N MUAP con l'aggiunta di un rumore gaussiano bianco a media nulla $w(n)$. Il segnale prelevato dagli elettrodi viene amplificato da almeno un amplificatore. Prima di essere visualizzato o registrato, il segnale viene processato prima per eliminare rumori a basse o alte frequenze e in seguito rettificato.

Il muscolo è formato da un gruppo di cellule specializzate capaci del rilassamento e della contrazione del muscolo. La funzione principale di queste cellule è quella di generare forza, movimento e l'abilità di comunicare parlando o scrivendo o usando altri modi per esprimer-

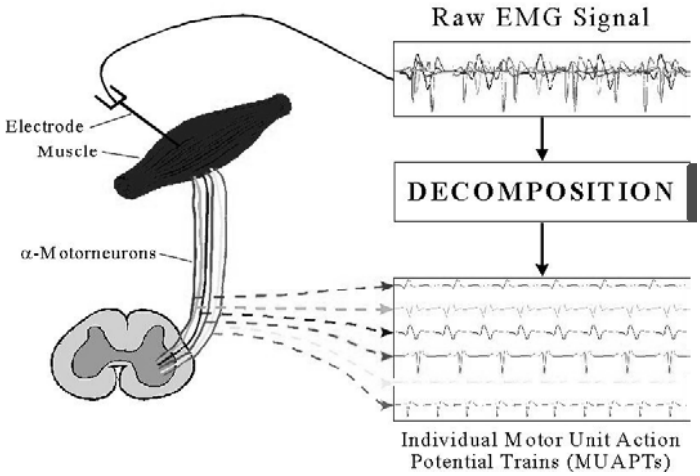

Figura 1.6: Decomposizioni dei MUAP di un segnale EMG

si. Il tessuto muscolare ha quattro funzioni chiavi: muoversi, spostare oggetti con il corpo, fornire una stabilità e generare calore. Tre tipi di muscoli possono essere identificati in base a struttura, proprietà di contrazione e meccanismi di controllo: muscoli scheletrici, muscoli lisci e muscoli cardiaci. L'EMG viene usato per lo studio dei primi, i quali sono attaccati direttamente all'osso e le loro contrazioni sono responsabili per il supporto e il movimentto dello scheletro. Le contrazioni del muscolo scheletrico sono avviate da impulsi generati da neuroni e generalmente sono azioni dovute ad un controllo volontario. Le fibre muscolari scheletriche sono ben fornite di neuroni per ogni contrazione, chiamati "motor neuron", questi ultimi si trovano in prossimità dei tessuto muscolare senza però esserne collegati. In genere un "motor neuron" fornisce stimolazione a più fibre muscolari. In risposta allo stimolo di un neurone la fibra muscolare si depolarizza, lascaindo scorrere il segnale sulla superficie mentre si contrae. Questa depolarizzazione, accompagnata al movimento di ioni, genera un campo elettrico vicino alla fibra muscolare. Un segnale EMG è la somma di MUAP che rappresentano la risposta al muscolo alla stimolazione del neurone. Il segnale EMG si presenta in natura in modo aleatorio e viene generalmente modellato come un processo di impulso filtrato. La MUAP è il filtro e il processo di impulso è un impulso neurale, spesso modellati come un processo di Poisson.[5]

1.4 Magnitude Squared Coherence (MSC)

La selezione delle features è riconosciuta come uno step critico in molte applicazioni di assistenza sanitaria. è ben risaputo che trovare un piccolo numero di features altamente discrimatorie può portare a miglioramenti significativi nella classificazione. Per ridurre la complessità di un dataset multimodale che include segnali EEG e EMG, ci sono due approci: selezione manuale o tecniche di machine learning. Diversi studi, tra cui [6], hanno dimostrato come la fusione di due segnali faccia aumentare le prestazioni di classificazione. L'uso di un approcio multimodale rispetto a quello monomodale ha riportato un miglioramento in termini di precisione di classificazione, quantificato con un aumento percentuale dal 70% al 90%. La fusione dei segnali è avvenuta tramite MSC (Magnitude Squared Coherence), che si ottiene prendendo la normalizzazione della densità spettrale incrociata tra EEG ed EMG. In particolare la feature MSC tra due processi stazionari stocastici $x(t)$ e $y(t)$, che in questa tesi sono rappresentati da segnali EEG ed EMG, è definita come

$$\gamma_{xy}^2(f) = \frac{\left|P_{xy}(f)\right|^2}{P_{xx}(f)P_{yy}(f)} \tag{1.2}$$

dove $P_{xy}(f)$ è la densità spettrale incrociata e $P_{xx}(f)$ e $P_{yy}(f)$ sono le densità autospettrali alla frequenza f [7].

Per la selezione delle features sono stati usati i metodi FeSC (Feature Selection with Consensus), pFSFS (p-value filtering with forward sequential feature selection) e LASSO(Least Absolute Shrinkage and Selection Operator) e confrontati i risultati tra loro. I metodi di involucro, come FeSC, determinano sempre il miglior sottoinsieme di funzionalità in base alla classificazione; i metodi di filtraggio, come LASSO, sono indipendenti dalla classificazione; mentre i metodi embedded, come pFSFS, combinano i precedenti. Per quanto riguarda lo studio di questa tesi le features selezionate sono state quelle provenienti da sensori EMG posizionati sulle braccia e da sensori EEG posizionati sulla zona motoria dell'emisfero sinistro. In particolare il primo esperimento ha preso come feature quella proveniente dal sensore AD C3 con frequenza di banda β, come suggerito dal paper [8], mentre il secondo esperimento ha tenuto conto di tutto il dataset, quindi di 32 EEG $\times$ 5 EMG $\times$ 11 bande di frequenza.

1.5 Over- sampling con SMOTE

L'algoritmo SMOTE (Synthetic Minority Oversampling Technique) è utilizzato per sovracampionare i dataset sbilanciati. Il suo principale vantaggio è la sua semplicità di utilizzo, grazie a questo esso comprende molte estensioni, per questo motivo SMOTE è l'algoritmo più comunemente usato. L'algoritmo SMOTE presenta però tre principali svantaggi: sovracampiona anche i campioni che contengono informazioni non rilevanti, sovracampiona anche i campioni affetti da rumore e non seleziona bene i vicini più prossimi per la scelta dei campioni sintetici. Esistono diverse versioni di questo algoritmo che migliorano la precisione e risolvono queste problematiche. Borderline-SMOTE [9], si concentra su campioni all'interno dell'area selezionata per rafforzare le informazioni sui confini della classe. RCSMOTE [10], si concentra sul range delle istanze sintetiche. K-means SMOTE [11], integra l'algoritmo di clustering insieme a SMOTE per gestire i campioni sovrapposti di classi differenti. G-SMOTE [12], genera campioni sintetici in una determinata regione geometrica. LNSMOTE [13], considera le informazioni sul quartiere locale dei campioni. DBSMOTE [14], usa l'algoritmo di scan DBSCAN che si basa sulla densità prima del campionamento. Gaussian-SMOTE [15], è una combinazione della distribuzione gaussiana e di SMOTE, basata sulla densità di probabilità del dataset. Tutte queste versioni dell'algoritmo SMOTE lo migliorano ma non estraggono o sfruttano a pieno le informazioni sulla distribuzione dei campioni, nè modificano la scelta casuale dei vicini più prossimi. Recentemente è stato sviluppato un nuovo algoritmo di oversampling: OS-CCD, che permette di migliorare significativamente l'algoritmo SMOTE con le relative estensioni. [16]

Qui di seguito viene spiegato il procedimento dell'algoritmo SMOTE, in particolare la versione K-means SMOTE. L'algoritmo k-means SMOTE, è un'estensione potenziata dell'algoritmo di cluster k-means combinato con il tradizionale algoritmo SMOTE con lo scopo di ricampionare le classi minoritarie. Il tradizionale SMOTE soffre di alcuni problemi: è meno efficace nella gestione di set di dati sbilanciati all'interno della classe; possono essere creati campioni di minoranze rumorose poichè le regioni di classe sovrapposte non sono ben distinte dalle regioni sicure; il confine decisionale è sfocato e i campioni lontani dal confine potrebbero essere sovracampionati. Per superare questa discordanza l'algoritmo si propone solo di ricampionare i dati insieme alle regioni attorno ai gruppi di campioni sparsi di minoranze. Questo metodo, non solo evita il rumore di confine ma considera anche lo sbilanciamento presente nelle classi. Il metodo usato consiste nei seguenti step:

1. Raggruppare i campioni minoritari con l'algoritmo k-means Clustering e calcolare la

distanza euclidea tra le coppie presenti nei cluster

2. calcolare la sparsità delle classi minoritarie per ogni cluster

$$sparsity(i) = \frac{\vec{d}^{m}}{N_{minor}} \tag{1.3}$$

con $\vec{d}$ che indica la media della distanza euclidea di ogni (i) cluster, m indica il numero delle features e N_{minor} il numero di campioni minoritari per cluster

3. Per ogni cluster si calcola il rapporto IR

$$IR = \frac{N_{major}}{N_{minor}} \tag{1.4}$$

con N_{major} che indica il numero di classi principali per ogni cluster, per evitare il caso in cui numeratore o denomitarore sono 0, si modifica il rapporto come segue

$$IR = \frac{N_{major} + 1}{N_{minor} + 1} \tag{1.5}$$

4. Dopo di chè si calcolano i pesi di campionamento in base alla scarsità dei campioni di minoranza in tutti i cluster come

$$\omega = \frac{\text{sparsity(i)}}{\sum_{i} \text{sparsity}(i)} \tag{1.6}$$

dove il peso viene quindi applicato per determinare il numero di campioni che dovrebbero essere generati in un cluster

5. Si adotta l'algoritmo SMOTE per generare i campioni sintetici $\tilde{\mathbf{x}}$ nei cluster dove $IR > 1$ con questa formula

$$\tilde{\mathbf{x}} = \mathbf{x} + \lambda \left(\mathbf{x}_{kn} - \mathbf{x}\right) \tag{1.7}$$

dove $\mathbf{x}$ è scelto casualmente tra i campioni minoritari e x_{kn} è un campione casualmente scelto dai K vicini più prossimi a x e λ è un numero casuale compreso tra $[0,1]$ [11].

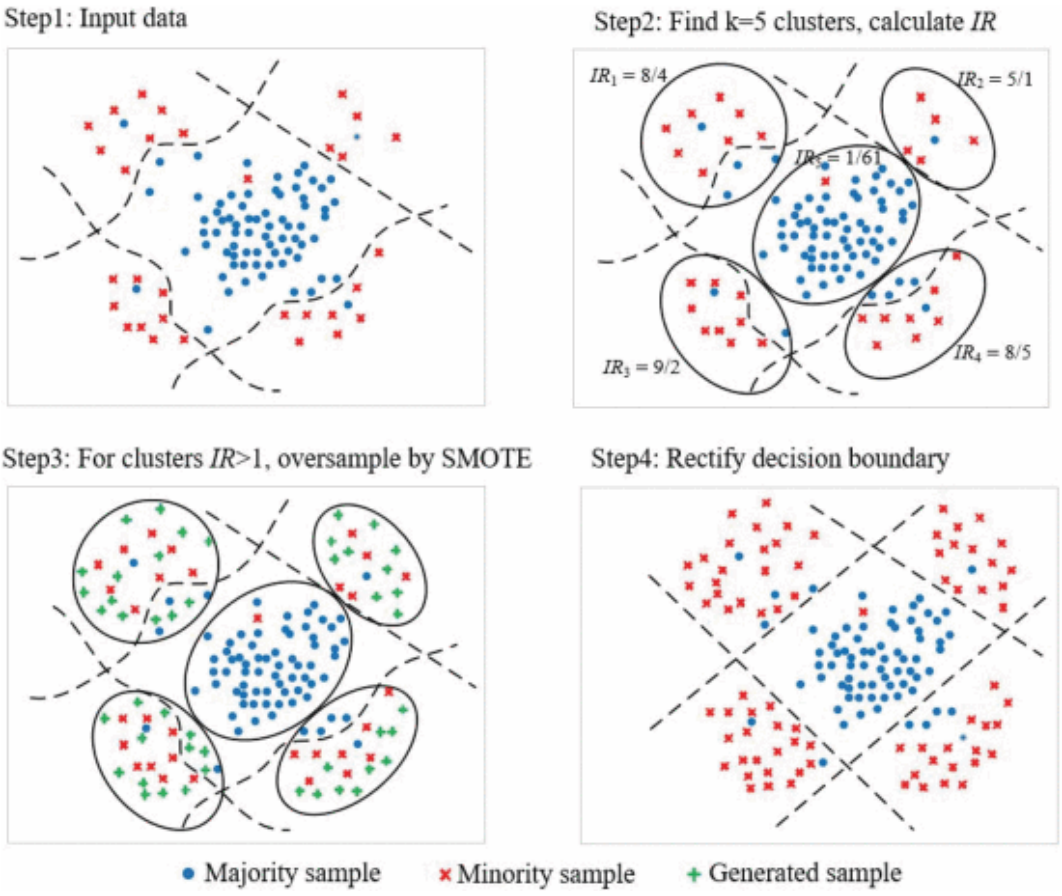

Figura 1.7: Processo dell'algoritmo K-means SMOTE

1.6 Algoritmo K-Means

Gli algoritmi di clustering sono dei modelli di apprendimento non supervisionato ampiamente sfruttati perchè permettono di dividere il dataset in gruppi, o cluster, utili per avere informazioni aggiuntive sui dati. Negli algoritmi di cluster si costruiscono i gruppi senza conoscere a quale classe appartiene ciascun campione. Di conseguenza, l'obiettivo è quello di raggruppare gli oggetti che sono più simili tra loro considerando l'intero insieme di feature o selezionandone alcune che si ritengono più importanti [17].
L'algoritmo di clustering K-Means [18], è considerato uno tra i più usati grazie alla sua semplicità. K-means è un algoritmo iterativo che prova a partizionare il dataset in 'k' predefiniti disinti e non sovrapposti sottogruppi (o cluster). K-Means è un algoritmo che crea Prototype-based cluster. Ovvero un cluster in cui tutti gli oggetti sono più simili al prototipo che definisce il gruppo. Solitamente il prototipo corrisponde al centroide, ossia la media di tutti i punti del cluster. Se si desidera creare k gruppi allora l'algoritmo sceglierà k centroidi iniziali e ogni altro punto sarà associato al centroide più vicino. A questo punto, si ricalcoleranno i centroidi sfruttando i k insiemi creati e si continuerà la procedura fino a che nessun centroide sarà diverso rispetto all'iterazione precedente oppure nessun punto cambierà di cluster. L'algoritmo si può quindi riassumere nel modo seguente:

1. si selezionano k punti che saranno considerati i centroidi iniziali
2. Si formano k gruppi assegnando ogni punto al centroide più vicino.
3. Si ricalcolano i centroidi facendo la media tra i campioni di ciascun cluster.
4. si ripetono gli step 2 e 3 finchè i centroidi non smettono di variare

Nella Figura 1.8 si vede un esempio in cui si fanno da sinistra a destra 4 iterazioni dell'algoritmo e si arriva così ad individuare tre gruppi distinti con i relativi centroidi. [19]

1.7 Stato dell'arte

Nonostante i progressi nel campo delle brain computer interfaces (BCI), l'uso del singolo segnale EEG per controllare dispositivi per l'assistenza medica, come fasi di riabilitazione, non è attualmente applicato a causa della sua inaffidabilità. Le interfacce ibride rappresentano una valida soluzione per arricchire le prestazioni. Questi metodi di classificazione che combinano interfacce uomo-macchina includono normalmente almeno una BCI e un altro segnale

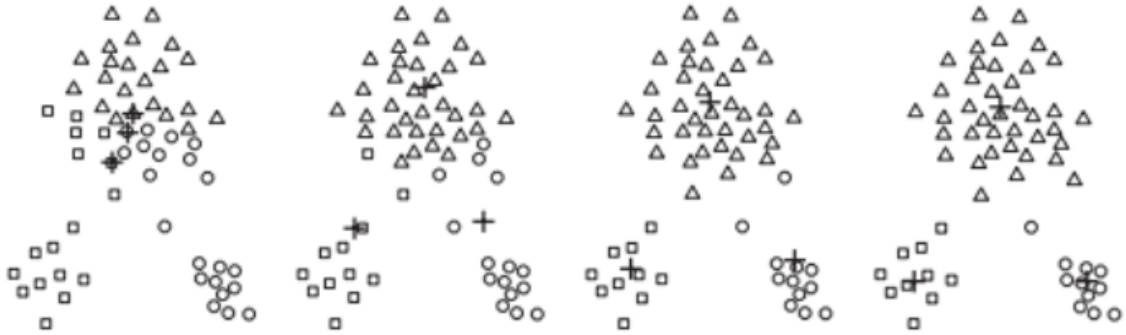

Figura 1.8: Processo di iterazione dell'algoritmo K-means

biomedico come ad esempio l'EMG. Stefano Tortora et. al. in [2] propongono un'interfaccia ibrida uomo-macchina per decodificare le fasi di deambulazione di entrambe le gambe dalla fusione di segnali EEG ed EMG.

Nei recenti studi si sono sviluppati sempre più metodi di fusione di segnali per interfacce ibride, in modo da arricchire e migliorare le prestazioni per le classificazioni. Un aspetto importante che non viene sempre considerato riguarda la presenza di dataset sbilanciati. Ji-Hoon Jeong et. al. in [20] ha fatto notare che c'è stato un lento progresso nel settore delle hybrid-BCI a causa di una mancanza di dataset ampi e uniformi; come soluzione ha fornito un ampio set di dati riguardanti 11 movimenti di arti superiori ottenuti in più sessioni di registrazioni.

Pertanto, si ritiene importante andare ad analizzare quali siano le tecniche di ribilanciamento e quanto efficaci risultino per ovviare al problema di dataset non uniformi. In questa sezione viene trattato il problema di avere dataset sbilanciati e vengono presentati alcuni dataset EMG-EEG disponibili pubblicamente.

1.7.1 Dataset sbilanciati

Molti problemi del mondo reale danno vita ad un numero disomogeneo di esempi appartenenti alle singole classi considerate. Tuttavia, si definisce dataset sbilanciato se c'è una significativa sproporzione del numero di esempi relativi a ciascuna classe. In altre parole, dunque, il dataset è sbilanciato quando una classe è rappresentata in numero molto minore rispetto alle altre [21].

Nel machine learning e in statistica, la classificazione con l'addestramento di un sistema con un set di dati etichettato per identificare un nuovo set di dati invisibile a cui appartenere. Di

recente, c'è stata un'enorme crescita dei dati e, sfortunatamente, mancano dati etichettati di qualità. Vari metodi di apprendimento automatico tradizionali presuppongono che le classi target abbiano la stessa distribuzione. Questa ipotesi non è sempre corretta, ad esempio nelle previsioni del meteo o nella diagnosi di malattie, poichè quasi la maggior parte delle istanze vengono etichettate come una classe maggioritaria, mentre le restanti istanze come quella minoritaria. Per questo motivo i modelli imparano di più dalle classi maggioritarie e non tengono conto di quelle minoritarie. Tutto questo si riflette sulle prestazioni dei modelli poichè questi funzionano male quando i dati sono sbilanciati. Questo viene chiamato problema di squilibrio delle classi. L'approcio più comune per superare questo problema è quello di usare delle tecniche di ricampionamento per bilanciare il dataset. Le tecniche per ricampionare si possono usare in entrambi i modi, ovvero sottocampionando o sovracampionando. Nel primo caso si riducono il numero di istanze maggioritarie in modo da eguagliare quelle minoritarie, mentre nel secondo caso si aumenta il numero di istanze minoritarie in modo da eguagliare quello delle maggioritarie. Un esempio è riportato in figura 1.9. Il secondo caso è quello che ha prestazioni migliori, perchè non vi è perdita di informazione e si aggiungono nuovi campioni chiamati sintetici, che vengono calcolati con vari metodi, ad esempio il metodo più comune è il metodo SMOTE. [16].

Figura 1.9: Differenza tra sottocampionamento e sovracampionamento.

1.8 Dataset EEG, EMG combinati

REF	TYPES OF SIGNAL	DATASET	ACCURACY
Uddin et al., 2020	ECG, accelerometer, gyroscope, magnetometer	MHEALTH dataset, PUC- Rio dataset, AReM dataset	-
Chambon et al., 2018	EEG, EMG, EOG	MASS dataset – session 3	-
Andreotti et al., 2018	EEG, EMG, EOG	SLPEDF-DB, MASS-DB, CAPSLP-DB, RDB	EEG+EOG: 0.68, 0.72, 0.62, 0.49. EEG+EOG+EMG: 0.67, 0.74, 0.61, 0.48.
Said et al., 2017 Al-Sa'D, et al., 2018	EEG, EMG EEG, EOG.	DEAP dataset.	78.1% dominance, 65.9% arousal
Kawde et al. 2017	EEG, EMG, EOG, GSR	DEAP dataset.	Valence 75.78%, Arousal 70.7%, Dominance 69.14%
Katsis et al., 2008	HR, BP, oxygen sat- uration, body tempera- ture, blood glucose, accelerometer, ECG, EEG	-	-
T. et al. 2020	EEG, EMG	Aquired dataset	Above 80% with fusion of EEG and EMG
Jeong j. Et al 2020	EEG, EMG, EOG	Multimodal signal dataset	-
Redunerick 2019	32 EEG, 4 EOG, 4 EMG, temperature, GSR, respiration, blood pressure, heart rate	bnci-horizon database	-
Xi 2020	EEG C3, EMG ECR, ECU, FD	Mendelay data	SVM original MSC 79.33% SVM enhanced MSC 83.50%
Luciw et al. 2014	EEG, EMG	WAY-EEG-GAL dataset	-

Nella tabella 1.8 sono riportati una lista di dataset che comprendono segnali EEG-EMG combinati disponibili pubblicamente e alcuni riferimenti di studi che hanno utilizzando tali dataset. Per alcuni è anche riportata la precisione dei dati e il metodo con cui è stata calcolata.

Metodi

2.1 Dataset WAY-EEG-GAL

Il dataset utilizzato per la tesi è WAY-EEG-GAL dataset (Wearable interfaces for hAnd function recoverY, Grasp And Lift) , disponibile online. Per ricavare questo dataset, è stato chiesto a 12 partecipanti di sollevare un oggetto, di cui venivano dinamicamente cambiati peso (165g, 330g, 660g) e superficie (carta vetrata, seta) ad ogni trial. L'EEG è stato registrato da 32 canali, mentre l'EMG da 5 muscoli fra mano e braccio. Contemporaneamente sono stati registrati anche la posizione dell'oggetto e la forza applicata da pollice e indice su di esso. Ogni soggetto doveva ripetere 34 volte la stessa operazione di sollevamento dell'oggetto (trial). Durante ogni sollevamento (lift), il soggetto era seduto al tavolo, con la spalla rilassata e il braccio vicino al busto. Durante il processo l'avambraccio era sollevato per tutto il tempo. Il braccio sinistro, invece, doveva rimanere vicino al busto. La luce rossa segnalava l'inizio del trial e il soggetto doveva allungarsi, predere e sollevare l'oggetto, il quale era mantenuto con il pollice e indice a metà della superficie laterale e sollevarlo fino a circa 5 cm dal tavolo. Quest'oggetto doveva essere portato dentro un cerchio e mantenuto nella stessa posizione fino a quando la luce rossa non si fosse spenta (dopo circa 2 secondi). A questo punto l'oggetto andava rimesso sul tavolo e il soggetto poteva ritornare nella posizione di partenza e riposare la spalla. Il peso di questo oggetto veniva cambiato tramite un meccanismo a magneti in modo inaspettato per il partecipante e ogni massa veniva utilizzata da 1 a 4 volte per poi essere cambiata con un'altra. Il cambiamento del peso e della superficie influiva sulla modulazione della coordinazione muscolare del soggetto in quanto un oggetto più pesante introduce un aumento sia nella forza utilizzata per sollevarlo, che in quella usata per stringerlo, mentre un oggetto più liscio introduce un aumento solo nella seconda. Il segnale EEG è stato acquisito ad una frequenza di 5 kHz e filtrato con un filtro passabanda 0.016-1000 Hz. Dopodichè è stato campionato a 500 Hz dall'amplificatore, il quale eseguiva anche un'operazione di filtro passa-basso per prevenire aliasing. L'EMG era invece campionato a 4 kHz, mentre tutti gli

altri segnali a 500 Hz. Il Dataset presenta un problema di sbilanciamento delle classi, nella figura 2.10 sono riportati i numeri di campioni per classe e soggetto. Rispettivamente la class 1 e la class2 nella tesi vengono definite così:

1. Task 1: indica la class 1, quindi il sollevamento di un oggetto ricoperto di carta vetrata
2. Task 2: indica la classe 2, quindi il sollevamento di un oggetto ricoperto di velluto

Subject Id.	SS	
	Class 1 (sandpaper)	Class 2 (silk)
P1	51	220
P4	39	210
P7	51	221
P11	50	221
P2	50	221
P3	51	220
P5	50	221
P9	51	220
Total	393	1754

Figura 2.10: Sbilanciamento delle classe per soggetto e task, [8]. SS indica il problema di classificazione dove l'oggetto da sollevare poteva essere ricoperto di carta vetrata (sandpaper) o velluto (silk).

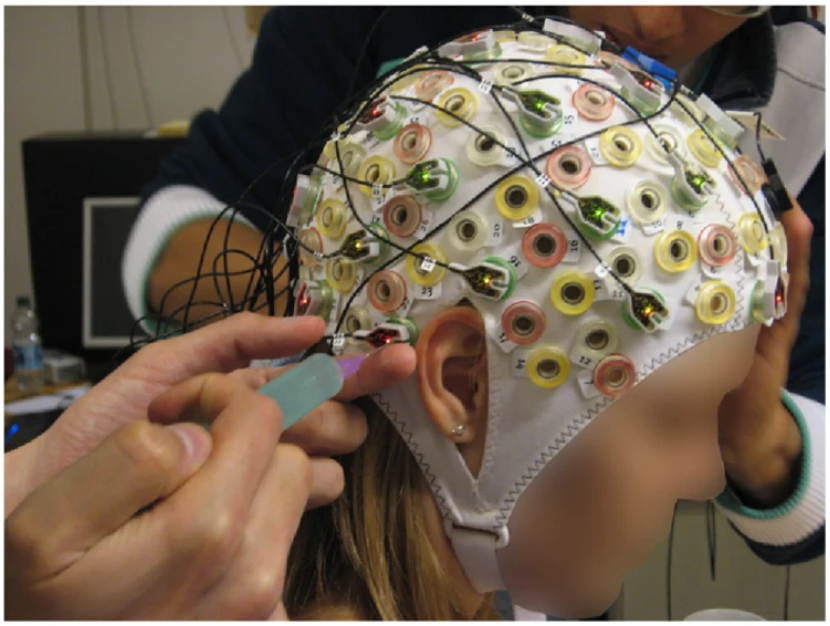

Figura 2.11: Sistema EEG a 32 elettrodi usato durante l'acquisizione dati (figura tratta da [22]).

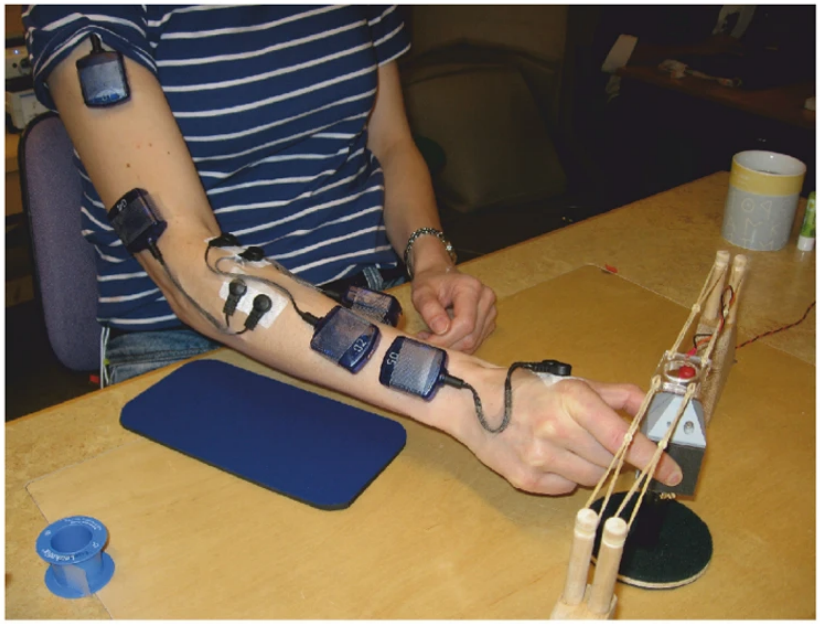

Figura 2.12: Posizionamento sensori EMG, ogni sensore viene collocato su un muscolo diverso del braccio (figura tratta da [22]).

Una descrizione più dettagliata del dataset è presente in [22]. Il dataset usato in questa tesi è un sottoinsieme del dataset appena descritto. In particolare è già stato fatto un primo lavoro di pre-processing: campionamento, fusione dei segnali EEG ed EMG ed estrazione delle features. Il campionamento ha portato a 3 diversi dataset:

1. no-sampling, il dataset è rimasto sbilanciato volutamente per poter confrontare i risultati con gli altri due dataset

2. over-sampling, è stato fatto un sovracampionamento utilizzando l'algoritmo SMOTE

3. down-sampling, sono stati tolti campioni dalla classe maggioritaria in modo casuale

Gli esperimenti sono stati fatti con lo scopo di controllare la variabilità dei cluster prima e dopo i due metodi di campionamento, in modo da capire chi tra i due porti a risultati migliori. Per ogni combinazione della tripletta EEG-EMG-banda di frequenza è stata estratta una feature MSC. Quindi ogni features (colonna della tabella) è la combinazione di queste 3 caratteristiche.

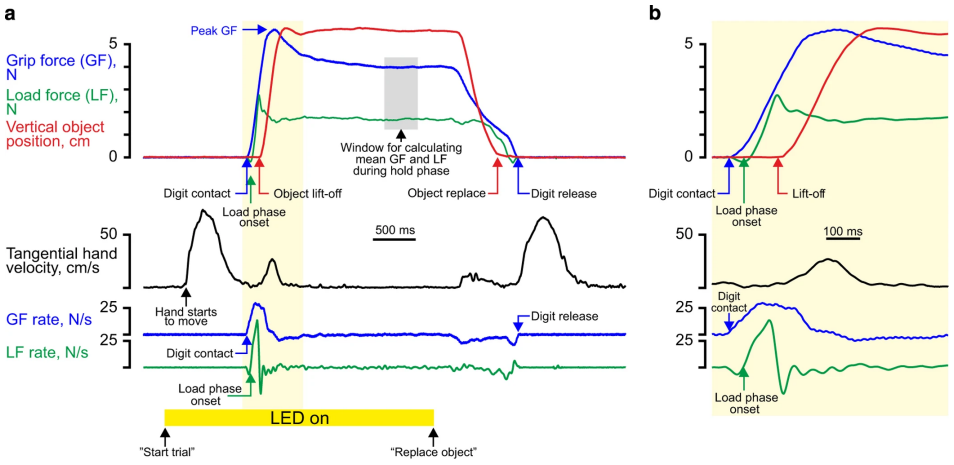

Figura 2.13: Nel grafico sono rappresentate le diverse fasi di un trial attraverso un'analisi delle forze di presa e di sollevamento, oltre alla posizione e velocità dell'oggetto. Delle frecce indicano i punti più importanti, fra cui l'accensione e lo spegnimento del LED [22]

2.2 Descrizione esperimenti

Sono stati fatti due esperimenti che si distinguono tra loro per il numero di features usate. Nel primo esperimento sono state usate solo le features corrispondenti al canale EEG C3, canale EMG AD e banda di frequenza β, mentre nel secondo sono state usate tutte le features disponibili, quindi 32 canali EEG, 5 canali EMG e 11 bande di frequenza, con un totale di 1760 features. I dati di applicazione dei due esperimenti vengono definiti nel seguente modo:

1. Esperimento 1: rappresenta le features AD C3 β
2. Esperimento 2: rappresenta l'intero dataset con tutte le bande di frequenza

Per ogni esperimento sono stati eseguiti tre test diversi, per confrontare la variabilità tra coppie di cluster. Sono stati fatti i seguenti confronti per valutare la variazione con over-sampling e down-sampling:

1. Test 1: per un paio di soggetti, task 1 vs task 2
2. Test 2: tutti i samples di un soggetto vs un altro soggetto
3. Test 3: task 1 prer tutti i soggetti vs task 2 per tutti i soggetti

Per eseguire questi test, si sono dovuti distinguere i casi di no-sampling, over-sampling e down-sampling. Nell'over-sampling si è modificata la matice iniziale, per ogni soggetto e task sono stati aggiunti campioni, con l'algoritmo SMOTE, in modo tale da avere tutte le classi con lo stesso numero di campioni, che corrisponde al massimo. Il caso down-sampling ha fatto il contrario: sono stati tolti campioni in modo casuale da ogni classe in modo da avere tutte le classi con lo stesso numero di campioni, che corrisponde al minimo. In base al tipo di campionamento si è quindi selezionata una delle 3 matrici. Il secondo step è quello di identificare nella matrice delle features quali estrarre: per l'esperimento 1 sono state estratte le due features (colonne), di interesse mentre per l'esperimento 2 si è tenuto conto dell'intero dataset. Una volta ottenute le colonne desiderate, sono state selezionate le righe che corrispondevano al soggetto della task in funzione del test. A questo punto si ha a disposizione la sottomatrice di interesse per esperimento e test. Questa matrice è stata divisa ulteriormente in due sottomatrici rappresentanti rispettivamente le due classi da confrontare $cl1$ e $cl2$. Per l' esperimento 1 $cl1$ e $cl2$ sono matrici bidimensionali, pertanto è stata data una rappresentazione grafica utile per il confronto visivo dopo la clusterizzazione. Dopodichè è stato applicato l'algoritmo K-Means con i seguenti comandi:

1. *k = 2;*
2. *c = [cl1;cl2];*
3. *opts = statset('Display','final');*
4. *[idx, c, sumd] = kmeans(c, k, 'Distance', 'cityblock', ...;*
5. *'Replicates',5,'Options',opts);*

Gli input sono importanti:

1. *c* = matrice [*n* x *m*] che contiene *n* punti in uno spazio m-dimensionale
2. *k* = numero di cluster che si vogliono ottenere, per entrambi gli esperimenti *k* = 2

Gli utput restituiti invece:

1. *idx* = è un vettore [*n* x 1] dove ogni riga indica rispettivamente a quale cluster è stata collocata
2. *C* = matrice [*k* x *m*], indica in uno spazio m-dimensionale la posizione dei *k* centroidi
3. *sumd* = vettore [*k* x 1] che indica la somma quadratica punto-centroide

Da questi 3 output sono stati ricavati la distanza tra centroidi, e la size dei due cluster.

2.3 Implementazione Matlab

Tutto questo è stato implementato in Matlab utilizzando il dataset WAY-EEG-GAL già processato con le features MSC estratte e preparate in tre matice di dimensioni [2147x1760] per no-sampling, [3536x1760] per over-sampling e [624x1760] per down-sampling. I metodi di estrazione della features MSC sono disponibili pubblicamente in [23]. Per poter usare l'algoritmo di clusterizzazione k-means è stato aggiunto il toolbox di Statistica e machine learning. Per effetture i test in modo più pratico, sono state scritte due funzioni, *normalizzazione(...)* e *confronto(...)*. La funzione *normalizzazione(...)* riceve come input una delle 3 matrici iniziali e due vettori per task di offset che contengono il numero di segmenti per soggetto. Per il dataset no-sampling i due vettori sono le prime due colonne della tabella 2.10:
$v\_offset\_task1 = [51, 39, 51, 50, 50, 51, 50, 51]$
$v\_offset\_task2 = [220, 210, 221, 221, 221, 220, 221, 219]$

Per l'over-sampling invece sono due vettori uguali contenenti lo stesso valore, che indica il valore massimo di campioni:

$v\_offset\_task1 = [221, 221, 221, 221, 221, 221, 221, 221]$

$v\_offset\_task2 = [221, 221, 221, 221, 221, 221, 221, 221]$

Per l'down-sampling il ragionamento è analogo al caso precedente, ma i valori sono tutti al numero minimo di campioni:

$v\_offset\_task1 = [39, 39, 39, 39, 39, 39, 39, 39]$

$v\_offset\_task2 = [39, 39, 39, 39, 39, 39, 39, 39]$

La funzione continua poi creando una struttura *c* che contiene i campioni divisi per soggetto. In questo modo è in grado di calcolare media *mean(c)* e deviazione standard *std(c)* solo per soggetto. Infine, per ogni record il dato viene normalizzato con la seguente formula:

$$c\_i\_new = \frac{c\_i - mean(c)}{std(c)} \tag{2.8}$$

Dalla struttura con i dati normalizzati per soggeto si ricostruisce una matrice simile a quella iniziale. La funzione *confronto(...)* invece riceve come imput la due sottomatrici $cl1$ e $cl2$ da confrontare ed esegue l'algoritmo k-means, con i dati ottenuti calcola la distanza eculidea tra centroidi e le size dei due cluster trovati. Per l'esperimento 1 questa funzione disegna anche su un piano cartesiano la mappa dei punti prima e dopo il clustering, in modo da avere un confronto visivo.

Risultati e discussione

3.1 Analisi dei dati

In questa sezione vengono analizzati i risultati dell'implementazione Matlab come descritto nel capitolo precedente. Inizialmente è stato effettuato l'esperimento 1 in modo da avere anche un riscontro visivo nel piano cartesiano oltre ad uno numerico. Per iniziare si sono effettuati i test 1,2 e 3. I dati sono riportati nella tabella 3.2. Inoltre per il test 2 e 3 è stato fatto un ulteriore confronto con i campioni normalizzati solo per soggetto, tabella 3.4. Nel test 1 è riportata una media dei confronti tra i risultati ottenuti con i soggetti P1, P4, P7 e P11 per le tabelle 3.2 e 3.6. Nelle figure 3.14, 3.15 e 3.16 sono mostrati 3 grafici di 3 coppie di features diverse tra loro. Queste figure servono per dimostrare che le due classi possono essere maggiormente distinguibili se si scelgono diverse features.

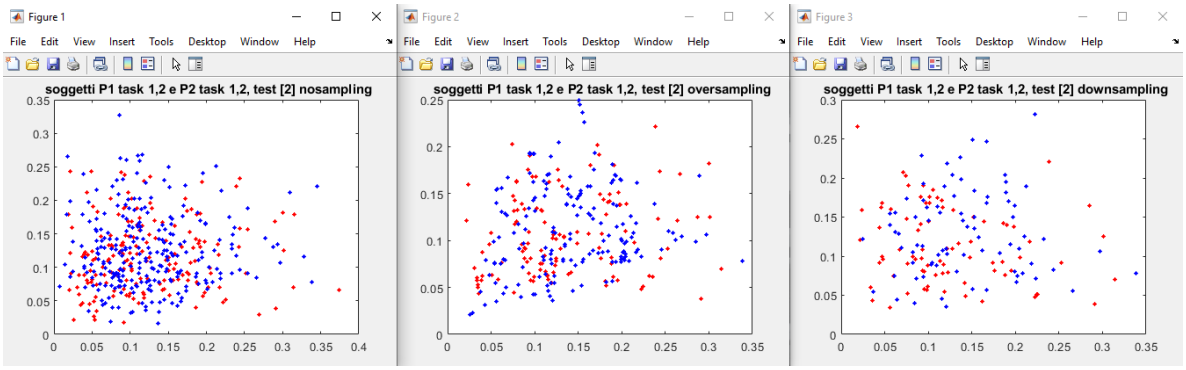

Figura 3.14: Esperimento 1, Test 2, soggetto P1, task 1 vs task 2, no-sampling, over-sampling e down-sampling non normalizzato, features AD C3 β1 e AD C3 β2.

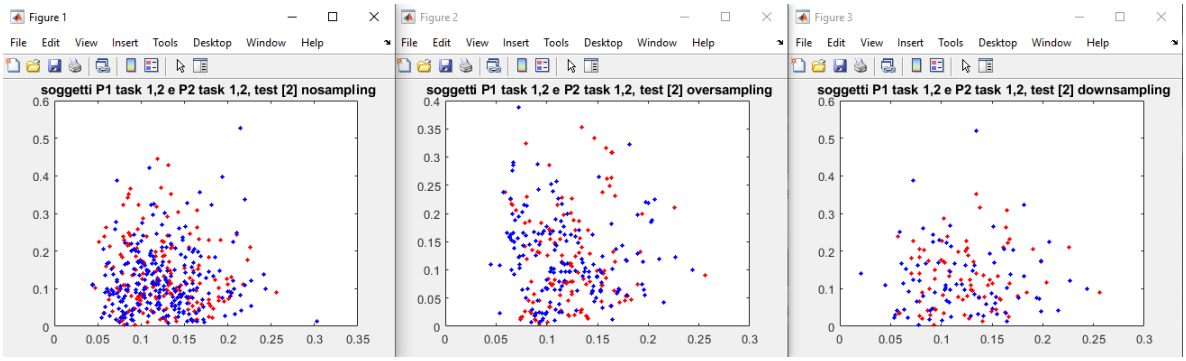

Figura 3.15: Esperimento 1, Test 2, soggetto P1, task 1 vs task 2, no-sampling, over-sampling e down-sampling non normalizzato, features AD Fp1 γ e BR FC6 θ.

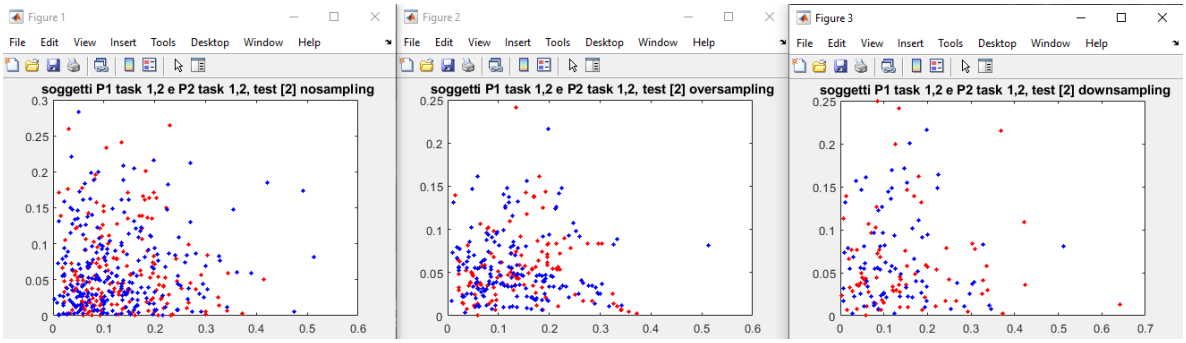

Figura 3.16: Esperimento 1, Test 2, soggetto P1, task 1 vs task 2, no-sampling, over-sampling e down-sampling non normalizzato, features FDI F7 δ e CED FZ α.

Tabella 3.1: Esperimento n°1 (utilizzate 2 features)

	TEST 1 (media)			TEST 2			TEST 3		
	dist	size 1	size 2	dist	size 1	size 2	dist	size 1	size 2
no	0,09952	0,03610	0,03837	0,1015	0,0353	0,039	0,0915	0,0401	0,0343
over	0,4950	0,03200	0,03240	0,1051	0,0336	0,0361	0,0874	0,0334	0,0376
down	0,5327	0,03560	0,03700	0,1053	0,0345	0,0373	0,0941	0,034	0,0386
				DATI NORMALIZZATI					
no	0,3857	0,1847	0,165	0,394	0,1774	0,1897	0,3887	0,1835	0,1647
over	0,3656	0,1577	0,1588	0,4402	0,1769	0,1692	0,3733	0,173	0,1634
down	0,3744	0,191	0,1724	0,04895	0,2063	0,1774	0,3809	0,167	0,1897

Tabella 3.2: Le righe indicano il dataset usato. Le sottocolonne rappresentano in ordine, la distanza tra centroidi e le dimensioni dei cluster.

Tabella 3.3: Tabella 3.2: Esperimento 1, Test 2 e 3 con normalizzazione per soggetto

	TEST 2			TEST 3		
	dist	size 1	size 2	dist	size 1	size 2
no	1,7606	0,6088	0,6713	1,5782	0,59	0,6907
over	1,88	0,645	0,6018	1,564	0,6718	0,5974
down	1,7553	0,6481	0,5712	1,5683	0,5651	0,6525

Tabella 3.4: Le righe indicano il dataset usato. Le sottocolonne rappresentano in ordine, la distanza tra centroidi e le dimensioni dei cluster.

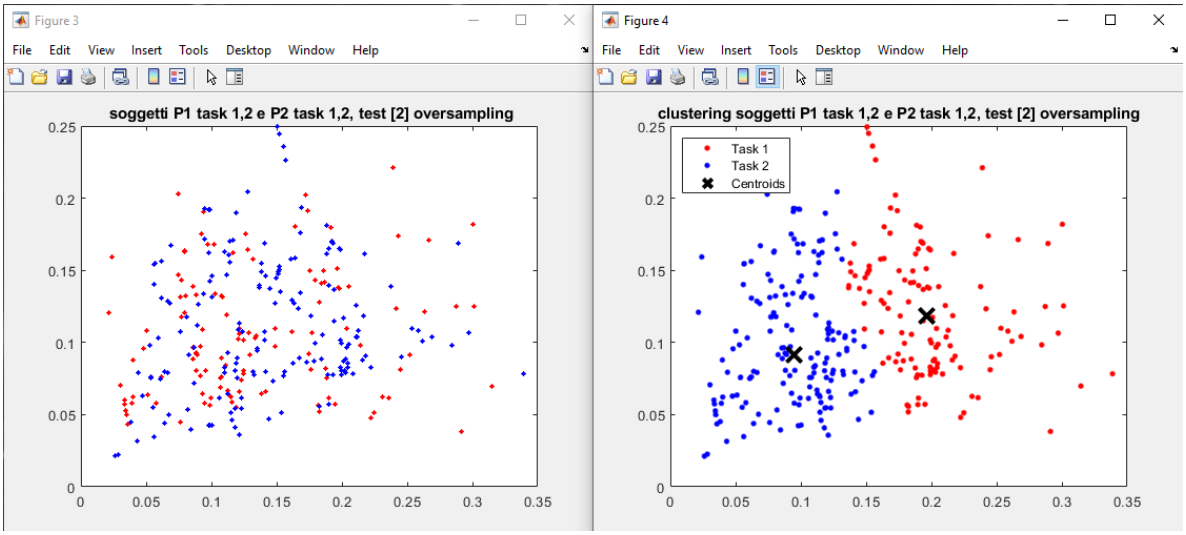

Figura 3.17: Esperimento 1, Test 2 , over-sampling, non normalizzato, features AD C3 $\beta 1$ e AD C3 $\beta 2$. La distanza tra i centroidi dopo l'over-sampling è di 0,1051

Da questo esperimento non si riescono a trarre conlcusioni definitive, a causa del fatto che sono state usate solo due features. L'unica cosa che però salta all'occhio guardando le tabelle 3.2 e 3.4 è come la normalizzazione per soggetto rispetto alla normalizzazione per soggetto e task aumenti considerevolmente la distanza tra centroidi. Si può inoltre notare visivamente che la clusterizzazione non ha prodotto risultati precisi, questo motivo può essere spiegato perchè contrariamente a quello che di solito si fa, qui è stata estratta un'unica tipologia di feature. Ciò significa che non è verosimile che alcune features abbiano ampiezze molto diverse dalle altre (cosa che di solito giustifica l'uso di normalizzazione per le features matrix prima della classificazione). Qui di seguito vengono riportate le immagini del test 2 in base alla normalizzazione. Le figure di sinistra rappresentano la distribuzione dei punti prima che venga applicato l'algoritmo k-means, mentre a quelle di destra è stato applicato.

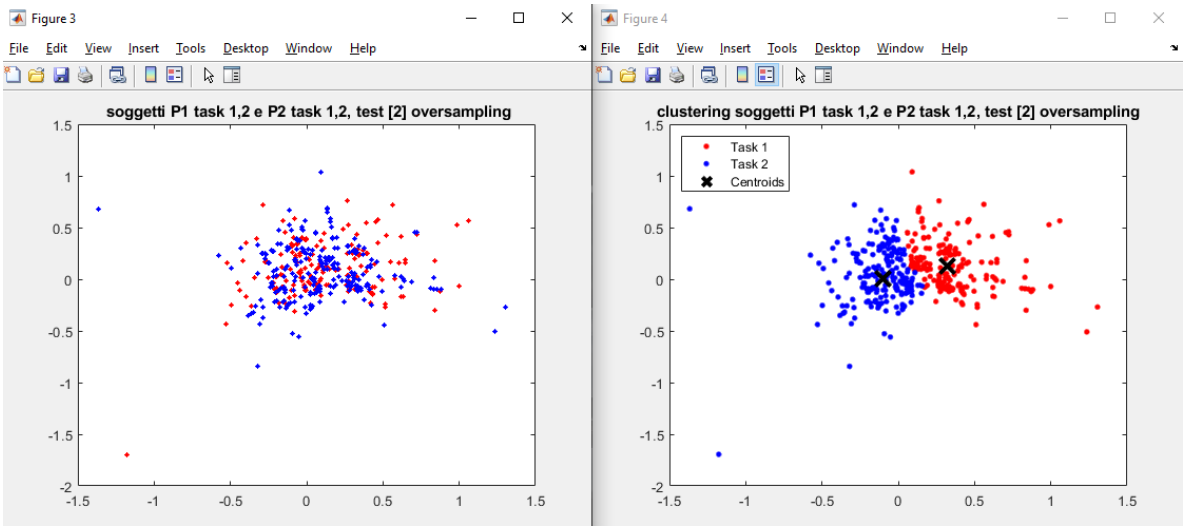

Figura 3.18: Esperimento 1, Test 2 , over-sampling, normalizzato per soggetto e task, features AD C3 β1 e AD C3 β2. La distanza tra i centroidi dopo l'over-sampling è di 0,4402

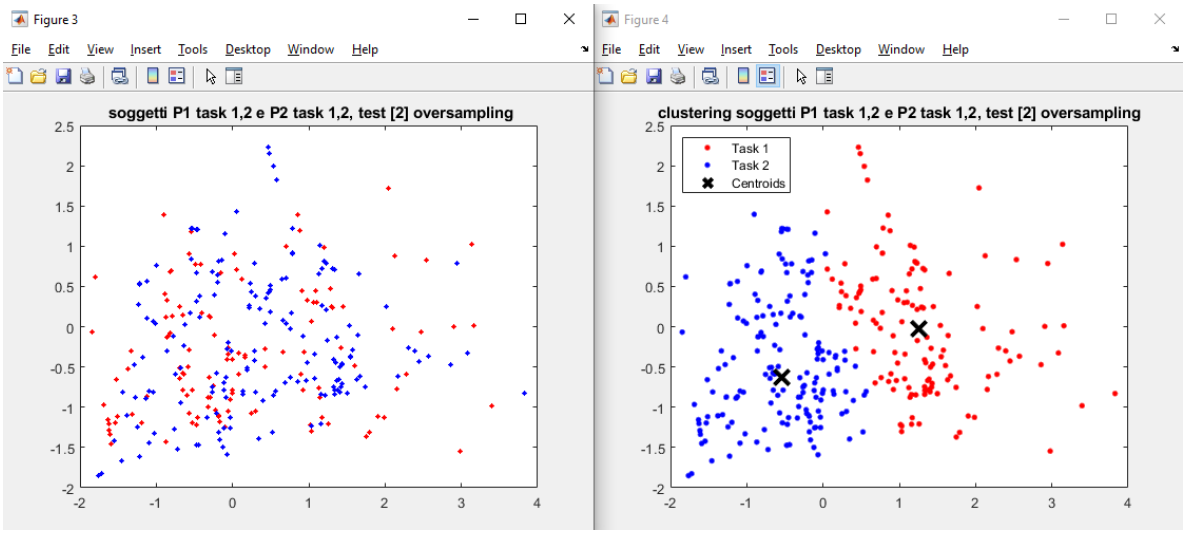

Figura 3.19: Esperimento 1, Test 2 , over-sampling, normalizzato solo per soggetto, features AD C3 β1 e AD C3 β2. La distanza tra i centroidi dopo l'over-sampling è di 1,8800

Per l'Esperimento 2, sono stati raccolti gli stessi dati.

Tabella 3.5: Esperimento 2

	TEST 1 (media)			TEST 2			TEST 3		
	dist	size 1	size 2	dist	size 1	size 2	dist	size 1	size 2
no	0,5667	36,587	34,9881	0,6062	36,128	35,026	0,5493	36,709	34,681
over	0,7761	31,779	31,6241	1,3523	27,259	32,943	0,5411	33,077	34,888
down	0,6704	36,395	35,3881	0,7583	36,702	32,706	0,6198	36,153	35,399
	DATI NORMALIZZATI								
no	0,597	36,8956	35,2528	0,59993	36,501	34,803	0,5774	36,521	35,2167
over	0,76	31,682	32,4987	0,7939	32,711	32,665	0,566	33,140	35,023
down	0,601	35,599	36,2238	0,7151	36,398	35,917	0,671	36,487	35,557

Tabella 3.6: Le righe indicano il dataset usato. Le sottocolonne rappresentano in ordine, la distanza tra centroidi e le dimensioni dei cluster.

Tabella 3.7: Esperimento 1, Test 2 e 3 con normalizzazione per soggetto

	TEST 2			TEST 3		
	dist	size 1	size 2	dist	size 1	size 2
no	1,1496	36,456	35,526	1,2115	35,3959	37,008
over	1,3673	33,86	31,089	1,3862	34,9789	33,7867
down	1,037	36,214	37,219	1,7314	36,9953	35,9411

Tabella 3.8: Le righe indicano il dataset usato. Le sottocolonne rappresentano in ordine, la distanza tra centroidi e le dimensioni dei cluster.

Anche l'esperimento 2 conferma che una normalizzazione per soggetto aumenta la distanza tra centroidi. Inoltre usando un numero maggiori di features gli effeti del sampling si riescono a notare, in particolare:

1. per il no-sampling si nota che la misura della distanza tra centroidi è sempre minore e le dimensioni dei cluster sono maggiori sempre rispetto ai casi di over-sampling e down-samping.

2. per l'over-sampling la distanza tra centroidi aumenta nella maggior parte dei casi e sembra farlo in modo significativo nel secondo esperimento, in particolare nel secondo test, inoltre le size dei cluster diminuiscono e sempre in modo significativo nel secondo esperimento test 2 tra soggetti diversi

3. per il down-sampling invece la distanza tra centroidi in genere aumenta ma non considerevolmente, mentre la dimensione dei cluster non sempre diminuisce e nei casi in cui diminuisce non c'è una grande variazione.

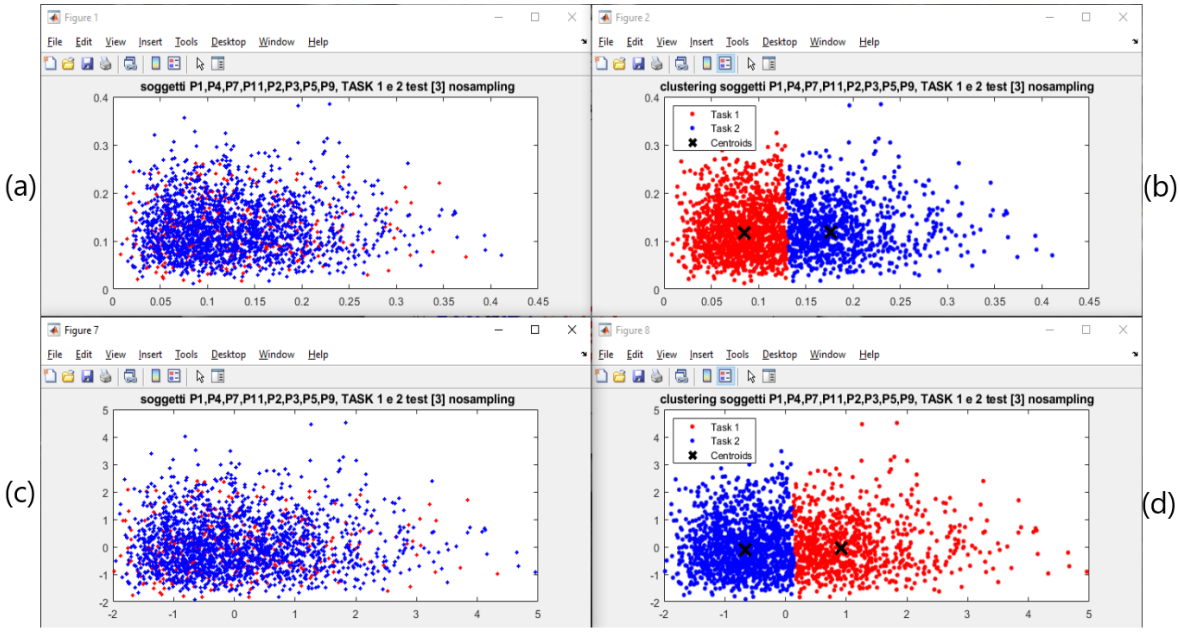

Figura 3.20: Esperimento 1, test 3, features AD C3 $\beta 1$ e AD C3 $\beta 2$. Le figure (c) e (d) rappresentano la versione normalizzata per soggetto di (a) e (b). Al dataset non è stato applicato nessun algoritmo di campionamento

I dati che sono stati trovati sono abbastanza in linea con quelli aspettati. L' over-sampling ha portato notevoli miglioramenti nella classificazione tra soggetto e in particolare nell'esperimento 2, quando sono state considerate un numero elevato di features. In oltre questo può probabilmente spiegare gli ottimi risultati in classificazione ottenuti da tutti gli algoriti di classificazione usati in [8] sullo stesso dataset. Un altro risultato interessante è che la normalizzazione ha separato maggiormente i centroidi, in particolare la normalizzazione per soggetto; si deduce quindi che c'è una variabilità maggiore tra soggetti. A seguire vengono presentate delle figure ottenute con l'esperimento 1, test 3 e con la normalizzazione per soggetto. Le figure di sinistra rappresentano la distribuzione dei punti prima che venga applicato l'algoritmo k-means, mentre a quelle di destra è stato applicato. Da queste figure si può notare come la clusterizzazione non sia ottime e la distanza tra centroidi aumenti con la normalizzazione per soggetto.

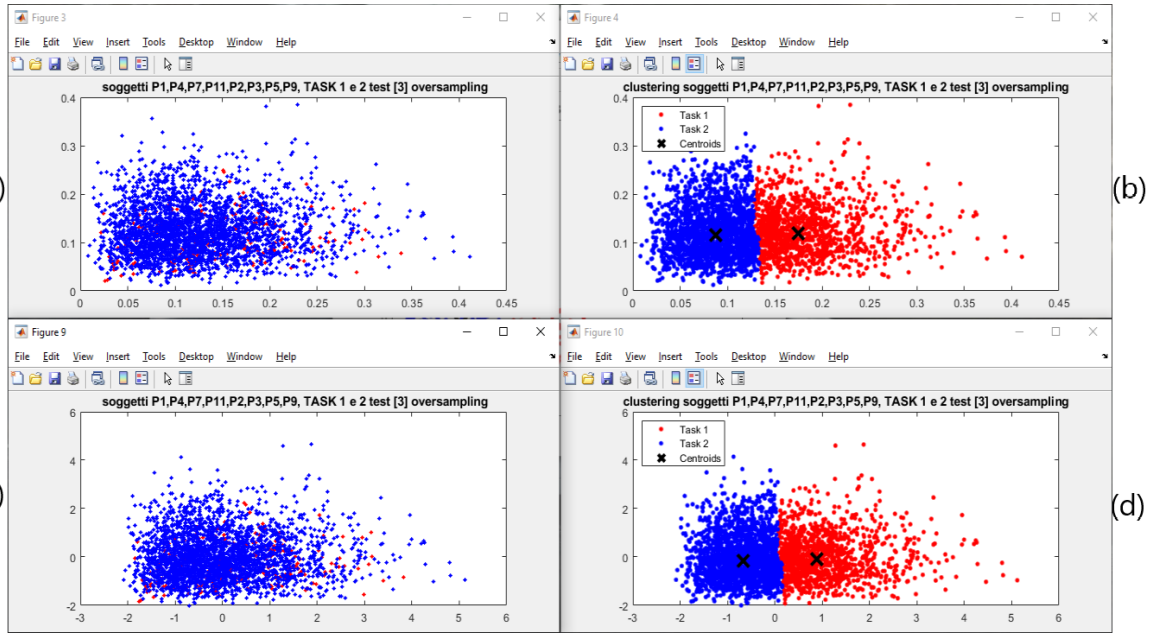

Figura 3.21: Esperimento 1, test 3, features AD C3 $\beta 1$ e AD C3 $\beta 2$. Le figure (c) e (d) rappresentano la versione normalizzata per soggetto di (a) e (b). Al dataset è stato applicato l'algoritmo di campionamento SMOTE

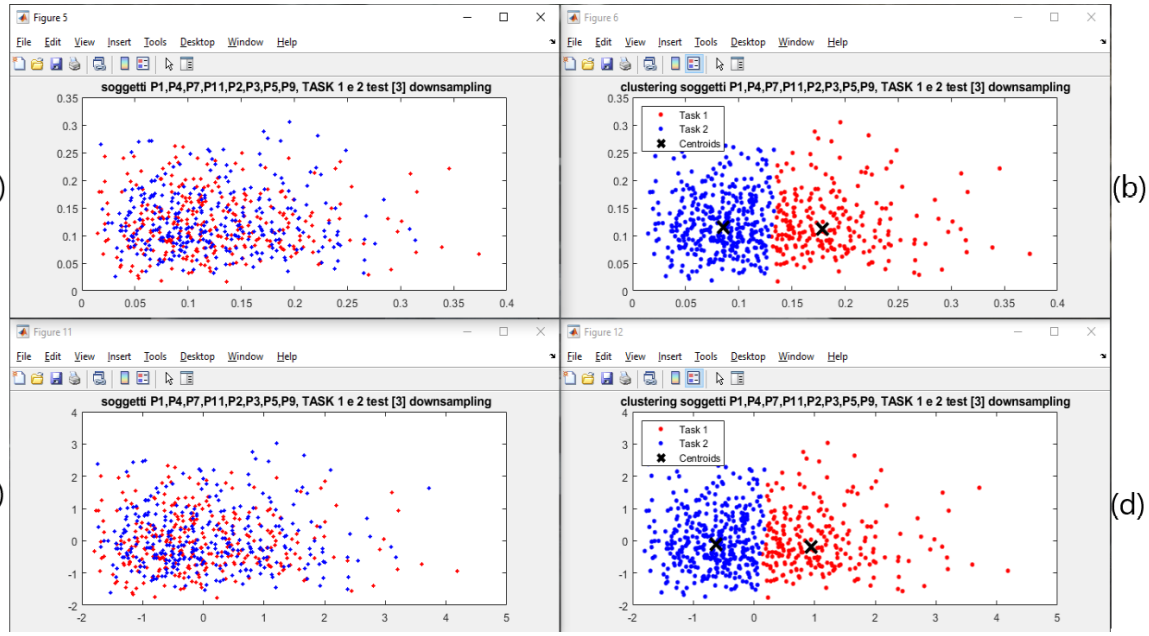

Figura 3.22: Esperimento 1, test 3, features AD C3 $\beta 1$ e AD C3 $\beta 2$. Le figure (c) e (d) rappresentano la versione normalizzata per soggetto di (a) e (b). Il dataset è stato sotto campionato in modo casuale

Conclusioni

Questa tesi tramite due esperimeti e diversi test, ha voluto dimostrare come i metodi di campionamento effettuati su dataset sbilanciati possono portare ad una miglior classificazione dei task.

Una considerazione da fare riguarda la clusterizzazione; si è notato che c'è poca variabilità intra-soggetto. Questo può essere dovuto al fatto che sono state considerate solo le features MSC.

Ad ogni modo, l'obbiettivo della tesi è quello di analizzare le due tipologie principali di campionamento e valutarne gli effetti sulla clusterizzazione. In particolare si è notato che applicando l'oversampling con l'algoritmo SMOTE si è confermata l'ipotesi iniziale, ovvero che il sovracampionamento è in grado di aumentare le prestazioni degli algoritmi di clustering aumentando la distanza tra centroidi e diminuiendo la size del cluster.

Un altro dettaglio interessante che è emerso è la normalizzazione per soggetto. Questo tipo di normalizzazione ha aumentato la distanza tra centroidi andando a separare in modo più marcato i cluster.

Un importante studio futuro è sicuramente quello di creare diversi dataset ottenuti tramite downsampling e oversampling e vedere quanto cambino le misure del clustering. Ulteriori approfondimenti di questa ricerca potrebbero interessante anche metodi di sovracampionamento alternativi, come ad esempio OS-CCD [16], oppure usare metodi ibridi, che combinino il sottocampionamento e il sovracampionamento.

Bibliografia

[1] Robert Leeb, Hesam Sagha, Ricardo Chavarriaga, and José del R Millán. A hybrid brain–computer interface based on the fusion of electroencephalographic and electromyographic activities. *Journal of neural engineering*, 8(2), 2011.

[2] Stefano Tortora, Luca Tonin, Carmelo Chisari, Silvestro Micera, Emanuele Menegatti, and Fiorenzo Artoni. Hybrid human-machine interface for gait decoding through bayesian fusion of EEG and EMG classifiers. *Frontiers in Neurorobotics*, page 89, 2020.

[3] Kenneth S Saladin, Raffaele De Caro, Giovanna Albertin, and Ciro Dalla Rosa. *Anatomia umana*. Piccin, 2012.

[4] J. A. Chambers S. Sanei. EEG signal processing,. pages 1–20, 2007.

[5] M.S. Hussain M. B. I. Reaz and F. Mohd-Yasin. Techniques of emg signals analysis: detection, processing, classifications and applications. pages 1–14, 2006.

[6] Thilina Dulantha Lalitharatne, Kenbu Teramoto, Yoshiaki Hayashi, and Kazuo Kiguchi. Towards hybrid EEG-EMG-based control approaches to be used in bio-robotics applications: Current status, challenges and future directions. *Paladyn, Journal of Behavioral Robotics*, 4(2):147–154, 2013.

[7] ShouYan Wang and MengXing Tang. Exact confidence interval for magnitude-squared coherence estimates. *IEEE signal processing letters*, 11(3):326–329, 2004.

[8] Giulia Cisotto, Martina Capuzzo, Anna Valeria Guglielmi, and Andrea Zanella. Feature stability and setup minimization for EEG-EMG-Enabled monitoring systems. (in revisione).

[9] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005.

[10] Paria Soltanzadeh and Mahdi Hashemzadeh. Rcsmote: Range-controlled synthetic minority over-sampling technique for handling the class imbalance problem. *Information Sciences*, 542:92–111, 2021.

[11] Dinghan Hu, Jiuwen Cao, Xiaoping Lai, Junbiao Liu, Shuang Wang, and Yao Ding. Epileptic signal classification based on synthetic minority oversampling and blending algorithm. *IEEE Transactions on Cognitive and Developmental Systems*, 13(2):368–382, 2021.

[12] Ke Cheng, Chen Zhang, Hualong Yu, Xibei Yang, Haitao Zou, and Shang Gao. Grouped smote with noise filtering mechanism for classifying imbalanced data. *IEEE Access*, 7:170668–170681, 2019.

[13] Tomasz Maciejewski and Jerzy Stefanowski. Local neighbourhood extension of smote for mining imbalanced data. In *2011 IEEE symposium on computational intelligence and data mining (CIDM)*, pages 104–111. IEEE, 2011.

[14] Chumphol Bunkhumpornpat, Krung Sinapiromsaran, and Chidchanok Lursinsap. Db-smote: density-based synthetic minority over-sampling technique. *Applied Intelligence*, 36(3):664–684, 2012.

[15] Hansoo Lee, Jonggeun Kim, and Sungshin Kim. Gaussian-based smote algorithm for solving skewed class distributions. *International Journal of Fuzzy Logic and Intelligent Systems*, 17(4):229–234, 2017.

[16] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. Machine learning with oversampling and undersampling techniques: overview study and experimental results. In *2020 11th international conference on information and communication systems (ICICS)*, pages 243–248. IEEE, 2020.

[17] Kristina P Sinaga and Miin-Shen Yang. Unsupervised k-means clustering algorithm. *IEEE access*, 8:80716–80727, 2020.

[18] David Pollard. Quantization and the method of k-means. *IEEE Transactions on Information theory*, 28(2):199–205, 1982.

[19] Imad Dabbura. K-means clustering: Algorithm, applications, evaluation methods, and drawbacks.

[20] Ji-Hoon Jeong, Jeong-Hyun Cho, Kyung-Hwan Shim, Byoung-Hee Kwon, Byeong-Hoo Lee, Do-Yeun Lee, Dae-Hyeok Lee, and Seong-Whan Lee. Multimodal signal dataset for 11 intuitive movement tasks from single upper extremity during multiple recording sessions. *GigaScience*, 9(10):giaa098, 2020.

[21] Massimiliano Morrelli. Addestramento con dataset sbilanciati. *arXiv preprint arXiv:2008.09209*, 2020.

[22] Matthew D Luciw, Ewa Jarocka, and Benoni B Edin. Multi-channel eeg recordings during 3,936 grasp and lift trials with varying weight and friction. *Scientific data*, 1(1):1–11, 2014.

[23] G. Cisotto. github.com/cisottogiulia/eeg-emg-analytics eeg-emg-analytics. 2021.