UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Laurea Triennale in Ingegneria Informatica

Validazione modelli di Machine Learning per il Riconoscimento Emotivo mediante approcci di calcolo della Feature Importance

Relatori:

Ing: Antonio Luca Alfeo

Prof: Mario G.C.A. Cimino

Candidato:

Giacomo Maldarella

ANNO ACCADEMICO 2022/2023

Abstract

Lo studio di ricerca affrontato si concentra su un ramo dell'intelligenza artificiale chiamato *Affective Computing*, esso è un campo interdisciplinare che si occupa di sviluppare sistemi e tecnologie capaci di riconoscere, interpretare, simulare e rispondere alle emozioni e agli stati emotivi umani.

Il Machine Learning è utile se non fondamentale nell'Affective Computing poiché permette ai sistemi di apprendere dai dati emotivi e di riconoscere e di indentificare le emozioni umane utilizzando algoritmi specifici.

Se ne ha un esempio pratico in campo medico dove questo ramo dell'intelligenza artificiale, può essere utilizzato per monitorare pazienti con disturbi dell'umore e d'ansia sociale attraverso analisi delle espressioni facciali e altri esperimenti specifici durante le terapie. In questo modo è più facile la diagnosi e si possono ottenere risultati personalizzati per il singolo paziente.

La ricerca si è basata su uno studio chiamato **K-EmoCon** condotto in Corea del Sud. Lo studio monitora due persone interagenti, con un terzo osservatore che ne facilita la valutazione, che utilizzano sensori cutanei e nervosi per valutare le emozioni reciproche. L'obiettivo è comprendere le dinamiche emotive durante l'interazione umana attraverso dati raccolti durante il processo di studio.
L'approccio scelto è stato quello di individuare un algoritmo di classificazione di Machine Learning che fosse in grado di prevedere con un'accuratezza più alta possibile l'intensità e la positività emotiva del soggetto, dell'interlocutore e di entrambi visti dall'esterno. Nel corso dello studio abbiamo osservato diversi approcci di classificazione come, ad esempio, il classificatore MLP oppure gli alberi decisionali che ci hanno permesso di avere risultati molto precisi ed accurati. Questa classificazione all'interno del dataset contenente i dati dell'esperimento ci ha permesso di assegnare istanze di dati a categorie distinte, permettendo di fare predizioni, di semplificare l'analisi e valutare le performace dei modelli.

Si è infine affrontato un aspetto cruciale dell'intelligenza artificiale: **l'Intelligenza Artificiale Spiegabile** (XAI) tramite il calcolo delle *feature importance*.
Le feature importance sono una misura utilizzata per determinare l'influenza o il contributo di ciascuna feature all'interno del modello predittivo che abbiamo usato. Questa misura permette di indentificare quali feature sono più significative per la previsione del modello e quali hanno un impatto minore o trascurabile. Per le diverse feature importance si è andati anche a valutare, tramite la correlazione di Pearson, la similarità che intercorre tra di loro per i diversi target di riferimento in modo da avere una statistica di quali sono le più informative e quali le meno informative. Infine, l'introduzione di un nuovo metodo per valutare le feature importance BoCsOr ha ampliato la conoscenza e l'efficacia della valutazione delle feature per futuri studi e applicazione dell'Affective Computing.

I risultati ottenuti dall'analisi dell'uso degli algoritmi di machine learning e dalle feature importance hanno fornito preziose informazioni sulle feature più influenti per la predizione dell'intensità e positività/negatività emotiva, consentendo una migliore comprensione dei fattori chiave che influenzano le emozioni umane durante le interazioni sociali.

Indice

Capitolo 1

Introduzione

L'ambito dell'intelligenza artificiale, trova applicazione in diversi settori, ma nell'era moderna ha rivoluzionato principalmente: la salute come nelle diagnosi e nel trattamento delle malattie, l'automotive migliorando l'efficienza dei trasporti e la sicurezza stradale, la finanza supportando l'analisi dei dati finanziari e il supporto cliente.

Uno degli ambiti emergenti più intricati e suggestivi di grande rilevanza è **l'Explainable Artificial Intelligence (XAI)**.

Il concetto di XAI rende i modelli più trasparenti e comprensibili per gli utenti e gli sviluppatori. In questo contesto, la nostra ricerca si concentra su un ramo specifico dell'intelligenza artificiale chiamato Affective Computing, che si occupa dello sviluppo di sistemi capaci di riconoscere e interpretare le emozioni e gli stati emotivi umani. La comprensione delle emozioni umane è cruciale per migliorare addirittura la qualità della vita delle persone, essa infatti va a promuovere una questione molto sottovalutata nell'era moderna, il benessere mentale.

La mia tesi esplora varie metodologie e approcci nell'ambito dell'IA per la predizione delle emozioni, con una focalizzazione sulla trasparenza e la spiegabilità degli algoritmi. Si analizzano algoritmi di machine learning come reti neurali e modelli ad albero decisionale, i quali affrontano il problema in modo diverso, generando risultati che, confrontati tra di loro, possono differire nei loro valori. In seguito, è possibile anche analizzare le **feature importance**, la correlazione tra le varie feature e infine una valutazione sulla loro importanza nel modello.

Queste misure ci hanno permesso di: effettuare la classificazione per addestrare modelli di machine learning cioè, avere un modello che impara dai dati e dalle loro etichette (le emozioni percepite nel nostro caso), per fare previsioni su nuovi input che potrebbero essere forniti in futuro e valutare le sue performance per capire quanto bene si comporta sulle previsioni.
Ciò ci ha consentito di valutare quanto le previsioni del modello corrispondono ai dati reali, aiutandoci a capire se il modello ha imparato in modo coerente e se può nel concreto generalizzare bene su nuovi input.

Infine è stato possibile identificare le variabili più significative nell'influenzare la previsione dei modelli, fornendo una maggiore trasparenza e spiegabilità delle decisioni assunte dall'intelligenza artificiale.

Il nostro caso studio si è focalizzato su un esperimento riguardante la predizione delle emozioni di due persone durante un'interazione, chiamato "K-EmoCon", condotto in Corea del Sud. Attraverso l'analisi delle espressioni facciali e l'utilizzo di algoritmi di classificazione, abbiamo mirato a comprendere le dinamiche emotive durante l'interazione umana. Come input si sono presi valori provenienti da vari sensori che, nell'individuo, hanno monitorato i segnali biomedici. Attraverso questi dati, l'algoritmo va a predire la valence e l'arousal provata dalla persona in uno specifico intervallo temporale. La *valence* rappresenta la positività o negatività dell'emozione mentre, l'*arousal* indica l'intesità dell'emozione.

L’obiettivo principale di questo studio è proprio quello di predire con la massima accuratezza possibile le emozioni provate, fornire una spiegazione del motivo della predizione e ottenere una migliore comprensione dei fattori chiave coinvolti nella predizione delle emozioni grazie all’analisi delle feature importance. Tutti i risultati dell’esperimento sono stati salvati in un dataset, che contiene la raccolta dati dei vari sensori utilizzati su un campione di persone eterogenee che hanno partecipato all’esperimento.

I vantaggi che l'intelligenza artificiale spiegabile (XAI) applicata all'Affective Computing offre sono svariati. Grazie alla spiegabilità dei modelli, è possibile comprendere come l'IA riconosce e interpreta le emozioni umane, fornendo giustificazioni razionali per le previsioni.

Abbiamo scelto di usare questo approccio perché esso è fondamentale per garantire la trasparenza e la compressibilità dei modelli di IA. Questo permette di spiegare le decisioni prese dal modello, individuare possibili bias e ci permette anche di avere una personalizzazione delle soluzioni.

I risultati più significativi sono visibili in ambito medico dove è possibile una diagnosi più accurata e personalizzata dei disturbi emotivi, migliorando il benessere dei pazienti. Inoltre, la XAI aiuta a identificare eventuali casi ambigui o di errore, consentendo di ottimizzare il modello e ottenere la fiducia dei medici.

La XAI sta diventando sempre più importante in quest’ambito dove dovrà sempre più essere integrata e si spera in futuro sia alla base delle conoscenze dei medici perché potrà fornire giustificazioni razionali per le diagnosi e fornire anche trattamenti personalizzati.

Capitolo 2

Related Works

Gregory N. Bratman professore in Nature, Health e Recreation presso la School of Environmental and Forest Sciences dell'Università di Washington, dimostra come l'esposizione alla natura può avere benefici affettivi, andando quindi ad interessare le nostre ricerche nel campo dell'Affective Computing.

Bratman afferma che:

> *Gli impatti affettivi dell'esposizione alla natura variano in termini di entità e durata. Essi spaziano da brevi cambiamenti a livello di stato emotivo, a cambiamenti più duraturi nei modelli di umore e pensiero, fino a modifiche nella prevalenza dei disturbi di salute mentale (Bratman, Olvera-Alvarez, & Gross, 2021)*

Negli ultimi anni, l'Affective Computing ha fatto notevoli progressi nell'approfondire la compressione delle emozioni umane. Si è riconosciuto come ogni individuo ha un modo unico di esprimere e percepire le emozioni, influenzato da diversi fattori personali ma soprattutto da fattori culturali e ambientali. Tuttavia, c'è ancora una sfida fa affrontare: molti modelli di machine learning sono ancora standardizzati e basati su dati aggregati, il che potrebbe ridurre la loro capacità di adattarsi alle esigenze emotive specifiche di ogni singola persona.

2.1 Il problema: Personalized Emotion IA

Il problema è che l'influenza dell'ambiente sulle emozioni può portare a variazioni significative nelle risposte emotive degli individui. Ciò potrebbe inficiare la riproducibilità e l'affidabilità degli esperimenti e delle misurazioni nell'individuo compromettendo la precisione delle previsioni e la comprensione delle dinamiche emotive personalizzate.

L'utilizzo di modelli di machine learning personalizzati, combinati con tecniche di spiegabilità (XAI), rappresenta la chiave per superare queste limitazioni. Questo approccio ci consentirebbe di individuare le caratteristiche emotive più rilevanti per ciascun individuo, identificando quali aspetti dei dati hanno un impatto significativo sulla previsione delle emozioni e permettendo di adattare e personalizzare i modelli di Affective Computing in modo da rispecchiare al meglio le esigenze e le peculiarità emotive di ciascuno. Così facendo, potremmo aprire la strada a una nuova era di soluzioni di IA più empatiche e sensibili alle sfumature delle nostre emozioni.

2.2 Il problema: La validazione del modello in ambito medico

Il problema della validazione del modello di machine learning è di fondamentale importanza nel campo medico, dove le decisioni basate su algoritmi possono avere un impatto diretto sulla salute e sulla vita dei pazienti. L'uso di modelli di machine learning in quest'ambito può portare a diagnosi più precise, personalizzazione delle terapie e identificazione tempestiva di patologie.

A causa della mancanza di trasparenza delle deep neural networks, è difficile per l'utente valutare se la decisione sia affidabile, compromettendo la fiducia con i medici (Zhang et al., 2022)

La validazione inadeguata di modelli di machine learning potrebbe portare a diversi problemi critici che possono influenzare negativamente la cura dei pazienti e la qualità delle diagnosi. Ecco alcuni esempi:

- Diagnosi errate: se il modello di machine learning utilizzato per la diagnosi medica non fosse stato adeguatamente validato su un ampio spettro di casi clinici, potrebbe produrre diagnosi errate o incomplete.
- Poca personalizzazione delle terapie: se il modello non è stato validato su una varietà di pazienti o non tiene conto di specifiche condizioni cliniche, potrebbe suggerire terapie inappropriate o inefficaci.
- Interpretabilità: i modelli di machine learning spesso riscontrano un problema molto importante chiamato "*black box*". Questo rappresenta una barriera significativa in quanto non è sempre chiaro come il modello arrivi a una determinata diagnosi o raccomandazione.

È qui che la XAI entra in gioco, grazie ad essa è possibile ottenere spiegazioni e giustificazioni sulle predizioni del modello, rendendolo più trasparente e comprensibile.

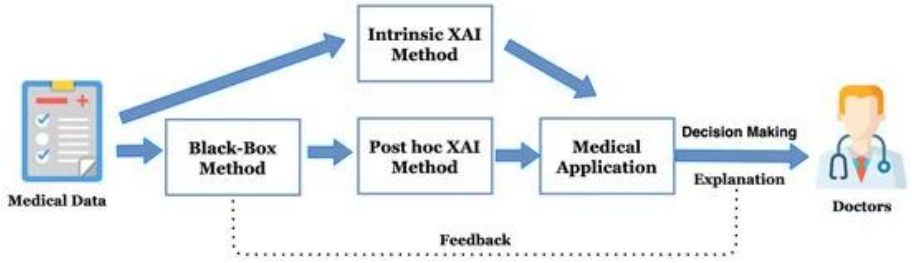


Figura 2.2: pipeline di un medical XAI

Come illustrato nella figura 2.2, l'utilizzo del metodo intrinseco di XAI consente all'applicazione medica di esaminare i dati e fornire decisioni e spiegazioni ai medici in un semplice passaggio. In alternativa, se l'applicazione medica utilizzasse XAI post hoc, verrebbero utilizzati metodi black-box per prendere decisione sui dati che, come abbiamo già visto, non è una buona pratica.

L'uso della XAI consente di affrontare il problema della validazione del modello in modo efficace. Vediamo alcuni esempi significativi:

- Interpretazione delle predizioni: la XAI fornisce spiegazioni sulle decisioni del modello, consentendo ai medici di capire meglio il ragionamento del modello.
- Feature importance: consentono di identificare le caratteristiche più influenti per le predizioni del modello. Questo può infatti aiutare a individuare fattori critici che il modello prende in considerazione per le decisioni.

Queste due problematiche rappresentano aree di grande rilevanza per migliorare la comprensione delle emozioni umane e l'assistenza sanitaria. L'introduzione dell'approccio XAI e in particolare delle feature importance apre nuove prospettive e vantaggi significativi per superare le limitazioni esistenti.

Nell'Affective Computing e in particolare, nell'esperimento preso in considerazione nel nostro studio, l'utilizzo delle feature importance per riconoscere le feature meno informative e l'utilizzo di modelli di machine learning personalizzati, ci consente di individuare con precisione le caratteristiche emotive rilevanti per ciascun individuo, creando soluzioni di IA più empatiche e sensibili alle sfumature delle nostre emozioni.

Nel contesto medico quest'uso è altrettanto cruciale, infatti una validazione inadeguata dei modelli di machine learning può portare a diagnosi errate e ad un'inadeguata personalizzazione delle terapie.

In conclusione, con la mia ricerca, ho scelto di considerare la spiegabilità dell'IA come un aspetto fondamentale, cercando di fornire un'interpretazione delle previsioni emotive attraverso l'uso delle metodologie, già citate, come XAI e il calcolo delle feature importance. Questo approccio rappresenta un passo avanti cruciale verso soluzioni di IA più etiche, trasparenti e sensibili alle esigenze individuali.

Capitolo 3

Design e Implementazione

3.1 Design

Basandomi sulla comprensione dei problemi emersi, in questo capitolo, darò una panoramica approfondita che servirà da base per la presentazione di una soluzione altrettanto dettagliata. Fornirò una chiara descrizione dell'approccio usato mettendo soprattutto in luce le varie implementazioni adottate quali: la classificazione, i modelli utilizzati con in particolare i vari algoritmi e l'importanza delle caratteristiche.

3.1.1 Classificazione

La classificazione nel machine learning è una tecnica fondamentale che consiste nel costruire modelli in grado di assegnare un'istanza di dati a una o più categorie predefinite. In altre parole, il processo di classificazione consiste nell'attribuire una classe predefinita (nel nostro caso le classi sono i numeri interi da 1 a 5, assegnabili ai diversi target utilizzati quali *external_arousal, external_valence, partner_arousal,partner_valence,self_arousal,self_valence*) a un input in base alle caratteristiche intrinseche di quell'input. L'obiettivo è quello di insegnare al modello a riconoscere schemi e relazioni nei dati in modo che possa effettuare predizioni accurate su nuovi dati che non ha mai visto in precedenza.

I modelli di classificazione nel machine learning possono variare sia in complessità ma anche in accuratezza di previsione, quelli da noi trattati sono stati: **reti neurali** - MLP (Multi-Layer Perceptron), gli **alberi decisionali** - Decision Tree e **K-Nearest Neighbors** (K-NN).

3.1.2 Regressione

La differenza fondamentale tra un algoritmo di classificazione e uno di regressione risiede nell'approccio alla variabile target. Un algoritmo di classificazione tratta ciascuna classe della variabile target come un'entità separata, ignorando le possibili somiglianze tra di loro. D'altra parte, un algoritmo di regressione cerca di prevedere un valore numerico come output, considerando le somiglianze tra le diverse categorie della variabile target.

Un algoritmo di regressione, ad esempio, ci è servito quando abbiamo approfondito la nostra ricerca su BoCsOR e proprio per questo verrà illustrato successivamente.

3.1.3 Modelli utilizzati

La ricerca per identificare l'algoritmo più adatto a gestire efficacemente il problema ha condotto alla valutazione e al confronto dei seguenti modelli:

MLP Classifier: algoritmo di classificazione basato su reti neurali artificiali. È composto da diversi strati di neuroni, tra cui uno strato di input, uno o più strati nascosti e uno strato di output. Ogni neurone riceve input ponderati dalle connessioni con i neuroni nel layer precedente e restituisce un output che viene passato al successivo layer, fin quando poi non viene raggiungo l'output finale.

Decision Tree: algoritmo di classificazione basato su un modello ad albero che suddivide iterativamente i dati in base alle caratteristiche più significative. Inizia con un nodo radice che rappresenta l'intero set di dati e poi si ramifica in nodi figli corrispondenti alle decisioni basate su determinate caratteristiche dei dati e ai valori di output desiderati.

K-Nearest Neighbors: algoritmo di classificazione utilizzato per la classificazione di dati in base alla loro somiglianza con i dati di addestramento. L'idea di base di K-NN è semplice: un oggetto viene classificato in base alla maggioranza delle classi dei suoi "vicini" più prossimi nel dataset di addestramento.

CatBoostRegressor: algoritmo di regressione basato su alberi di decisione che si concentra su una migliore gestione delle variabili categoriche. Utilizza un metodo di ottimizzazione chiamato "boosting" per addestrare un insieme di alberi sequenziali che migliorano progressivamente le previsioni. Questo algoritmo è noto per la sua robustezza e facilità d'uso, richiedendo meno sforzo nella preparazione dei dati rispetto ad altri algoritmi di regressione.

3.1.3.1 Alberi di decisione

Come abbiamo appena citato l'algoritmo Decision Tree fa uso degli alberi di decisione. Si tratta di veri e propri alberi dove ogni nodo rappresenta una *feature*, ogni *link/branch* rappresenta una rule e ogni foglia rappresenta un *outcome*. Essi sono largamente utilizzati perché imitano il pensiero a livello umano; quindi, è molto semplice comprendere i dati e fare delle buone interpretazioni.

Gli algoritmi che ti permettono di andare a costruire un albero di decisione sono svariati:

1. CART *(Classification and Regression Trees)*
2. ID3 *(Iterative Dichotomiser 3)*
3. C4.5 *(successore ID3)*
4. MARS *(Multivariate Adaptive Regression Splines)*
5. CHAID *(Chi-squared Automatic Interaction Detector)*

Di seguito verranno spiegate le varie misure essenziali e i primi tre algoritmi più significativi.

3.1.3.2 Entropia

L'entropia di una variabile casuale è il livello medio di "informazione" o "incertezza" inerente ai possibili esiti della variabile. Data una variabile casuale discreta X che assume i valori dell'alfabeto Y distribuita secondo $p: Y \to [0,1]$:

$$H(X) = \sum_{x \in Y} p(x) \log p\,(x)$$

- X: il corrente set per la quale l'entropia inizia ad essere calcolata.
- Y: il set di classi.
- p(x): la proporzione del numero di elementi nella classe Y rispetto al numero di elementi nel set X.

3.1.3.3 Information Gain

L' information gain è la misura della differenza di entropia da prima a dopo che l'insieme S sia diviso su un attributo A. In altre parole, quanta incertezza in S è stata ridotta dopo aver diviso l'insieme S sull' attributo A.

$$IG(A,S) = H(S) - \sum_{t \in T} p(t)H(t)$$

- H(S): entropia del set S.
- p(t): la proporzione tra il numero di elementi in t e il numero di elementi nell'insieme S.
- H(t): entropia del subset t.

3.1.3.4 Gini Index

Indice utilizzato come funzione di costo per valutare le suddivisioni nel dataset. Esso dà un'idea di quanto sia buona uno split in base a quanto sono miste le classi nei due gruppi creati dallo split stesso. Una separazione perfetta si traduce in un gini index pari a 0.

$$Gini = \sum_{i \neq j} p(i)p(j)$$

3.1.3.5 Algoritmo: Cart

Step dell'algoritmo:
1. Calcolare l'indice Gini per il set di dati
2. per ogni attributo: 1.1 calcolare l'indice gini per tutti i valori categorici 2.1 prendere l'entropia informativa media per l'attributo corrente 3.1 calcolare il gini gain
3. scegliere chi ha il miglior attributo con gini gain.
4. ripeti finché non otteniamo l'albero desiderato

3.1.3.6 Algoritmo: Id3

Algoritmo utilizza un approccio ricorsivo per costruire l'albero decisionale.
Per creare un albero, prima di tutto bisogna identificare il root node. Per fare ciò bisogna svolgere una top-down greedy search per determinare l'attributo che meglio classifica i dati di addestramento e, essendo un algoritmo ricorsivo, andare a ripeterlo per ogni ramo.

Step dell'algoritmo:

1. calcolare l'entropia per il data-set.
2.per ogni attributo: 1.1 calcolare l'entropia per tutti i valori categorici 2.1 prendere l'entropia informativa media per l'attributo corrente 3.1 calcolare l'information gain.
3. prendere l'information gain più alto.
4. ripetere finché non otteniamo l'albero desiderato.

3.1.3.7 Algoritmo: C4.5

Algoritmo ricorsivo che seleziona ricorsivamente la caratteristica che fornisce il massimo information gain e la utilizza per suddividere ulteriormente l'albero.

Step dell'algoritmo:

1. Scegliere il dataset con gli attributI e gli attributi target definiti.
2. calcolare l'Information gain e l'entropia per ogni attributo.
3. Prendi l'attributo con maggiore information gain, e rendilo root node.
4. calcolare l'information gain per gli attributi restanti.
5. creare il recurring child nodes avviando la suddivisione in corrispondenza del nodo di decisione.
6. ripetere questo processo fino a coprire tutti gli attributi.

3.1.4 Feature Importance

L'**Explainable Artificial Intelligence (XAI)** si dedica a sviluppare algoritmi e strumenti che consentono di comprendere il ragionamento alla base delle decisioni prese da modelli di intelligenza artificiale. L'obiettivo è spiegare il "perché" di un determinato output o risultato generato da un modello, consentendo agli utenti di prendere decisioni consapevoli e di interpretare i risultati in modo significativo.

Una delle strategie impiegate per spiegare i risultati dei modelli è quella delle **feature importance** (importanza delle caratteristiche). Questa misura valuta l'influenza di ciascuna caratteristica o variabile del dataset sull'output predetto dal modello. In sostanza, misura quanto ogni attributo contribuisce alle previsioni effettuate dall'algoritmo, permettendo di comprendere quali fattori siano più influenti nel guidare le previsioni e come ciascuna variabile contribuisca alla formazione delle stesse.

3.1.4.1 Calcolo negli alberi decisionali

Quello che andremo ad approfondire ora è capire i calcoli matematici che ci sono dietro le feature importance.

Prima di tutto dobbiamo fare delle assunzioni:

1. abbiamo un problema di classificazione binaria per predire se un'azione è valida o meno.
2. abbiamo addestrato un DecisionTreeClassifier sui dati di addestramento.

Il calcolo delle feature importance comporta due passaggi:

1. Calcolare l'importanza per ogni nodo.
2. Calcolare ogni feature importance utilizzando la suddivisione dell'importanza del nodo su quella feature.

1. Per calcolare l'importanza di ogni nodo nell'albero decisionale, la formula è la seguente:

$$Importance\_Node_k = (\%\_of\_sample\_reaching\_Node_k \; X \; Impurity\_Node_k - \%\_of\_sample\_reaching\_left\_subtree\_Node_k X \; Impurity\_left\_subtree\_Node_k - \%\_of\_sample\_reaching\_right\_subtree\_Node_k X \; Impurity\_right\_subtree\_Node_k) / 100$$

La formula calcola l'importanza del nodo sommando il contributo ponderato del Gini Index di tre parti del nodo: il nodo stesso, il sottoalbero sinistro del nodo e il sottoalbero destro del nodo. I contributi di ciascun sottoalbero sono pesati dalla percentuale di campioni che raggiungono i rispettivi sottoalberi. Infine, il risultato viene diviso per 100 per ottenere un valore normalizzato tra 0 e 1.

2. Passiamo ora invece a calcolare le feature importance per ogni feature presente, la formula da applicare è:

Feature importance for feature K= *Σnode's importance splitting on feature K / Σ all node's importance*

Il numeratore è una somma delle importanze dei nodi però di tutti i nodi che si dividono su una particolare caratteristica "k" diviso la somma di tutte le importanze dei nodi.

3.1.5 Algoritmi di valutazione

Nella nostra analisi, utilizzeremo tre diverse metriche per valutare l'importanza delle caratteristiche*: Permutation Importance*, *SHAP values* e *Buondary Crossing Solo Ratio*.

3.1.5.1 Permutation Importance

Consiste nel calcolare quanto cambia la performance del modello quando si permutano casualmente i valori di una specifica feature, mantenendo gli altri invariati.

L'approccio funziona nel seguente modo: per ogni feature di interesse, il valore della feature viene mescolato casualmente tra tutte le istanze del dataset, creando un set di dati "permutato". Questo nuovo set di dati con le feature permutate viene quindi passato al modello, che genera nuove previsioni.

La differenza tra le perfomance del modello sul set di dati originale e sul set di dati permutato non è altro che l'importanza della feature. Se la feature è importante per il modello, le previsioni peggiorano sensibilmente quando i suoi valori vengono permutati. D'altro canto, se la feature ha poco impatto sulle previsioni, la performance del modello dovrebbe rimanere relativamente stabile.

In altri termini le feature che causano un calo significativo delle performance quando vengono permutate casualmente sono considerate più importanti, mentre quelle che hanno meno effetto sono considerate meno importanti.

3.1.5.2 Shapley Additive Explanations

Il concetto chiave di SHAP è il valore di Shapley, esso rappresenta il contributo marginale di una feature in una previsione, considerando tutte le possibili combinazioni di variabili e attribuendo a ciascuna variabile una parte della differenza tra le previsioni con e senza la variabile stessa. In altre parole, misura quanto ciascuna variabile contribuisce alla differenza tra la previsione effettiva e la previsione media. Gli SHAP values ci permettono di valutare quanto ogni caratteristica influisce positivamente o negativamente sulla variabile target.

3.1.5.3 Boundary Crossing Solo Ratio

Fin ora abbiamo visto due algoritmi chiavi per valutare l'importanza delle caratteristiche. Questi modelli però hanno dei problemi e delle limitazioni come, ad esempio, il costo computazionale. La complessità temporale di SHAP cresce in modo esponenziale con il numero di caratteristiche e in modo lineare con il numero di campioni nel dataset. Questo problema non è specifico solo di SHAP, ma riguarda la maggior parte delle misure delle feature importance, che tendono ad essere computazionalmente costose.
Per andare a risolvere questo problema e quindi per ridurre i costi computazionali si introducono le *counterfactual explations*, esse si basano sul concetto di "controfattualità" che riguarda ciò che sarebbe potuto accadere se i dati fossero stati diversi o se una determinata variabile fosse stata cambiata. Andando nello specifico, data un'istanza di dati 'i' e la sua classe predetta, un controfattuale è un'istanza 'c' simile a 'i' che è stata riconosciuta come appartenente a una classe diversa. Ciò corrisponde a trovare quella simile istanza e comprendere la minima modifica necessaria per cambiare l'esito della classificazione.
BoCsOR risolve questi problemi perché è un metodo che valuta quanto una specifica caratteristica influenzi la classificazione del modello quando il valore di quella caratteristica viene variato. Questo viene fatto confrontando le classificazioni di campioni vicini al *confine decisionale* del modello con le classificazioni ottenute dai corrispondenti campioni controfattuali dove una caratteristica è stata modificata. La frequenza con cui i campioni vicini al confine decisionale producono un risultato di classificazione diverso rispetto ai compaiono controfattuali è utilizzata per determinare l'importanza delle caratteristiche.
Il confine decisionale di un modello di classificazione non è altro che una regione nello spazio delle feature dove il modello prende decisioni diverse per diverse classi di output. Per individuare i campioni che si trovano vicino al confine decisionale, un approccio è quello di considerare la minima distanza tra le diverse classi. Questo valore, misurato tipicamente attraverso la distanza euclidea, rappresenta la separazione tra un'istanza e la sua istanza più vicina appartenente alla classe controfattuale. Per individuare i controfattuali si possono considerare i suoi K-Nearest-Neighbors della classe controfattuale (a,b,c nella figura 3.1.1).

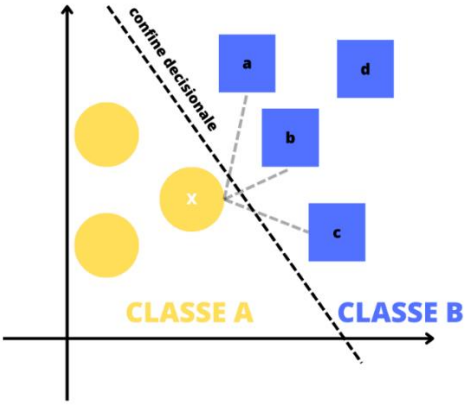


Figura 3.1.1

Tuttavia, sorge una sfida: l'istanza più vicina di una classe diversa potrebbe non corrispondere alla minima modifica richiesta per ottenere una classificazione diversa. Per superare questo problema, vengono creati dei punti intermedi tra ogni controfattuale possibile e l'istanza originale. E

attraverso un'esplorazione *step-by-step* lungo i segmenti tra il campione in questione e i suoi vicini, il punto intermedio più vicino (X->a nella figura 3.1.2) che corrisponde a un esito di classificazione diverso viene considerato come il controfattuale più vicino per il campione in questione.

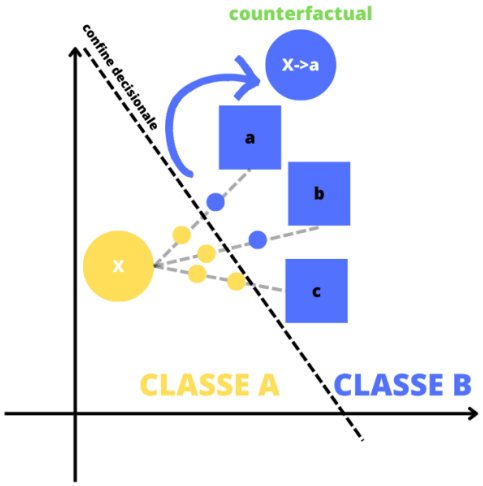


Figura 3.1.2

Ci sono però delle caratteristiche che da sole possono portare al superamento del confine decisionale così da causare un cambiamento della classificazione. Per andare a trovarle si parte dal controfattuale e si sostituisce (uno alla volta) i valori delle caratteristiche con quelli del campione originale, se questa sostituzione corrisponde al superamento del confine decisionale, tale caratteristica viene considerata rilevante.

Infine, BoCSoR valuta l'importanza di una caratteristica considerando la frequenza con cui le modifiche di quella caratteristica da sole comportano il superamento del confine decisionale del modello, considerando i campioni vicini ad esso. L'algoritmo ha una complessità temporale caratterizzata da una crescita lineare rispetto al numero di caratteristiche e da una crescita quadratica rispetto al numero di campioni, questa complessità temporale è minore se confrontata con quella di SHAP (crescita lineare rispetto al numero di campioni e da una crescita esponenziale rispetto al numero di caratteristiche).

3.2 Implementazione

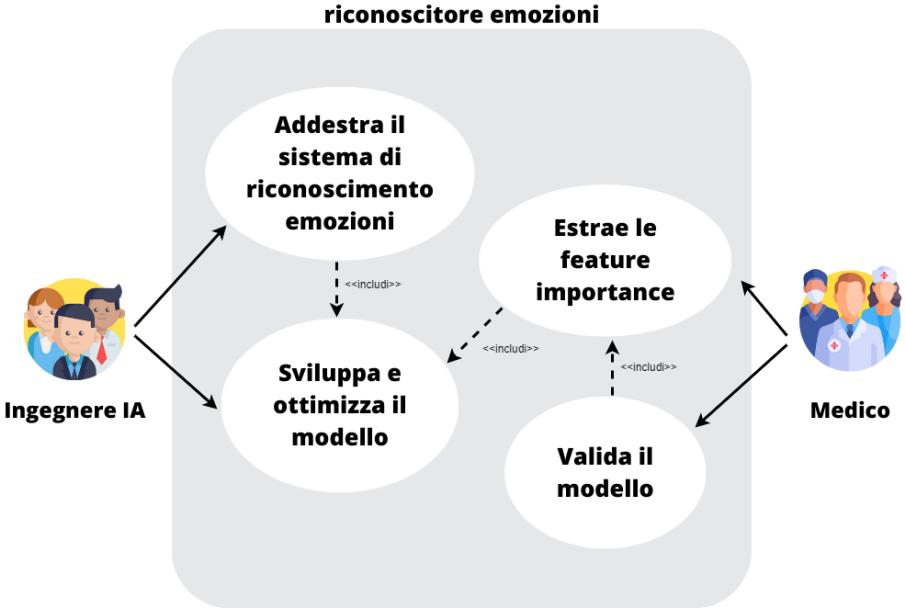


Figura 3.2

Un possibile **Use-Case** del software realizzato è quello mostrato in figura. Nel nostro scenario, il software di riconoscimento delle emozioni si basa su dati raccolti da numerosi sensori posti sul corpo di una persona che vanno a valutare emozioni e i parametri vitali. Proprio per questo il principale settori di applicazione è il campo medico.

Il medico è un attore principale e svolge un ruolo centrale essendo il destinatario principale del nostro software.

Dall'altra parte, abbiamo un attore secondario, un ingegnere specializzato in machine learning e IA. Esso è responsabile della costruzione del modello e collabora con il medico per assicurarsi che il software soddisfi le esigenze mediche.

Casi d'Uso per Ingegneria IA:

Sviluppo e Ottimizzazione: L'ingegnere IA sviluppa e ottimizza modelli di machine learning come MLP (Multi-Layer Perceptron) e DecisionTree per il riconoscimento delle emozioni. L'obiettivo è creare un modello che abbia il potenziale per imparare in modo efficace dall'addestramento. L'ottimizzazione invece è dipendente dai risultati ottenuti tramite le feature importance e la validazione del modello.

Addestra il sistema di riconoscimento emozioni: L'ingegnere IA dopo aver sviluppato e ottimizzato il modello, addestra il sistema di riconoscimento delle emozioni sviluppando algoritmi e modelli di machine learning che consentono al sistema di comprendere ed interpretare le emozioni umane in base ai dati forniti.

Casi d'Uso per il Medico:

Analisi delle Feature Importance: Il medico analizza le feature importance dei modelli di machine learning per identificare quali attributi o caratteristiche influenzano maggiormente le previsioni del

sistema. Questa analisi contribuisce alla comprensione dei fattori chiave che guidano le decisioni dei modelli. È importante notare che questa analisi, proprio per la convenienza nell'uso della XAI, viene svolta da un non componente dell'IA.

Valida il Modello: Il medico utilizza i risultati per determinare se il modello che è stato addestrato è in grado di fare previsioni accurate e affidabili sui dati. Questo processo aiuta a valutare l'efficacia del modello e la sua capacità di generalizzare dai dati di addestramento ai nuovi dati.

In realtà però un software di riconoscimento delle emozioni potrebbe essere applicato in numerosi ambiti:

1. Industria del marketing: il riconoscimento delle emozioni può essere utilizzato per valutare le reazioni dei consumatori a pubblicità, prodotti o servizi. Questo aiuta le aziende a ottimizzare le loro strategie di marketing in modo da suscitare risposte emotive positive.
2. Educazione sicurezza e benessere sul lavoro: nel campo dell'istruzione come dei lavoratori, il riconoscimento delle emozioni può essere applicato per valutare l'efficacia degli strumenti didattici e lavorativi. Ciò consente di identificare situazioni di elevato stress emotivo e prendere provvedimenti per garantire la sicurezza e il benessere degli studenti e dei lavoratori.
3. Settore dell'intrattenimento: nel settore dei giochi o dell'intrattenimento, il riconoscimento delle emozioni può essere utilizzato per personalizzare l'esperienza dell'utente. Un gioco potrebbe adattarsi dinamicamente alla reazione emotiva di un giocatore, rendendo l'esperienza più coinvolgente.

Capitolo 4

Case Study

4.1 Cenni sulle librerie utilizzate

Il codice è stato interamente implementato utilizzando la libreria **scikit-learn** di Python.

Questa libreria è fondamentale perché offre un'ampia gamma di algoritmi di classificazione e perché fornisce moduli di pre-elaborazione dei dati, pipelining, analisi delle prestazioni e funzionalità per la selezione delle feature.

La libreria ci ha permesso di:

- effettuare una divisione dei dati tra set di addestramento e set di test utilizzando la funzione *test_train_split*,
- utilizzare i classificatori come *MLPClassifier, DecisionTreeClassifier* e *KNeighborsClassifier*,
- addestrare i classificatori tramite la funzione *fit*
- calcolare l'accuratezza del modello, cioè la percentuale di predizioni corrette rispetto al numero totale di predizioni, tramite la funzione *score*.

La libreria **Pandas** invece è stata utilizzata per caricare i dati del dataset tramite la funzione *read_csv*.

Per la creazione dei grafici, si è fatto uso della libreria **matplotlib**, in particolare attraverso le funzioni *bar*, *xticks*, *ylabels*, *xlabels*, *ylim*, *title* e *show*. E inoltre si è anche usata la libreria *seaborn* che è basata su matplotlib ma è progettata per creare grafici statistici e per generare rapidamente visualizzazioni di dati complessi.

Per lavorare con gli array multidimensionali e matrici è stata poi usata la libreria *nunmpy*, molto importante infatti per i dati di input per le label di addestramento e di test.

Per l'importanza delle caratteristiche si è fatto ricorso alla libreria **sklearn.inspection**, un modulo all'interno della libreria scikit-learn, che ci ha permesso di usare le funzioni come *feature_importance_* e *permutation_importance*.

Per cacolare gli SHAP values invece, la libreria utilizzata è stata **shap**. Essa ci ha permesso ad esempio tramite l'utilizzo della funzione *shap.explainer* di calcolare i valori specifici.

Si è rilevata utile la libreria **BoundaryCrossingSoloRatio**, per andare ad usare l'algoritmo BoCsOR, e la libreria *sklearn.utils* che fornisce varie funzioni come quella del *resample*. Questo ci ha permesso di avere un sampling randomico bilanciato sul valore della classe e ci ha assicurato di beccare lo stesso numero di esempi per ciascuna classe senza sbilanciare l'analisi verso l'una o l'altra.

4.2 Il Dataset

K-EmoCon è un dataset che raccoglie dati raccolti da vari sensori utilizzati su un campione di 32 persone eterogenee (sia uomini che donne, di età compresa tra 19 e 36 anni), mentre partecipavano a una discussione, divisi in coppie, sul tema "L'accettazione dei rifugiati dello Yemen

sull'isola di Jeiu" in Corea del Sud.
L'unico requisito per la partecipazione era una minima conoscenza della lingua inglese. Durante la discussione, ogni coppia di partecipanti è stata dotata di sensori e gli interlocutori si sono alternati in turni di circa 2 minuti ciascuno, fino ad un totale di 10 minuti. La conversazione è stata ripresa da una telecamera frontale, con la presenza di un moderatore e un osservatore esterno.
Alla fine del dibattito, i partecipanti hanno rivisto il video di sé stessi e hanno fornito valutazioni sulle emozioni provate in determinati istanti della ripresa. Questo modo di operare è definito *affect judgement protocol* e viene ampiamente utilizzato nell'ambito dell'autovalutazione delle emozioni. È ritenuto particolarmente performante perché procedendo con le annotazioni a posteriori si evita di interrompere il flusso della discussione, condizione necessaria per avere dati più accurati.
Al fine di uniformare i dati non è stato chiesto di fornire un nome come descrizione dell'emozione (ad esempio gioia, felicità rabbia), ma di fornire due parametri:

1. Intensità (Arousal)
2. Positività (Valence)

La valutazione doveva essere effettuata ad intervalli di 5 secondi e su una scala numerica da 1 a 5. Il soggetto doveva fare poi lo stesso tipo di lavoro cercando di capire le emozioni della controparte, assegnando anche in questo caso dei punteggi.
Anche agli osservatori esterni è stato chiesto di valutare entrambi i soggetti. I dati disponibili sono dunque: *self_arousal, self_valence, partner_arousal, partner_valence, external_arousal, external_valence*. La particolarità dello studio è proprio questa, le emozioni sono state valutate da 3 punti di vista differenti: dal soggetto in questione, dal suo interlocutore e da soggetti esterni alla discussione.

Durante un dibattito, i partecipanti indossavano i seguenti sensori:

1. *Empatica E4 Wristband*: catturava la fotopletismografia (PPG), l'accelerazione a 3 assi, la temperatura corporea (TEMP) e l'attività elettrodermica (EDA). La frequenza cardiaca e l'intervallo inter-battito (IBI) venivano derivati dalla pressione sanguigna (BVP) misurato da un sensore PPG.
2. *Polar H7 Bluetooth Heart Rate Sensor*: rilevava le frequenze cardiache utilizzando un sensore elettrocardiogramma (ECG) ed era utilizzato per integrare un sensore PPG nell'E4, che è suscettibile ai movimenti.
3. *NeuroSky MindWave Headset*: raccoglieva i segnali dell'elettroencefalogramma (EEG) tramite due elettrodi a secco, uno sulla fronte (canale fp1-10/20 system nel lobo frontale) e uno sul lobo dell'orecchio sinistro.
4. *LookNTell Head-Mounted Camera*: una telecamera attaccata a un'estremità di una fascia di plastica che veniva indossata sulla testa dei partecipanti per catturare video dalla prospettiva in prima persona.

Tutti i dispositivi elencati possono funzionare in un ambiente mobile. Empatica E4 conserva i dati sul dispositivo e i dati raccolti vengono successivamente caricati su un computer. Il sensore *Polar H7* e l'auricolare *MindWave* possono comunicare con un telefono mobile tramite Bluetooth *Low Energy (BLE)* per archiviare i dati.

Riassunto generale sul Dataset e sulla raccolta dei dati

Data collection summary	
Numero di partecipanti	32 (20 uomini e 12 donne)
Età dei partecipanti	Da 19 a 36 (mean = 23.8 anni, stdev. = 3.3 anni)
Durata della sessione	Totale 172.92 min, (mean = 10.8 min, stdev. = 1.04 min)
Categorie di annotazioni sulle emozioni	1 - 5: Arousal, Valence
	1 - 4: Cheerful, Happy, Angry, Nervous, Sad
	Choose one: Common BROMP affective categories + less common BROMP affective categories
Segnali fisiologici misurati	3-axis Acc. (32 Hz), BVP (64 Hz), EDA (4 Hz), heart rate (1 Hz), IBI (n/a), body temperature (4 Hz), EEG (8 band, 32 Hz), ECG (1 Hz)
Dataset contents	
Conversazioni audio	172.92 min (per 16 sessioni)
Conversazioni filmati	223.35 min (per 21 sessioni)
Segnali fisiologici	Riferirsi al Dataset contents subsection
Annotazioni delle emozioni (# ogni 5 sec)	**Self**: 4,159
	Partner: 4,159
	5 external observers: 20,803

L'insieme delle feature ricavate dai sensori è quindi il seguente:

1. *Attention*: parametro che misura il livello di attenzione dell'utente (varia da 1 a 100)
2. *Meditation*: parametro che misura il livello di rilassatezza dell'utente (varia da 1 a 100)
3. Onde celebrali: il dispositivo effettua un elettroencefalogramma dividendo la loro frequenza in 8 fasce:
   a) *Delta* (0.5-2.75 Hz)
   b) *Theta* (3.5-6.75 Hz)
   c) *Low-alpha* (7.5-9.25 Hz)
   d) *High-alpha* (10-11.75 Hz)
   e) *Low-beta* (13-16.75 Hz)
   f) *High-beta* (18-29.75 Hz)
   g) *Low-gamma* (31-39.75 Hz)
   h) *Middle-gamma* (41-49.75 Hz)

4. *X, Y, Z*: misura di un accelerometro all'interno del bracciale
5. *E4_BVP*: fotopletismografia
6. *E4_EDA*: misura dell'attività elettrodermica
7. *E4_HR*: frequenza del battito cardiaco
8. *E4_IBI*: intervallo di tempo fra un battito cardiaco ed un altro
9. *E4_TEMP*: misura della temperatura corporea

Capitolo 5

Risultati Sperimentali

5.1 Concetti utili per l'analisi dei risultati

Per ottenere una comprensione approfondita dei risultati conseguiti, di seguito vengono presentate le metriche e le tecniche usate per valutare le prestazioni dei modelli e per confrontare i dati.

5.1.1 Indici di precisione

Gli indici di precisione si riferiscono a misure o metriche che valutano quanto un modello sia accurato nelle sue previsioni o nelle sue classificazioni. Questi indici sono utilizzati per valutare le prestazioni di un modello in base ai dati di input e alle previsioni generate dal modello stesso.
Di seguito vengono presentate alcune delle principali metriche di precisione utilizzate per valutare i modelli:

- *Precision Score:* è una misura della precisione di un modello di classificazione una volta addestrato che rappresenta la frazione di previsioni corrette rispetto al totale delle previsioni positive fatte dal modello.

$$\boldsymbol{Precision} = \frac{TruePositivises}{TruePositivises + FalsePositivises}$$

  True Positives: sono i casi positivi correttamente previsti dal modello.
  False Positives: sono i casi negativi erroneamente previsti come positivi dal modello.

  La percentuale generale di previsioni corrette invece è stata calcolata tramite l'*accurancy* che invece rappresenta la frazione di previsioni corrette rispetto al totale delle previsioni fatte dal modello, senza la distinzione in casi positivi e negativi.

$$\boldsymbol{Accurancy} = \frac{TruePositivises\ +\ TrueNegatives}{TotalePredictions}$$

- *Recall:* è una misura della frazione di casi positivi effettivi che sono stati correttamente previsti dal modello. In altre parole, quanto il modello è in grado di individuare tutti i casi positivi.

$$\boldsymbol{Recall} = \frac{TruePositivises}{TruePositives\ +\ FalseNegatives}$$

- *F1-Score:* è una misura che combina le due metriche in un'unica misura: la precision score e recall. Questa metrica è spesso utilizzata nelle situazioni in cui è fondamentale trovare un equilibrio tra il minimizzare i falsi positivi (precisione) e il minimizzare i falsi negativi (recall).

$$\boldsymbol{F1 - Score} = \frac{Precision\ x\ Recall}{Precision\ +\ Recall}$$

Nel mio studio, mi sono limitato a calcolare l'accuratezza del modello addestrato. Questa misura, ottenuta mediante la funzione *score* sulle variabili di test, rappresenta la percentuale di previsioni corrette rispetto al totale delle previsioni fatte dal modello.

5.1.2 Coefficienti di correlazione

I coefficienti di correlazione in un modello indicano quanto due cose siano legate tra loro. Essi aiutano a capire come le feature si influenzano reciprocamente e sono importanti per fare previsioni basate su queste relazioni.

- *Correlazione di Pearson:* è una misura statistica utilizzata per valutare la relazione lineare tra due variabili continue. Fornisce un output compreso tra -1 e +1 dove: +1 indica una correlazione positiva perfetta, 0 indica mancanza di correlazione e -1 indica una correlazione negativa perfetta.
- *Correlazione di Kendall:* è un indice di correlazione non parametrico che valuta la concordanza tra le classificazioni delle variabili invece che la stretta relazione tra i loro valori. Questo coefficiente misura quanto le classificazioni delle variabili siano in accordo tra loro. Il suo valore oscilla tra -1 e 1, con 1 che rappresenta una perfetta correlazione positiva, indicando quindi un forte grado di concordanza tra le classificazioni delle variabili.
- *Correlazione di Spearman:* è un'altra misura di correlazione non parametrica che che è basata sui ranghi delle osservazioni anziché sui loro valori effettivi. Il suo valore varia da -1 a 1, con significato simile a quello della correlazione di Pearson.

Nel mio studio, ho utilizzato esclusivamente la correlazione di Pearson per valutare la similarità tra i vettori di feature importance ottenuti per le diverse etichette. Questo mi ha permesso di misurare quanto esse siano associate in modo lineare tra le diverse categorie di dati, fornendo informazioni sulle relazioni tra queste categorie.

5.2 Introduzione all'indagine

L'obiettivo principale di questa analisi è fornire una visione d'insieme completa dei risultati ottenuti. Partiremo da una panoramica sull'accuratezza dei vari algoritmi di classificazione. Successivamente esamineremo in dettaglio le feature importance ottenute attraverso l'algoritmo Decision Tree, riconosciuto anche dai nostri studi per la sua efficienza, per poi trovare una relazione tra le feature importance dei diversi target presi in considerazione.

Un passo cruciale sarà poi la valutazione delle feature importance ottenute tramite l'algoritmo BoCSoR, permettendo così un confronto diretto con quelle ottenute precedentemente. È importante notare che queste due metodologie differiscono nel loro approccio, in quanto una si basa su un algoritmo di classificazione, mentre per l'altra si è usato un algoritmo di regressione.

Infine, concluderemo analizzando e confrontando le prestazioni degli algoritmi di valutazione utilizzati. Questo ci permetterà di determinare quale di essi si adatta meglio ai nostri dati e alle nostre esigenze specifiche.

5.3 Scelta dell'algoritmo

Al fine di studiare il dataset contente le varie misurazioni, il processo è iniziato con la divisione dei dati in set di addestramento e di test, con una proporzione del 80% per il set di addestramento e

del 20% per il set di test. Il passo successivo ha coinvolto la ricerca di un modello di machine learning che mirasse a ottenere l'accuratezza massima possibile. I modelli sono stati allenati utilizzando il set di addestramento, e successivamente sono stati calcolati gli indici di precisione per valutare le loro performance.

5.3.1 Valutazione e selezione dei classificatori

Verrà ora riporta in primo luogo una tabella con i risultati per le diverse classificazioni, seguita da degli istogrammi con le misure di accuratezza dei classificatori.

Classificatore	**Accuratezza Train**	**Accuratezza Test**	**Precision Score**
MLP	0.8477	0.8500	0.9055
K-NN	0.9733	0.9513	0.9511
Decision Tree	**1.0**	**0.9997**	**0.9998**

Questo tipo di grafico a barre è utile per confrontare visivamente le prestazioni dei diversi classificatori su entrambe le metriche.

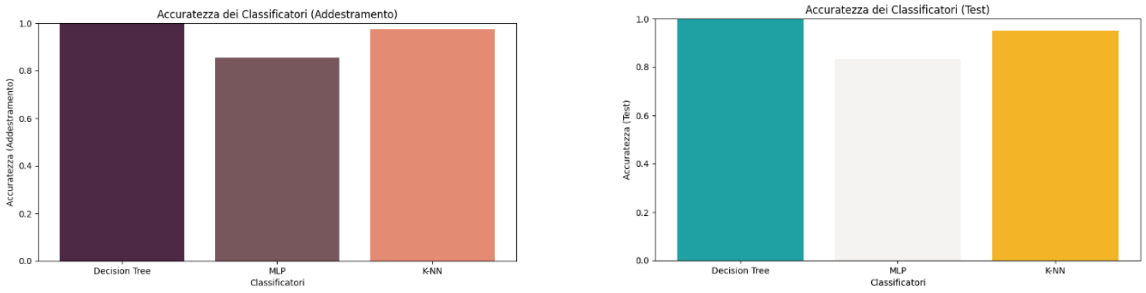


Figura 5.3.1: Differenza di accuratezza tra i classificatori utilizzati

In generale possiamo notare che tutti e tre i classificatori mostrano un'accuratezza di test notevolmente elevata, con punte superiori al 95%. Questo indica che tutti e tre i modelli hanno dimostrato una buona capacità predittiva sui dati di test.

I risultati più significativi però si possono vedere sul classificatore *Decision Tree*, ciò ci può suggerire che l'approccio basato su alberi decisionali sia ben adatto a questo tipo di problema.

5.4 Analisi delle feature importances

Dopo aver visto che i risultati più significativi vengono dal classificatore Decision Tree si va a produrre le feature importance per tanti decision tree quanto le label da predire: *external_arousal, external_valence, partner_arousal, partner_valence, self_arousal, self_valence*. Il tutto è stato fatto per andare a fare un'analisi del modello, così da identificare quali sono le feature più importanti e quali siano più o meno semplici da riconoscere ai fini del nostro esperimento.

Ciò è stato possibile grazie alla funzione *feature_importances_* che applicata sul modello addestrato restituisce un array contenente le feature importance di ciascuna variabile target.

Per fare una classificazione corretta, per ogni label si è preso le feature informative (*top 5*) e le feature non informative (*bottom 5*)
Confronti per ***Arousal***:

***external_arousal*:**

TOP 5	BOTTOM 5
E4_IBI: 0.3038506204526722	Meditation: 0.001318483160978129
E4_HR: 0.25558367118876574	y: 0.0009840689722824785
middleGamma: 0.11029335048818492	E4_BVP: 0.0009443607410342868
theta: 0.0893534451641983	lowAlpha: 0.0005543424775102027
E4_EDA: 0.0688458997100982	highAlpha: 0.00015151356286548774

***partener_arousal*:**

TOP 5	BOTTOM 5
E4_TEMP: 0.40833065693331655	Meditation: 0.0008989114249262359
E4_IBI: 0.20604488826915174	highAlpha: 0.0005788943256198921
E4_HR: 0.14290424758192588	E4_BVP: 0.00042101562465773895
x: 0.11153456840289816	lowGamma: 0.0001568305036896757
theta: 0.05894292236070961	lowBeta: 0.0

***self_arousal*:**

TOP 5	BOTTOM 5
E4_IBI: 0.4777012078068229	lowAlpha: 0.0
x: 0.15280176563386558	highAlpha: 0.0
delta: 0.12547955875708464	highBeta: 0.0
y: 0.0969878877658371	lowGamma: 0.0
E4_HR: 0.07304458405866147	theta: 0.0

Confronti per ***Valence***:

***external_valence*:**

TOP 5	BOTTOM 5
lowGamma: 0.28932170864319673	Attention: 0.0
lowBeta: 0.18502878189533742	highAlpha: 0.0
lowAlpha: 0.14514932881088088	highBeta: 0.0
E4_HR: 0.14175235061046915	middleGamma: 0.0
E4_IBI: 0.0737068111164296	Meditation: 0.0

***partner_valence*:**

TOP 5	BOTTOM 5
E4_IBI: 0.30451074429537606	Attention: 0.0016464291887690874
E4_TEMP: 0.25604541452991025	theta: 0.0002197727342549985
x: 0.09701270032020168	E4_BVP: 0.0
E4_HR: 0.08930443087720172	lowAlpha: 0.0
z: 0.0605824789561942	highBeta: 0.0

***self_valence*:**

TOP 5	BOTTOM 5
E4_HR: 0.2878254842530246	highBeta: 0.007198427733748468
E4_TEMP: 0.1529954978669433	Meditation: 0.0005373401739011171
E4_IBI: 0.1318424941799653	Attention: 0.0003851477811364211
z: 0.1016064037805378	E4_BVP: 4.730394738644882e-07
E4_EDA: 0.0735461896690359	highAlpha: 0.0

I risultati ottenuti dalla selezione delle feature informative e non informative ci dimostrano come le due feature ***E4_IBI*** ed ***E4_HR*** sono comuni tra le prime cinque feature informative per entrambe le etichette di *arousal* e *valence*. Questo ci suggerisce che le due caratteristiche possono avere un impatto significativo sulla previsione delle emozioni. Come già citato *E4_IBI* rappresenta l'Intervallo tra Battiti Cardiaci (Inter-Beat Interval), mentre *E4_HR* indica la Frequenza Cardiaca. Il fatto che siano tra le feature più informative suggerisce che i cambiamenti nei battiti cardiaci e nella frequenza cardiaca possono essere indicatori rilevanti per valutare l'intensità e la positività delle emozioni.
D'altra parte, notiamo che nelle prime cinque feature non informative ci sono delle etichette a comune. Ad esempio, ***highAlpha*** che non fornisce informazioni rilevanti per la previsione delle variabili target.

Per rafforzare queste conclusioni, risulta fondamentale ricercare studi scientifici nella letteratura medica e nelle neuroscienze che abbiano previamente investigato le correlazioni tra queste feature e le risposte emotive di arousal e valence. Possibili studi verranno proposti successivamente nelle conclusioni.

5.5 Correlazione di Pearson

Un'analisi approfondita è stata svolta poi, nello studio delle feature importance per le diverse variabili target. L'obiettivo principale era valutare la similarità tra i vettori di feature importance associati a ciascuna di queste etichette (figura 5.5.1).

Spieghiamo però prima in maniera dettagliata cosa fa la **correlazione di Pearson**. Essa è una misura statistica utilizzata per valutare la relazione lineare tra due variabili continue. Fornisce come output un valore compreso tra -1 e +1 dove:

- +1 indica una correlazione positiva perfetta, le due variabili aumentano insieme in modo lineare
- 0 indica una mancanza di correlazione, variabili non correlate linearmente
- -1 indica una correlazione negativa perfetta, le due variabili variano in direzione opposta in modo lineare

Se le due variabili, che nel nostro caso sono i vettori di feature importance, variano in direzione opposta in modo lineare vuol dire che hanno un andamento tra di loro inversamente proporzionale. Ad esempio, se consideriamo due vettori di feature importance, A e B, e la correlazione tra di loro è pari a -1, significa che, quando le feature rappresentate da A aumentano in importanza, le feature rappresentate da B diminuiscono in importanza in modo lineare, e viceversa. D'altra parte, quando la correlazione è vicina a +1 indica una correlazione positiva

perfetta, il che significa che i due vettori di feature importance aumentano insieme in modo lineare.

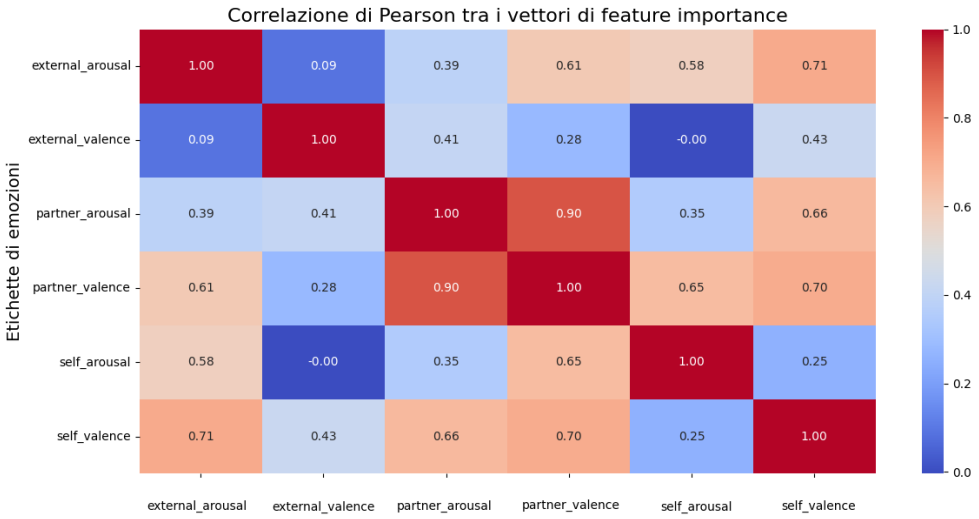


Figura 5.5.1: risultati ottenuti per la correlazione di Pearson

Non sussiste una particolare anomalia nell'osservare la massima similarità tra le feature di arousal e quelle di valence. Questi risultati (figura 5.5.2) suggeriscono che determinate caratteristiche possano essere rilevanti per entrambe le dimensioni dell'esperienza emotiva, indicando potenziali aspetti di sovrapposizione o interazione tra le due.

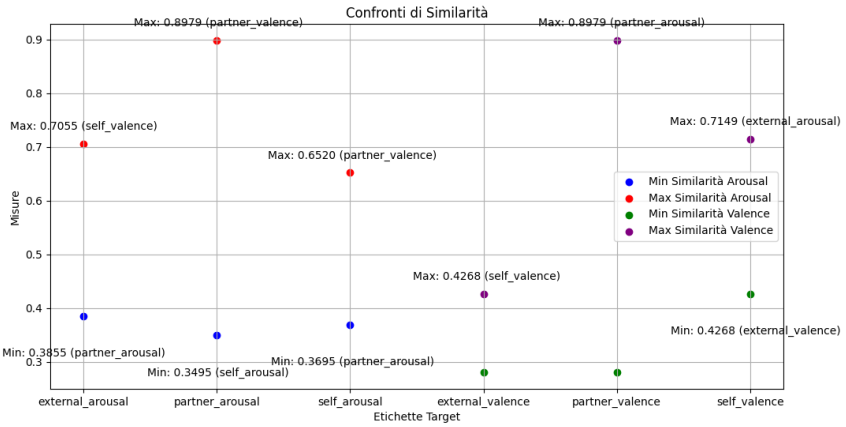


Figura 5.5.2: grafico a dispersione sulle similarità massime e minime per i diversi target

5.6 Valutazione feature importance: BoCSoR

Anche attraverso **BoCSoR** si è svolta un'analisi dettagliata delle feature importance. Per fare una classificazione corretta, per ogni label si sono, come nel caso precedente, prese le feature informative (*top 5*) e le feature non informative (*bottom 5*).

Confronti per ***Arousal***:

external_arousal:

TOP 5	BOTTOM 5
lowAlpha: 0.2107728337236534	highAlpha: 0.0
Delta: 0.20608899297423888	highBeta: 0.0
Theta: 0.1990632318501171	lowGamma: 0.0
x: 0.14988290398126464	middleGamma: 0.0
y: 0.10772833723653395	Meditation: 0.0

partener_arousal:

TOP 5	BOTTOM 5
lowBeta: 0.2227204783258595	highAlpha: 0.0
Delta: 0.21674140508221226	highBeta: 0.0
lowAlpha: 0.21674140508221226	lowGamma: 0.0
middleGamma: 0.21674140508221226	theta: 0.0
z: 0.12705530642750373	Meditation: 0.0

self_arousal:

TOP 5	BOTTOM 5
lowAlpha: 0.2568370986920333	highAlpha: 0.0
Delta: 0.2140309155766944	highBeta: 0.0
middleGamma: 0.2140309155766944	lowBeta: 0.0
E4_EDA: 0.15101070154577884	lowGamma: 0.0
E4_IBI: 0.08204518430439953	Meditation: 0.0

Confronti per ***Valence***:

external_valence:

TOP 5	BOTTOM 5
theta: 0.19619422572178477	E4_EDA: 0.0
E4_HR: 0. 17979002624671916	Attention: 0.0
lowAlpha: 0. 13123359580052493	highAlpha: 0.0
delta: 0. 13057742782152232	highBeta: 0.0
middleGamma: 0. 13057742782152232	Meditation: 0.0

partner_valence:

TOP 5	BOTTOM 5
lowAlpha: 0.5552995391705069	highBeta: 0.0
E4_HR: 0. 43548387096774194	lowGamma: 0.0
lowBeta: 0.009216589861751152	middleGamma: 0.0
	theta: 0.0
	Meditation: 0.0

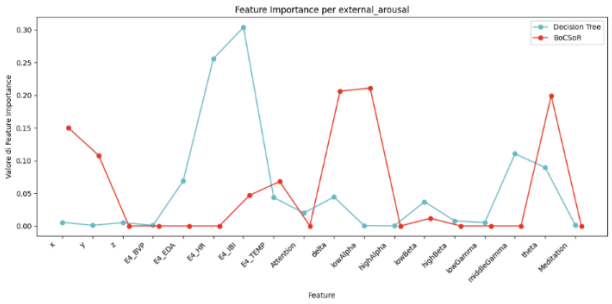
self_valence:

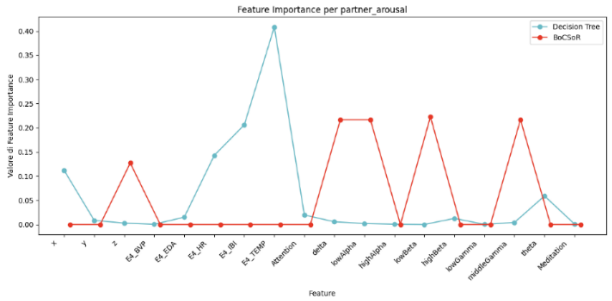
TOP 5	BOTTOM 5
delta: 0.2892561983471074	lowBeta: 0.0
theta: 0. 2892561983471074	highBeta: 0.0
E4_HR: 0. 2024793388429752	lowGamma: 0.0
x: 0. 10743801652892562	middleGamma: 0.0
E4_TEMP: 0. 07024793388429752	Meditation: 0.0

I risultati ottenuti ci dimostrano come per l'arousal, le caratteristiche di bassa frequenza come ***lowAlpha*** e ***delta*** hanno un forte impatto nelle previsioni, mentre feature come ***highAlpha***, ***highBeta***, ***lowGamma*** e ***Meditation*** risultano meno rilevanti. Per quanto riguarda la valence, ***E4_HR*** spicca come la caratteristica più influente, mentre ***highBeta*** e ***Meditation*** appaiono meno determinanti.

5.7 Confronto tra le feature importance

Una considerazione interessante, dopo aver fatto un ordinamento *top 5* e *bottom 5* dei i vari target e per i vari algoritmi, deriva dai seguenti grafici in cui ho condotto un'analisi comparativa tra le feature calcolate utilizzando i due diversi approcci: la funzione feature_importance_ sul **Decision Tree** e **BoCSoR**. Questo confronto è stato fondamentale per valutare come ciascun metodo attribuisca importanza alle diverse feature nei dati raccolti.





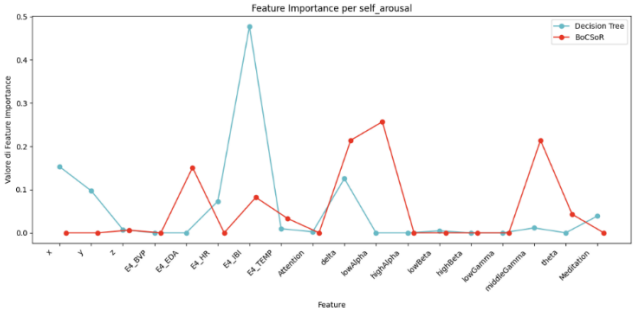


Figura 5.7.1: Confronti per Arousal

I risultati di questa analisi per l'arousal hanno evidenziato alcune differenze significative. Ad esempio, il Decision Tree ha attribuito un'alta importanza alle feature come ***E4_IBI*** e ***E4_HR***, mentro BoCSoR ha dato più rilevanza a feature come ***lowApha*** e ***delta***.

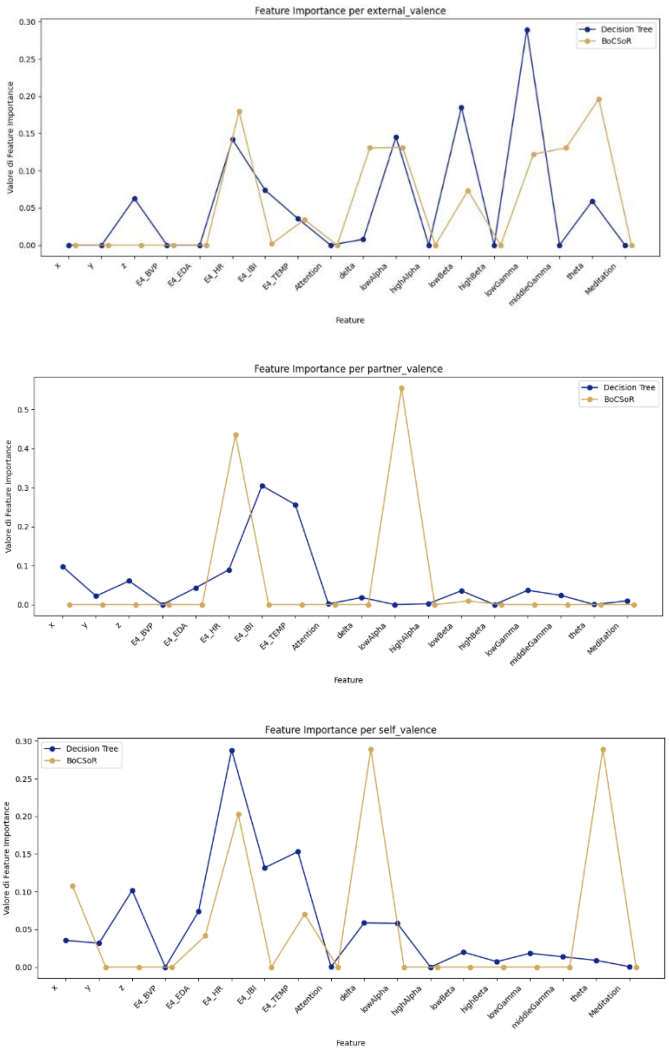


Figura 5.7.2: confronti per Valence

È interessante notare anche che per la valence, alcune feature sono state considerate importanti sia dal Decision Tree che da BoCSoR, ***E4_HR*** ne è un esempio evidente. Questo può indicare una convergenza nelle conclusioni tra i due metodo sulla rilevanza di determinare feature nella previsione della valence.

5.7.1 Algoritmi di valutazione

Ecco una panoramica dei risultati ottenuti mediante l'analisi dei tre diversi algoritmi: **Permutation Importance**, **SHAP** e **BoCSoR**. I risultati rilevano le differenze nelle feature importance attribuite ai diversi attributi per solo un target di riferimento *external_valence*.

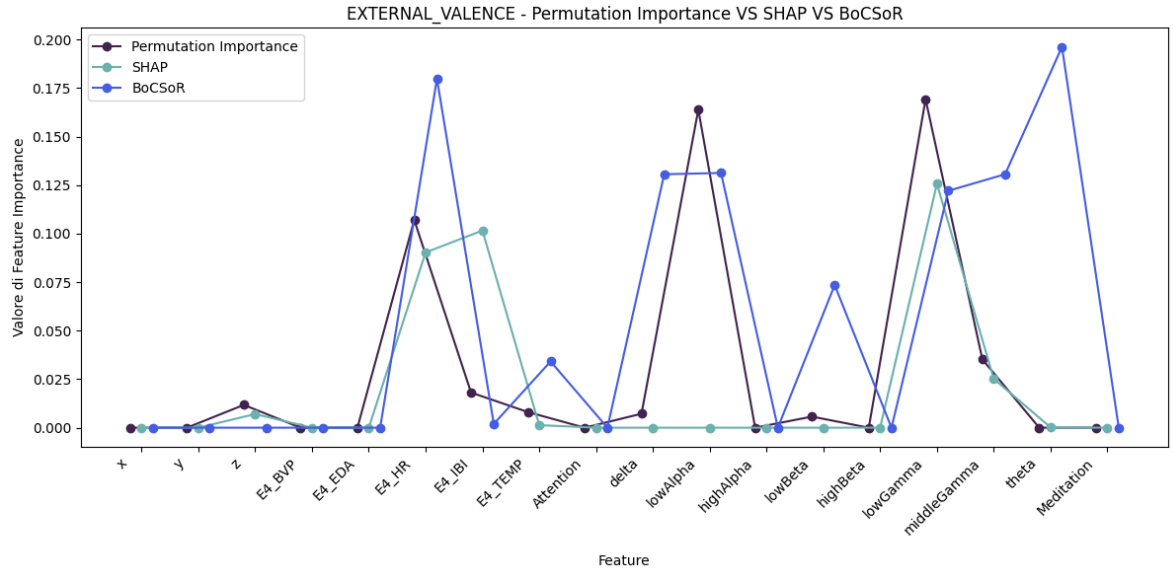


Figura 5.7.1

In particolare, tutti e tre gli algoritmi in generale mostrano valori simili tra di loro. Le feature importance calcolate con Permutation Importance e SHAP sono state valutate utilizzando il classificatore Decision Tree. Per quanto riguarda BoCSoR invece, ho usato l'algoritmo di regressione CatBostRegressor, già citato nei capitoli precedenti.

Va notato che BoCSoR ha mostrato risultati che, seppur leggermente differenziati dagli altri due algoritmi, sono comunque simili. Questa coerenza nei risultati potrebbe essere attribuita alla notevole accuratezza del modello di regressione CatBoostRegressor.
La sua capacità di predire le relazioni tra le feature è un indicatore della sua affidabilità nel contesto dell'analisi delle feature importance.

Questo confronto infine, è stato cruciale per comprendere meglio come le diverse metodologie di valutazione delle feature importance possono influenzare la selezione delle feature e, di conseguenza, la capacità predittiva del modello. Inoltre, ha messo in evidenza l'importanza di considerare attentamente quale approccio sia più adatto al contesto specifico di analisi dei dati emotivi.

Capitolo 6

Conclusioni

Dopo l'osservazione dei risultati ottenuti possiamo trarre le seguenti conclusioni con un buon grado di oggettività:

1. La valutazione delle prestazioni e dell'accuratezza ha mostrato che, nell'ambito del nostro dataset, i modelli basati su alberi decisionali sono quelli che si sono distinti per ottenere le migliori prestazioni.
2. L'analisi delle feature importance attraverso metodi come *SHAP*, *Permutation Importance*, *BoCSoR* e l'uso di funzioni come *feature_importance_* ci hanno fornito una comprensione approfondita delle caratteristiche chiave nei dati. L'impiego di una varietà di metodi ha ampliato un quadro più completo e generale consentendoci di trarre informazioni significative sul nostro dataset relativo all'esperimento.
3. Le varie analisi condotte sulle correlazioni tra i vari target ci hanno dimostrato come anche target appartenenti a categorie diverse (*arousal* e *valence*) possono avere similarità tra di loro.
4. Il confronto tra i vari algoritmi di valutazione ha portato a trovare una similarità tra essi. In particolare, BoCSoR ha dimostrato una notevole coerenza nei risultati ottenuti, con differenze minime rispetto agli altri metodi. Questa consistenza può essere attribuita alla robustezza del modello di regressione utilizzato.

Partiamo subito dal risultato finale: l'algoritmo di classificazione **Decision Tree** si è dimostrato il migliore tra quelli analizzati. Questo successo è attribuibile all'utilizzo degli alberi decisionali, come ampiamente dimostrato nei capitoli precedenti. Gli alberi decisionali sono noti per la loro facilità di utilizzo e per la loro capacità di addestrare il modello in modo efficiente in termini di tempo. Inoltre, questi modelli sono in grado di usare dati eterogenei ed essere robusti in caso di mancanza di dati. Tuttavia, è fondamentale sottolineare che, come evidenziato nel corso delle nostre analisi, i Decision Tree richiedono un'attenta regolazione poiché possono incorrere nell'*overfitting*, cioè nell'adattamento eccessivo ai dati di addestramento, compromettendo la loro capacità di generalizzazione su nuovi input. Pertanto, nonostante i numerosi vantaggi offerti da questo algoritmo, è essenziale adottare un approccio bilanciato e attento alla configurazione dei parametri al fine di sfruttarne appieno il potenziale.

Dalla nostra analisi abbiamo potuto in seguito constatare che le feature più importanti sono:

1. **E4_IBI**: misurazione dell'Intervallo Interbattito (IBI), che è il tempo trascorso tra due battiti cardiaci consecutivi.
2. **E4_HR**: misurazione della frequenza cardiaca.

Queste due feature provengono da una scrematura molto dettagliata che, come dicevo, è stata condotta da vari algoritmi.

Non è per niente casuale il fatto di aver trovato queste due feature molto legate alle emozioni umane. In uno studio recente condotto da Jos F. Brosschot, Julian F. Thayer su 33 individui, intitolato "Heart rate response is longer after negative emotions than after positive emotions", si

evidenzia come indipendentemente dalla reattività iniziale, le risposte cardiovascolari dopo emozioni negative tendono a persistere più a lungo rispetto a quelle dopo emozioni positive.

Spesso solo l'affetto negativo [la valence] ha effetti cardiovascolari (Shapiro et al., 2001, citato in Brosschot & Thayer, 2003).

In questo contesto, il nostro studio ha identificato una chiara connessione tra la valenza emotiva (*valence*) e la frequenza cardiaca (*HR*). Infatti, la valenza emotiva si è riferita alla qualità effettiva delle emozioni che in questo caso ha tenuto di conto della negatività delle emozioni invece, la frequenza cardiaca è stata utilizzata come misura dell'attivazione del sistema nervoso.

In sintesi, si può vedere tramite questo articolo di letteratura scientifica come la valence emotiva e la frequenza cardiaca siano largamente legate tra di loro. Questo collegamento nel nostro caso quindi ci porta a fare una considerazione: è normale trovare come feature importance più significativa **E4_HR**. Inoltre, va sottolineato che il sensore *Polar H7 Bluetooth Heart Rate Sensor* è stato di fondamentale importanza in questo contesto.

Ora, infatti, grazie all'aiuto della spiegabilità degli algoritmi (XAI), è opportuno considerare l'ulteriore miglioramento dei sensori cardiovascolari per ottenere risposte ancora più precise e significative sulle reazioni emotive umane nei futuri studi.

La chiara utilità della **XAI** nella mia tesi è emersa in modo evidente. Non solo ha contribuito a fornire una comprensione approfondita dei risultati ottenuti, ma ha anche reso più accessibile l'analisi dei dati complessi ad un vasto pubblico compresi coloro che non sono esperti in intelligenza artificiale, come abbiamo ad esempio potuto vedere con i *Case-Use*. Questo dimostra come la XAI possa essere una potente alleata per democratizzare il campo dell*'IA*, aiutando le persone a migliorare le loro competenze e ottenere una comprensione più profonda nei propri settori di interesse.

Bibliografia

[1] Bratman, G. N., Olvera-Alvarez, H. A., & Gross, J. J. (2021). The affective benefits of nature exposure. Social and Personality Psychology Compass. https://doi.org/10.1111/spc3.12630

[2] Zhang, Y., Weng, Y., & Lund, J. (2022). Applications of Explainable Artificial Intelligence in Diagnosis and Surgery. Diagnostics, 12(2), 237. https://doi.org/10.3390/diagnostics12020237

[3] Brosschot, J. F., & Thayer, J. F. (2003). Heart rate response is longer after negative emotions than after positive emotions. International Journal of Psychophysiology, 50(3), 181-188. https://doi.org/10.1016/S0167-8760(03)00146-6