UNIVERSITÀ DI PISA

Dipartimento di Ingegneria dell'Informazione

Sviluppo di una rete neurale per il riconoscimento di valence e arousal

Gerardo Alvaro
Leonardo Fontanelli
Francesco Fornaini
Riccardo Polini

A.Y. 2018-2019

Indice

1 Introduzione

Lo scopo di questo progetto è quello di progettare e sviluppare un sistema intelligente in grado di capire lo stato d'animo di una persona sulla base di segnali biomedici ottenuti mediante appositi sensori. Un'emozione è caratterizzata da due fattori: *valence* e *arousal*. Il primo corrisponde all'affettività che può essere positiva o negativa, mentre il secondo misura quanto uno stato d'animo sia eccitante o calmante.

Possiamo quindi riconoscere nello spazio bidimensionale *valence-arousal* gli stati d'animo riportati in figura 1, caratterizzati da una coppia di valori *(v, a)* ognuno compreso tra 1 e 9.

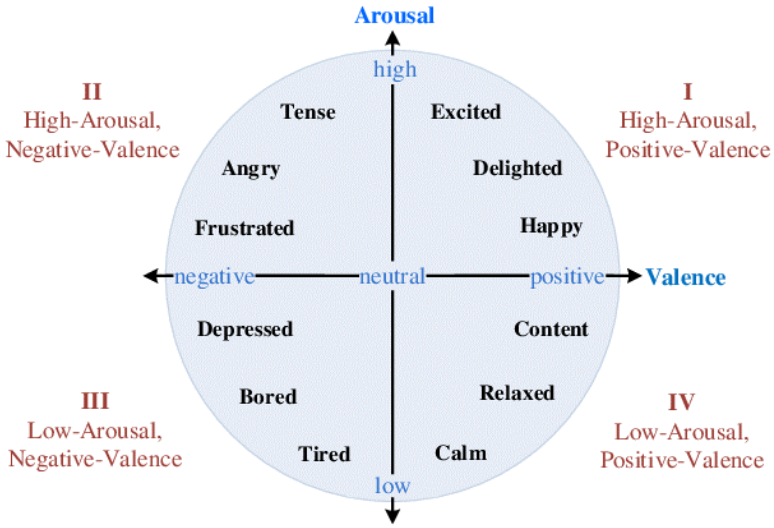


Figura 1: Spazio bidimensionale valence-arousal

2 Data set

Il dataset che avevamo a disposizione era composto da 32 partecipanti che hanno collaborato all'esperimento, i cui segnali biomedici sono stati costantemente monitorati mentre guardavano una serie di 40 video musicali della durata di un minuto a testa. L'emozione provata dal partecipante durante ogni video viene calcolato in termine di *valence* e *arousal* grazie all'utilizzo di *Self-Assessment Manikin*, mostrato in figura 2, una tecnica di valutazione che rende facile esprimere il livello dei due fattori in base allo stato d'animo della persona

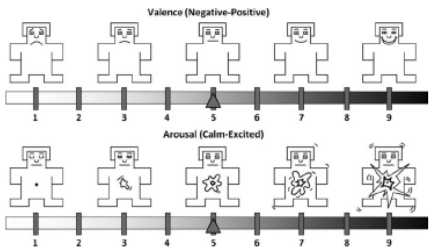


Figura 2: Self-Assesment Manikin

I segnali registrati da ogni sensore sono riportati nella tabella 1, le prime 32 righe, per esempio, rappresentano i segnali di un elettroencefalogramma ognuno proveniente da un elettrodo posizionato in una parte diversa della testa del partecipante. Per ogni persona ci sono state fornite due tabelle, '*Data*' e '*Labels*', composte dai campi riportati nella tabella 2; la prima tabella è tridimensionale ed è composta dall'ID del video, dall'ID del sensore e infine dagli 8064 campioni che formano il segnale GSR; la seconda invece è composta dall'ID del video e da 4 fattori: *valence*, *arousal*, *dominance* e *liking*; si prega di notare che per il nostro scopo la terza e la quarta colonna di questa tabella possono essere ignorate.

Sensor_ID	Signal
1	Fp1
2	AF3
3	F3
4	F7
5	FC5
6	FC1
7	C3
8	T7
9	CP5
10	CP1
11	P3
12	P7
13	PO3
14	O1
15	Oz
16	Pz
17	Fp2
18	AF4
19	Fz
20	F4
21	F8
22	FC6
23	FC2
24	Cz
25	C4
26	T8
27	CP6
28	CP2
29	P4
30	P8
31	PO4
32	O2
33	hEOG (horizontal EOG)
34	vEOG (vertical EOG)
35	zEMG (Zygomaticus Major EMG)
36	tEMG (Trapezius EMG)
37	GSR
38	Respiration belt
39	Plethysmograph
40	Temperature

Figura 3: Corrispondenza segnali sensori

Array name	Array dimension	Array contents
data	40 x 40 x 8064	Video_ID x Sensor_ID x Data (sampled at 128 Hz)
labels	40 x 4	Video_ID x Label (valence, arousal, dominance, liking)

Figura 4: Tabelle a disposizione per ogni utente

3 Progettazione e sviluppo di un Multi-Layer Perceptron (MLP)

Lo scopo di questa parte del progetto è quello di sviluppare due *multi-layer perceptron (MLP)* che stimano in maniera accurata i livelli di *valence* e *arousal* delle persone mentre stanno guardando i video, sulla base dei risultati ottenuti dai sensori visti in tabella 1. Per il corretto funzionamento della MLP è di fondamentale importanza la scelta delle *features* da passare come input alla rete. Il nostro approccio è stato del tipo *'Step by step'* per notare ogni possibile miglioramento o peggioramento del risultato finale della nostra rete in base alle statistiche passate come input.

3.1 Selezione features

Questa fase è composta dell'estrazione e successivamente della selezione delle *features* statistiche che meglio approssimano i segnali in nostro possesso. In primo luogo, dobbiamo stabilire un insieme di statistiche che, a nostro avviso, potrebbero essere più significative per la corretta identificazione dei segnali; successivamente, questo pool di features viene passato come input alla *Sequential Feature Selection (SFS)* che selezionerà un sottoinsieme delle statistiche che meglio dovrebbero predire i segnali. Per garantirci una selezione più accurata possibile ogni SFS viene eseguita, ogni volta, per 10 iterazioni e le statistiche che useremo come input della MLP saranno quelle scelte il maggior numero di volte. Per un'ottimale estrazione delle statistiche abbiamo preso in considerazione diversi parametri per notare eventuali miglioramenti nel risultato finale. I parametri presi in considerazione sono stati i seguenti:

Scelta del taglio iniziale Dato che ogni partecipante guarda i video in sequenza, per assicurarci che lo stato d'animo dello stesso non venga influenzato dal video visto precedentemente, la prima parte di ogni segnale verrà scartata. Nei nostri esperimenti la parte di video scartata varia tra i 15 e i 30 secondi al fine di ottenere i risultati migliori.

Features nel tempo e features in frequenza con divisione per bande di frequenza Ricordando che un segnale può essere visto sia nel dominio del tempo sia in quello della frequenza, come possiamo vedere dalla figura 5, abbiamo scelto delle caratteristiche che a nostro avviso potessero approssimare al meglio il segnale in entrambi i domini.

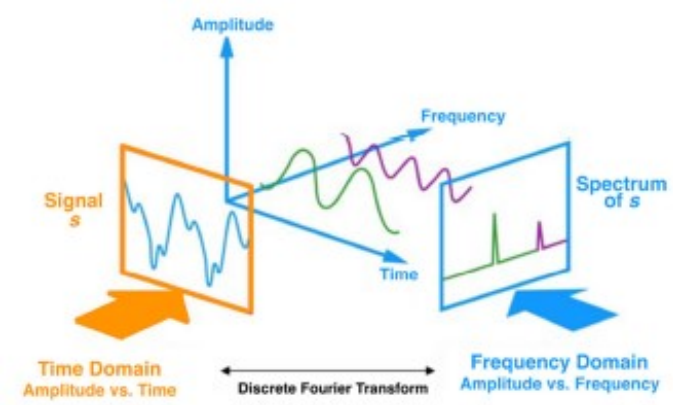


Figura 5: Segnale nel dominio del tempo e della frequenza

Nel caso del dominio nel tempo abbiamo preso in considerazione le seguenti features:

- Valor medio
- Varianza
- Mediana
- Minimo
- Massimo
- Asimmetria
- Deviazione standard
- Area sottesa

In frequenza invece abbiamo utilizzato:

- Banda totale al 99% di potenza
- Estremo inferiore di banda del 99% di potenza
- Estremo superiore di banda al 99% di potenza
- Valor medio in frequenza
- Valore del picco più alto
- Frequenza di picco più alto
- Potenza in decibel

Ci siamo dedicati alla ricerca di informazioni per migliorare i risultati, a tal proposito è stata trovata un'interessante pubblicazione riguardante il riconoscimento di valence ed arousal tramite l'elettroencefalogramma [1].

Questo paper indica che il segnale di monitoraggio dell'encefalogramma, è diviso in 4 bande di frequenza: *delta* (δ: 1–3 Hz), *theta* (θ: 4–7 Hz), *alpha* (α: 8–13 Hz), *beta* (β: 14–30 Hz) e *gamma* (γ: 31–50 Hz). Attraverso dei filtri passa banda, abbiamo quindi diviso il segnale in 4 bande di frequenza, escludendo la banda delta in quanto specificato che essa sia poco influente per i nostri scopi in quanto viene utilizzata per gli studi del sonno. Inoltre, in prima istanza, abbiamo considerato l'inserimento di un filtro passa-basso, in modo tale da "filtrare" il segnale e togliere il rumore di fondo.

Finestra unica o finestre multiple Dopo aver effettuato il taglio iniziale, è possibile considerare il segnale rimanente come un'unica finestra, ottenendo così un solo valore per ogni feature del segnale; o suddividerlo in più finestre contigue non sovrapposte per ottenere, per ogni segnale, tante statistiche quante sono le finestre.

Questa divisione in finestre multiple la possiamo vedere rappresentata in figura 6.

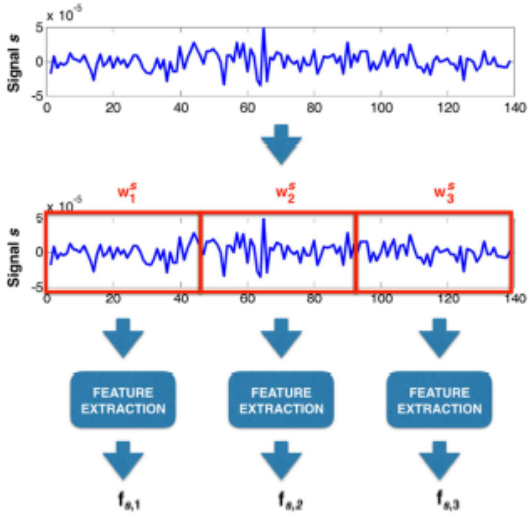


Figura 6: Divisione del segnale in finestre

Eventuale uso di segnali normalizzati Abbiamo considerato l'ipotesi di effettuare la normalizzazione sui dati in input nel dominio del tempo in modo tale da limitare l'escursione dei valori entro un certo intervallo predefinito; la normalizzazione è stata eseguita con la funzione di matlab *zscore*.

Eventuale rimozione di sensori meno utili allo scopo finale Dopo esserci opportunamente documentati abbiamo effettuato simulazioni andando a diminuire il numero di sensori utilizzati (originariamente 40), considerando solo quelli che evidenziano maggiormente una variazione dei valori di valence e arousal. Per ridurre il set di dati, sono stati implementati due script "covariance_remove" e "reduce_set" che hanno la funzione rispettivamente di rimuovere i sensori che hanno un coefficiente di covarianza maggiore di 0.95, ovvero quelli che sostanzialmente portano le stesse informazioni e che possono quindi essere considerati ridondanti, mentre il secondo crea un dataset bilanciato, nel senso che va ad eliminare i training samples che hanno Valence ed Arousal uguali.

4 Step by step solution

Come detto precedentemente, il nostro approccio al problema è stato quello di procedere per "step". Siamo partiti dall'inclusione di un numero minimo di features, per poi andare a includere tutte le possibili metriche, la *Sequential FS* andrà a selezionare tra queste, le colonne più predittive per il nostro scopo.

Primo step Come punto di partenza nel lavoro e per capire la variazione delle performance della MLP. Abbiamo incluso come input per la *Sequential FS* soltanto caratteristiche nel dominio del tempo, tra cui: media, varianza, massimo e minimo per ogni segnale dei sensori, per ogni persona. Riportiamo i risultati della MPL per quanto riguarda i valori di regressione per Valence e Arousal, avendo considerato 10 hidden neurons come riferimento:

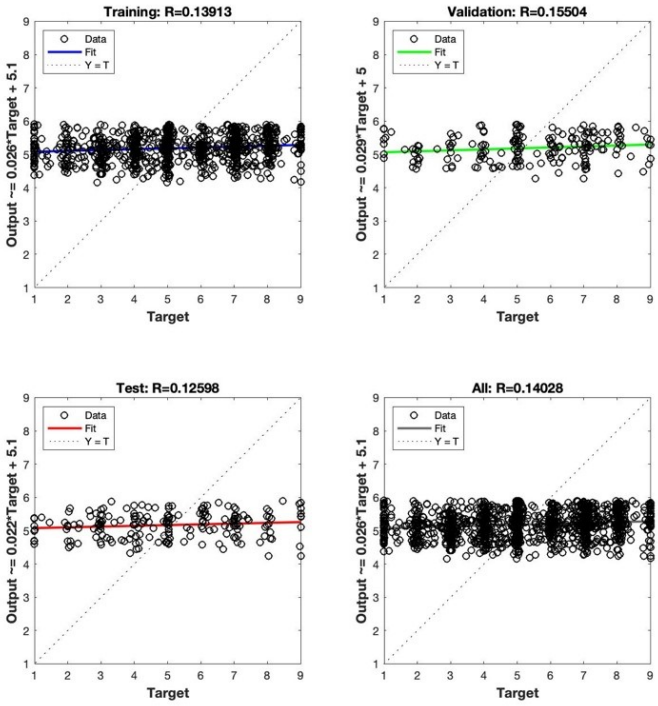


Figura 7: Valence - primo step

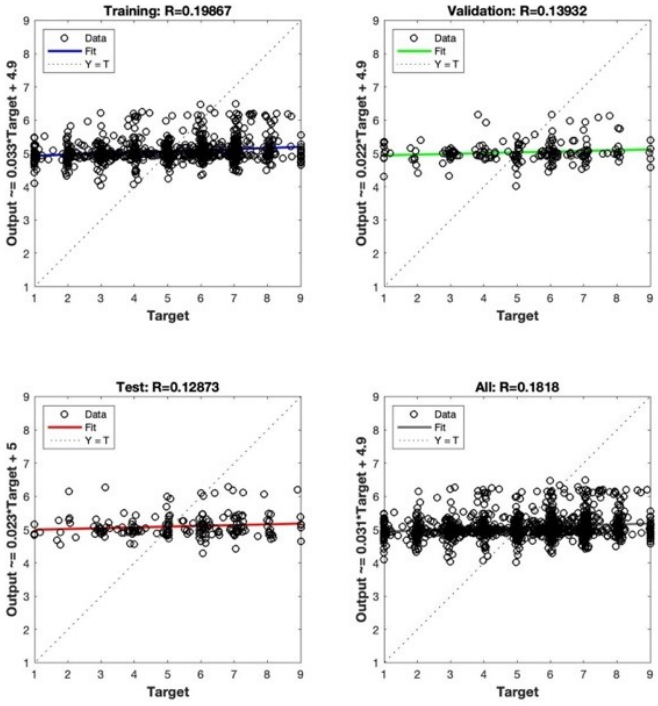


Figura 8: Arousal - primo step

Secondo step In questo caso abbiamo aggiunto il valore della skewness, ovvero la misura dell'asimmetria della distribuzione di probabilità di una variabile random attorno alla sua media; e la deviazione standard. Ottenendo questi risultati:

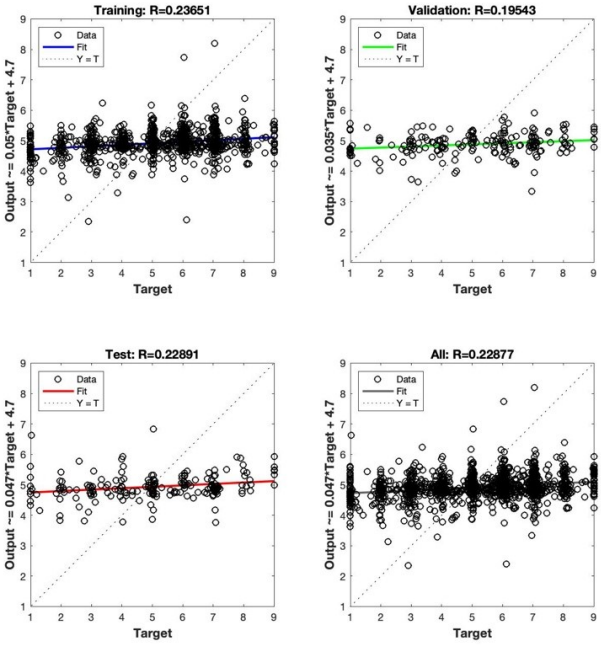


Figura 9: Valence - secondo step

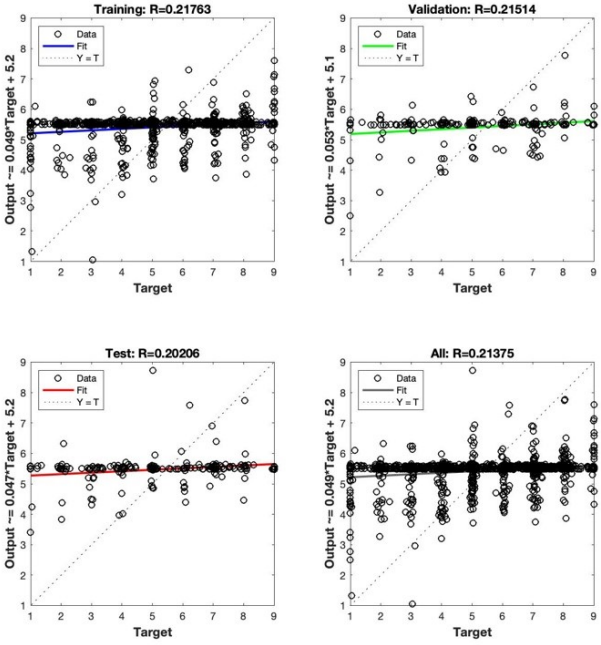


Figura 10: Arousal - secondo step

Terzo step Considerando che con le precedenti features, non è stato ottenuto un risultato soddisfacente, si procede a un nuovo tipo di approccio, andando ad utilizzare finestre multiple e normalizzando le feature. I risultati ottenuti sono i seguenti:

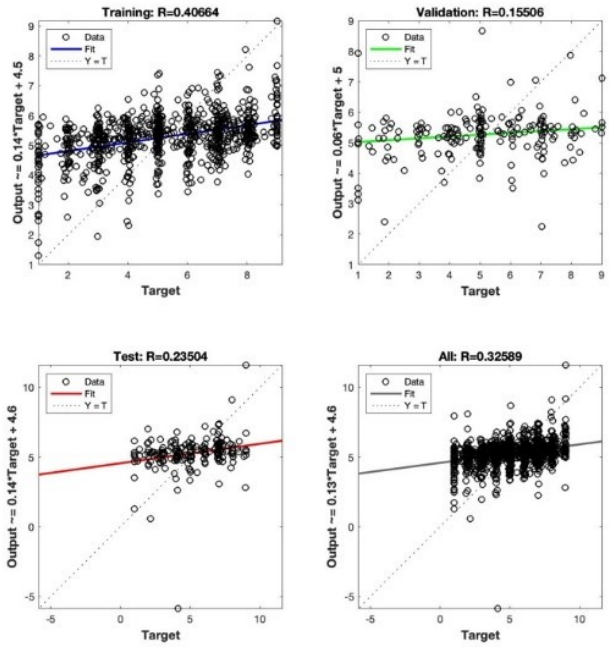


Figura 11: Valence - terzo step

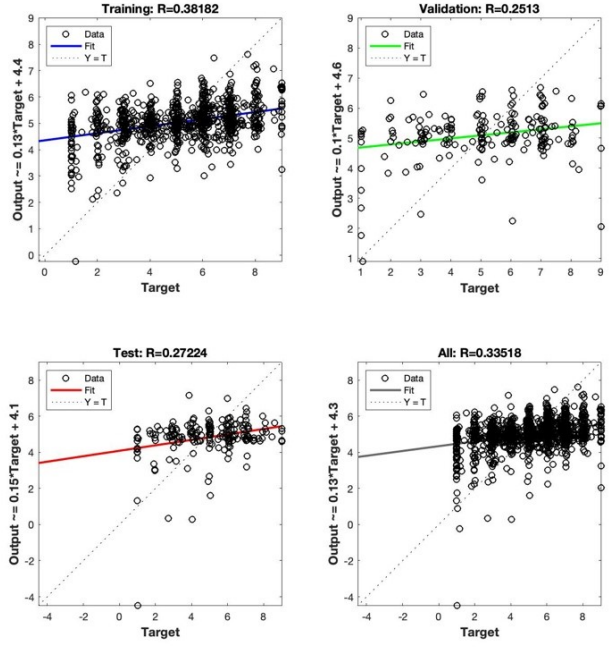


Figura 12: Arousal - terzo step

Quarto step Non ottenendo miglioramenti significativi abbiamo utilizzato anche le features nel dominio della frequenza, utilizzando anche un filtro passa basso per campionare il segnale senza il rumore di fondo. Questi sono i risultati ottenuti:

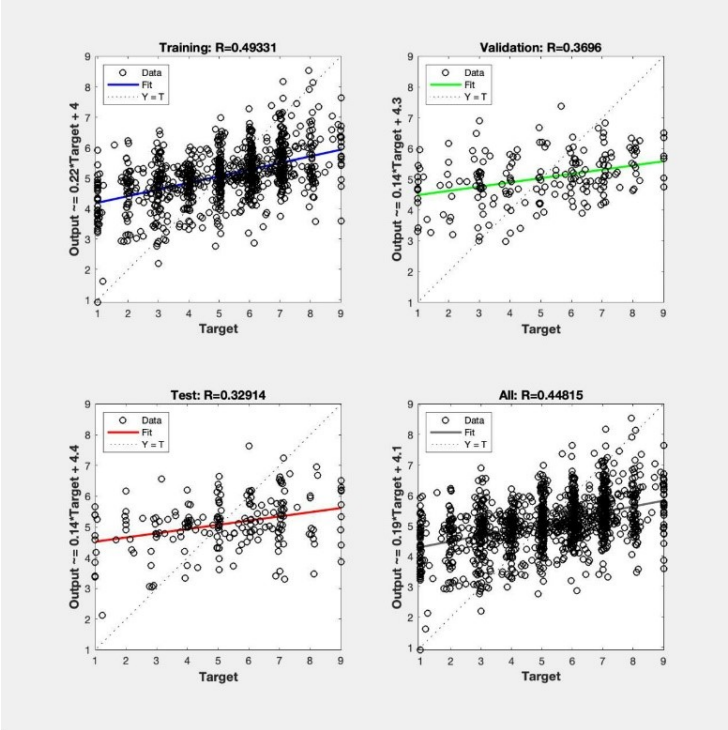


Figura 13: Valence - quarto step

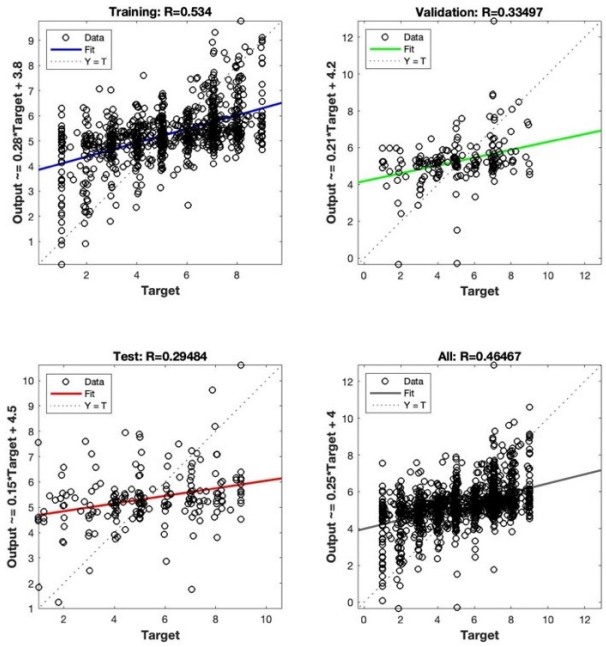
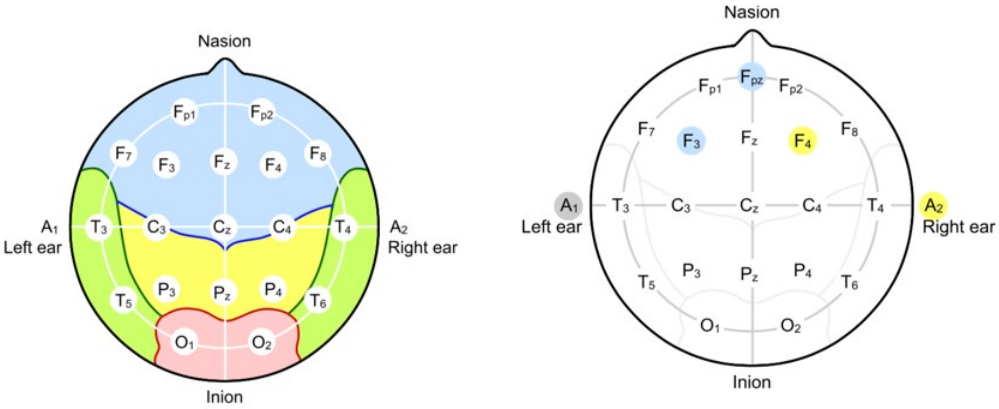


Figura 14: Arousal - quarto step

Quinto step I risultati precedenti hanno mostrato una difficoltà da parte della rete ad evitare l'overfitting. Per cercare di ridurre questo problema, abbiamo pensato di "aiutare" la *Sequential FS*, andando a diminuire il numero di colonne da selezionare. Con questo intento è stata trovata una pubblicazione [2] dove viene mostrata la disposizione dei sensori relativi al monitoraggio encefalografico, in modo tale da avere cognizione della disposizione dei sensori (Figura 15a). Sempre dallo stesso paper, abbiamo notato come in questo tipo di ricerca venisse ridotto il set di sensori considerato, per la previsione di Valence e Arousal. In particolare, sono stati considerati solo i sensori relativi al lobo frontale e dietro le orecchie. In questo modo, considerando solo questi tipi di sensori, si riduce di molto il set di dati da dare come input della Sequential FS.



(a) Disposizione dei sensori EEG

(b) Sensori scelti per il riconoscimento di Valence e Arousal

Figura 15: Studio della disposizone dei sensori EEG

I risultati ottenuti evidenziano come l'overtraining sia ridotto, ma il risultato relativo alla regressione non sia ancora considerato soddisfacente.

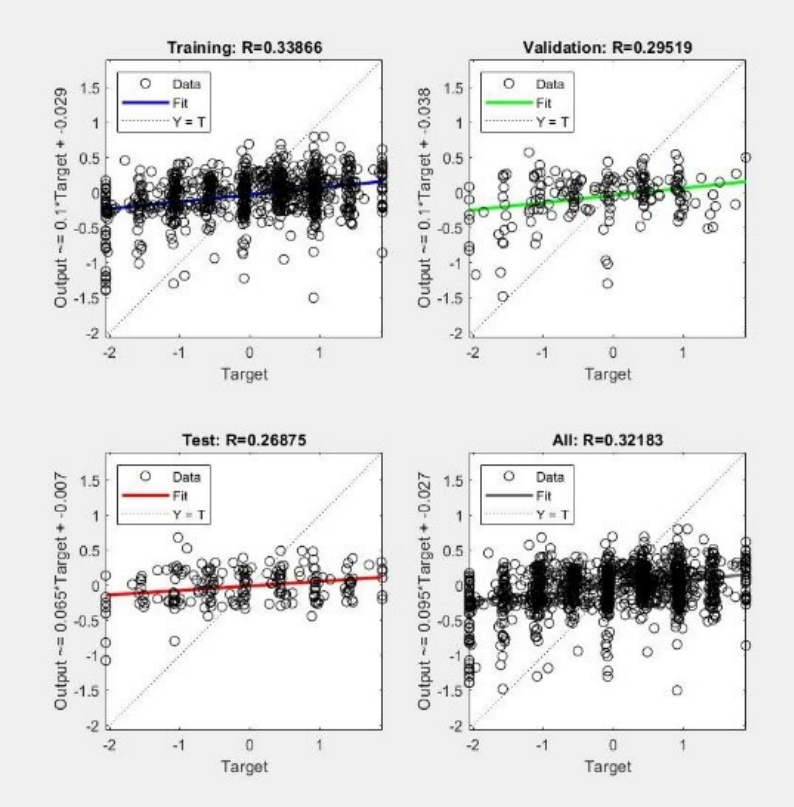


Figura 16: Valence - quinto step

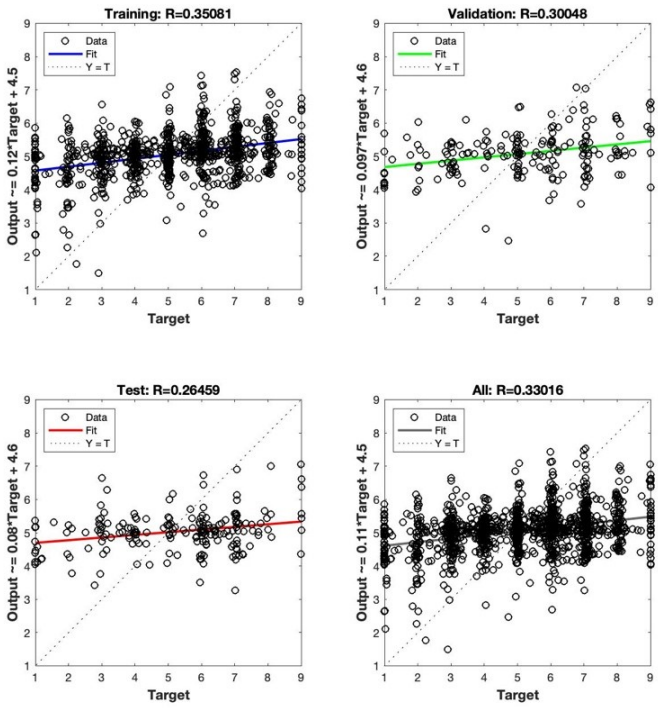


Figura 17: Arousal - quinto step

Ultimo step Come detto precedentemente abbiamo utilizzato un filtro passa banda per dividere il segnale in quattro bande di frequenza differenti, come indicato nel paper citato nel capitolo precedente. Considerando che la riduzione dei sensori non ha portato miglioramenti significativi abbiamo deciso di riutilizzare tutti i sensori messi a disposizione. I risultati finali che abbiamo ottenuto sono i seguenti:

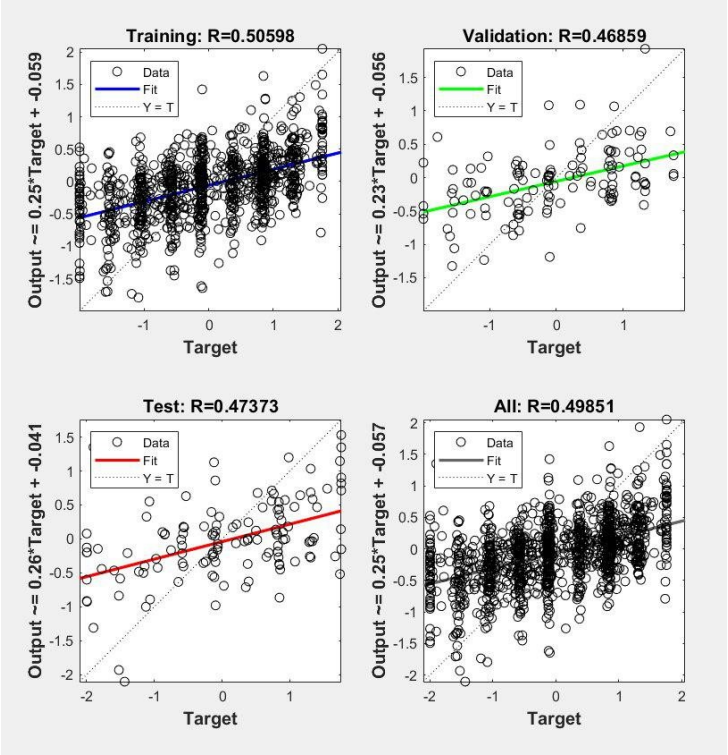


Figura 18: Valence - ultimo step

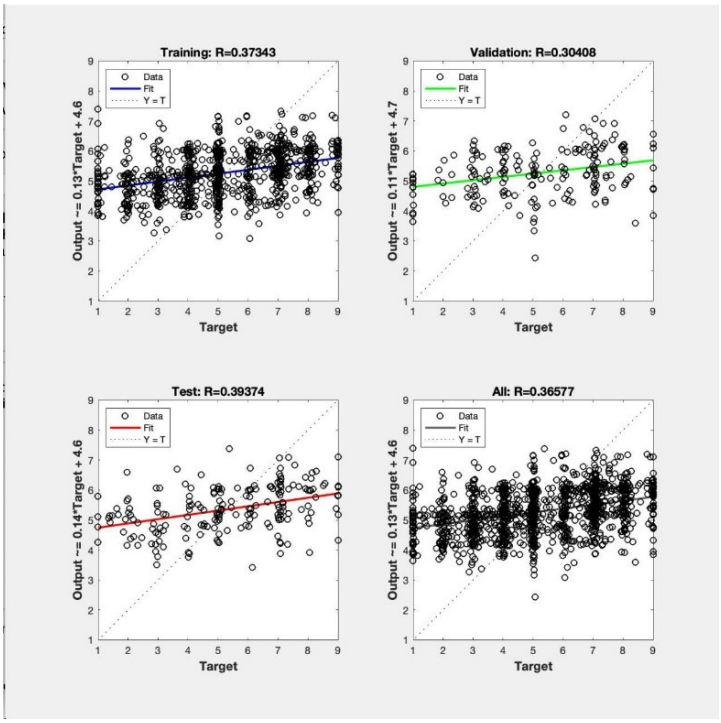


Figura 19: Arousal - ultimo step

È interessante inoltre notare che nel caso della valence con questa configurazione abbiamo ottenuto miglioramenti significati utilizzando l'algoritmo di training Bayesian Regularization.

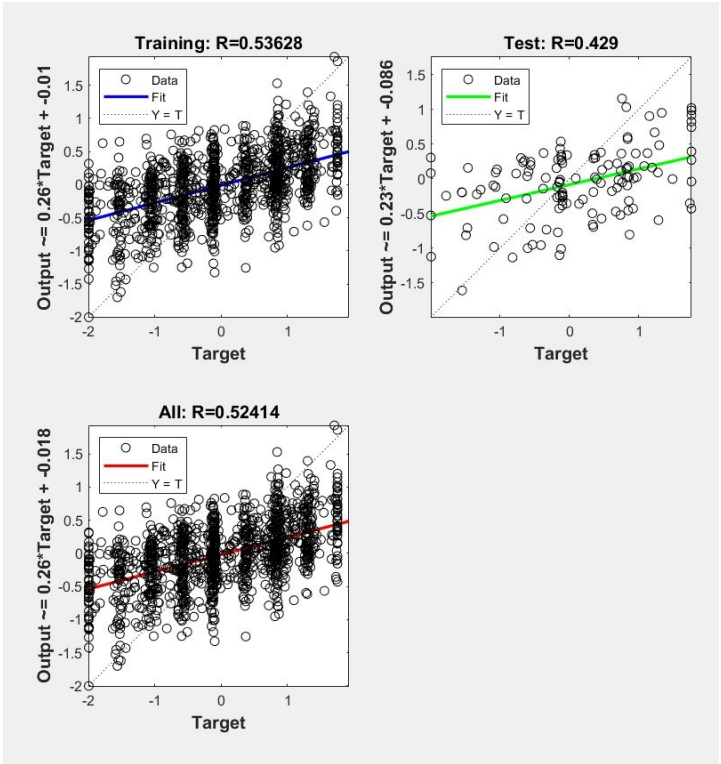


Figura 20: Valence - Bayesian Regularization

5 Radial Basis Function

Come obiettivo finale del progetto era richiesta la realizzazione di una rete di tipo RBF per la stima di *valence* e *arousal*. Il primo passo in questa direzione consisteva nel riadattare le matrici di input e output che sono rispettivamente 1280x20 e 1280x1. Per fare ciò abbiamo trasposto le matrici sensor_valence, sensor_arousal, labels_valence e labels_arousal.

A questo punto le abbiamo passate come parametri alla funzione newrb() insieme al MSE (*Mean Squared Error*) desiderato ed allo spread, appositamente calcolato. In particolare lo spread è stato calcolato nel seguente modo:

- Estrazione del Massimo e del minimo delle matrici di input;
- Calcolo della distanza d fra i due;
- $Spread = d/2$;

L'ultimo passaggio è stato ottenuto tramite ripetute prove sperimentali. Abbiamo visto che fra le varie formulazioni dello spread, quella che lo esprime come d/2 fornisce un livello di Regressione migliore rispetto a tutte le altre.

Per quanto riguarda la parte applicativa, abbiamo scelto di usare la funzione newrb() piuttosto che la newrbe() perché, dato un obiettivo (*goal*) e lo *spread*, essa crea una rete RBF in maniera iterativa aggiungendo un neurone alla volta. La condizione di terminazione è verificata quando il MSE scende sotto l'obiettivo oppure quando si raggiunge un numero massimo di neuroni utilizzati.

Dopo alcune prove abbiamo notato che variando il MSE il numero di neuroni utilizzati varia. In particolare se diminuisce il MSE, aumentano i neuroni utilizzati dalla rete. Di seguito viene riportato un grafico che mostra l'andamento della regressione calcolato con vari valori di MSE, con spread fissato al valore ottimale calcolato con il metodo mostrato precedentemente.

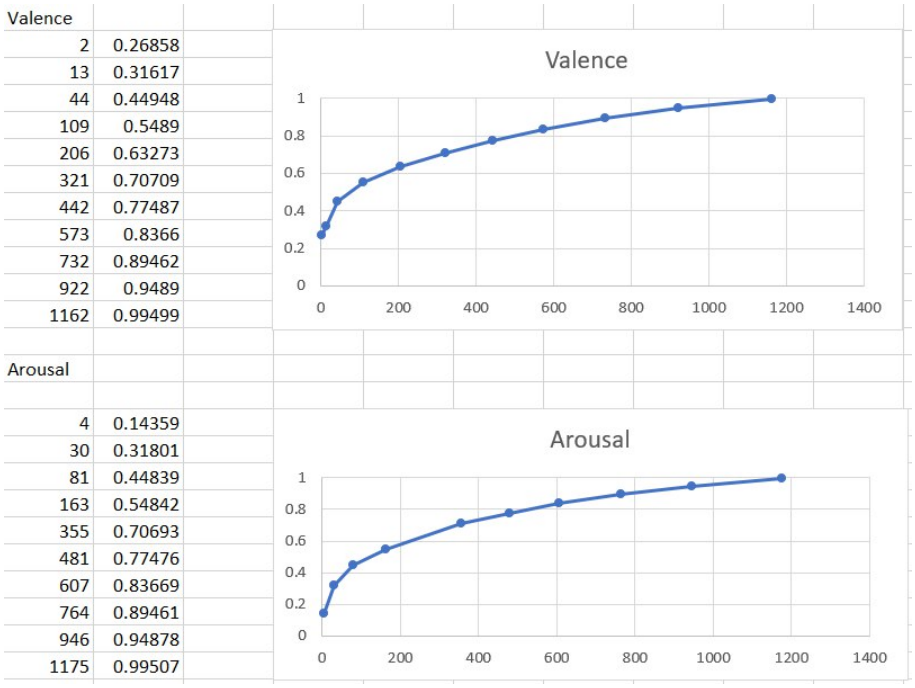


Figura 21: Neuroni necessari per Valence e Arousal dato uno specifico MSE. La prima colonna indica il numero di neuroni, la seconda l'R ottenuto

Ovviamente, aumentando enormemente il numero dei neuroni della rete, riusciamo ad ottenere un valore di Regressione vicino ad 1, ma è importante notare che tale risultato è dovuto al fatto che la rete 'memorizza' il mapping input-output, senza effettivamente imparare qualcosa da esso. Inoltre, dato il numero eccessivo di neuroni, risulta molto costoso dal punto di vista computazionale.

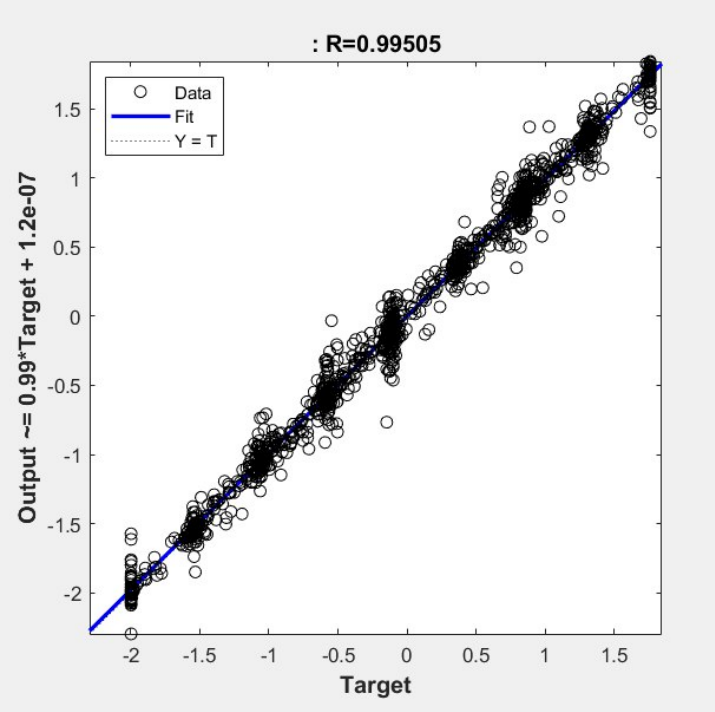


Figura 22: Valence with 1162 neurons

6 Conclusioni

Data la natura dei segnali, al fine di ottenere buoni valori di Regressione è necessario passare nel dominio della frequenza. Grazie ai numerosi tentativi effettuati, abbiamo ottenuto discreti miglioramenti ma non ancora sufficienti per la tipologia di rete richiesta. Questo può essere dovuto principalmente all'imprecisione del dataset iniziale. Questa imprecisione può essere causata dalla soggettività delle emozioni registrate dai vari individui: due persone che guardano lo stesso video potrebbero provare due emozioni distinte, oppure la stessa emozione ma con intensità differente, a causa della soggettività personale. Reputiamo comunque il nostro lavoro un buon punto di partenza per sviluppi futuri.

Riferimenti bibliografici

[1] Yuan-Pin Lin, Chi-Hong Wang, Tzyy-Ping Jung, Senior Member, IEEE, Tien-Lin Wu, Shyh-Kang Jeng, Jeng-Ren Duann, Member, IEEE, and Jyh-Horng Chen, Member, IEEE. *"EEG-based Emotion Recognition in Music Listening"*.

[2] Danny Oude Bos Department of Computer Science, University of Twente. *"ECG-based Emotion Recognition – The influence of Visual and Auditory Stimuli"*.