**Dislexia-iCon**

**Un Sistema di Machine Learning per la Predizione e il supporto alla dislessia**

**Gruppo di lavoro:**

* Russo Nicola, 737220, [n.russo26@studenti.uniba.it](mailto:n.russo26@studenti.uniba.it)
* Squarcella-Gorgoglione Francesco Pio, 758467, [f.squarcellagorgo@studenti.uniba.it](mailto:f.squarcellagorgo@studenti.uniba.it)
* Troiano Francesco, 779566, [f.troiano7@studenti.uniba.it](mailto:f.troiano7@studenti.uniba.it)

**Link GitHub:**

<https://github.com/Russo888/Dislexia-iCon-Un-Sistema-di-Machine-Learning-per-la-Predizione-e-il-Supporto-alla-Dislessia.git>

AA 2024-2025

**Indice:**

[INTRODUZIONE](#Introduzione)

[1. DATASET](#Dataset)

[1.1. Descrizione dataset (dominio)](#Descrizionedatasetdominio)

[1.2 Osservazione grafica dei dati:](#Osservazionegraficadeidati)

[2. ONTOLOGIA](#Ontologie)

[2.1 Analisi Dominio:](#AnalisiDominio)

[2.1.1 Classi:](#Classi)

[2.1.2 Object property:](#Objectproperty)

[2.1.3 Data property:](#Dataproperty)

[2.1.4 Individuals:](#Individuals)

[2.2 Software per la realizzazione dell’Ontologia:](#SoftwareperlarealizzazionedellOntologia)

[3. QUERY](#Query)

[3.1 DL Query:](#DLQuery)

[3.2 OwlReady:](#OwlReady)

[4. APPRENDIMENTO SUPERVISIONATO](#Apprendimento_supervisionato)

[4.1 Decisioni di progetto:](#Decisionidiprogetto)

[4.2 Metriche di valutazione:](#Metrichedivalutazione)

[4.3 Selezione delle Feature:](#SelezionedelleFeature)

[4.4 Preprocessing dei Dati:](#PreprocessingdeiDati)

[4.4.1: Dummification](#Dummification)

[4.4.2: Smote](#SMOTE)

[4.5 Divisione dei Dati:](#DivisionedeiDati)

[4.6 K-Nearest Neighbors (KNN):](#KNN)

[4.6.1 Decisioni di progetto:](#KNNdecisionidiprogetto)

[4.6.2 Ottimizzazione numero di vicini:](#KNNOttimizzazionenumerodivicini)

[4.6.3 Addestramento e predizione:](#KNNAddestramentoepredizione)

[4.6.4 Valutazione Finale:](#KNNValutazioneFinale)

[4.7 Random Forest:](#RandomForest)

[4.7.1 Decisioni di progetto:](#RandomForestDecisionidiprogetto)

[4.7.2 Ottimizzazione scelta numero alberi decisionali:](#RandomForestOttimizzazionesceltanumero)

[4.7.3 Valutazione finale](#RandomForestValutazioneFinale)

[4.8 Support Vector machines:](#SVM)

[4.8.1. Decisioni di progetto](#SVMDecisionidiprogetto)

[4.8.2 Ottimizzazione parametro gamma:](#SVMottimizzazionegamma)

[4.8.3 Valutazione Finale:](#SVMValutazioneFinale)

[4.9 Neural Network:](#Neural_netwoek)

[4.9.1 Decisioni di progetto:](#NeuralNetworkDecisionidiprogetto)

[4.9.2 Creazione modello e addestramento:](#NeuralNetworkCreazioneaddestramento)

[4.9.3 Valutazione Finale:](#NeuralNetworkValutazioneFinale)

[4.10 Conclusioni](#Conclusioni):

[5 CLUSTERING:](#Clustering)

[5.1 Decisioni di progetto:](#clusteringDecisionidiprogetto)

[5.2 K-means:](#clusteringkmeans)

[5.3 Valutazione Finale:](#clusteringValutazioneFinale)

[6 CONCLUSIONE](#Conclusioni)

[6.1 Sviluppi futuri](#SVILUPPIFUTURI)

**Introduzione:**

Questo progetto nasce con l’idea di predire una diagnosi di spettro autistico sfruttando un dataset disponibile online. A tal fine, sono state implementate metodologie di apprendimento supervisionato e non supervisionato. Inoltre, è stata sviluppata anche un'ontologia di riferimento che permette di adottare un quadro formale della realtà oggetto di studio.

Il disturbo dello spettro autistico (ASD) è una condizione del neurosviluppo. Sfortunatamente, i tempi di attesa per una diagnosi di ASD sono lunghi e le procedure non sono convenienti. L'impatto economico dell'autismo e l'aumento del numero di casi di ASD in tutto il mondo rivela un urgente bisogno di sviluppare metodi di screening efficaci e facilmente implementabili. Pertanto, uno screening ASD efficiente in termini di tempo e accessibile è imminente per aiutare gli operatori sanitari e informare le persone se devono perseguire una diagnosi clinica formale.

**1. Dataset:**

Presentiamo un set di dati relativo allo screening della dislessia, raccolto da un esperimento condotto su un gruppo eterogeneo di soggetti. Questo dataset contiene caratteristiche influenti utilizzabili per l'analisi, in particolare per identificare tratti di dislessia e migliorare la classificazione dei casi. Il set di dati registra una serie di metriche comportamentali e cognitive che si sono dimostrate efficaci nel rilevare la dislessia.

**1.1. Descrizione dataset (dominio):**

Questo dataset si basa su una raccolta di 3644 soggetti. A ciascun partecipante è stato somministrato un test strutturato in 32 attività, ciascuna caratterizzata da specifiche misurazioni quantitative delle performance.

Il test include parametri come:

* **Clicks**: Numero di risposte fornite.
* **Hits**: Numero di risposte corrette.
* **Misses**: Numero di errori.
* **Score**: Punteggio ottenuto.
* **Accuracy**: Accuratezza delle risposte.
* **Missrate**: Tasso di errore.

Oltre a queste metriche, vengono raccolte informazioni individuali come **sesso, età, lingua madre e altre lingue parlate**.

Se un soggetto presenta determinate caratteristiche nei punteggi ottenuti durante il test, vi è un'alta probabilità che manifesti tratti riconducibili alla dislessia. Il campo "Dyslexia" del dataset indica la presenza o assenza di diagnosi di dislessia per ciascun individuo. Questo set di dati rappresenta una risorsa utile per lo studio dei pattern cognitivi e per migliorare gli strumenti di diagnosi della dislessia.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Feature | Type | Description | | age | Integer | Anni d’età | | gender | Binary (0 / 1) | Mascho o Femmina (1=M, 0=F) | | ethnicity | Integer (1 - 10) | Elenco delle etnie comuni in formato testo | | jaundice | Binary (0 / 1) | ***Born with jaundice***: Se il caso è nato con ittero (colorito giallastro) | | is\_autistic | Binary (0 / 1) | ***Is\_Autism:*** valore booleano che indica se il soggetto è affetto da autismo | | PDD\_parent | Binary (0 / 1) | ***Family member with ASD***: Se un membro della famiglia è affetto da autismo | | test\_compiler | Integer (1 - 4) | ***Who completing the test:*** 1) Family member, 2) Health Care Professional, 3) Self, 4) Others | | country\_of\_res | String | ***Country of residence:*** nazione di residenza del paziente | | used\_app\_before | Binary (0 / 1) | ***Used the screening app before***: Se l'utente ha utilizzato un'app di screening | | A1\_Score | Binary (0 / 1) | ***Question 1 Answer:***  Tuo/a figlio/a ti guarda quando chiami il suo nome? | | A2\_Score | Binary (0 / 1) | ***Question 2 Answer:***  È facile per te avere un contatto visivo con tuo/a figlio/a? | | A3\_Score | Binary (0 / 1) | ***Question 3 Answer:***  Tuo/a figlio/a indica per indicare ciò che vuole? (ad esempio un gioco che non riesce e a raggiungere) | | A4\_Score | Binary (0 / 1) | ***Question 4 Answer:***  Tuo/a figlio/a punta a condividere interesse con te? | | A5\_Score | Binary (0 / 1) | ***Question 5 Answer:***  Tuo/a figlio/a finge? (ad esempio prendersi cura delle bambole, parlare su un telefono giocattolo) | | A6\_Score | Binary (0 / 1) | ***Question 6 Answer:***  Tuo/a figlio/a segue dove stai guardando? | | A7\_Score | Binary (0 / 1) | ***Question 7 Answer:***  Se tu o qualcun altro nella famiglia è visibilmente agitato, tuo/a figlio/a mostra segni di voler confortarli? (ad esempio, accarezzando i capelli, abbracciandoi) | | A8\_Score | Binary (0 / 1) | ***Question 8 Answer:***  Tuo/a figlio/a mostra interesse a partecipare a giochi di gruppo con altri bambini? | | A9\_Score | Binary (0 / 1) | ***Question 9 Answer:***  Tuo/a figlio/a usa semplici gesti (ad esempio il saluto) | | A10\_Score | Binary (0 / 1) | ***Question 10 Answer:***  Tuo/a figlio/a fissa il nulla senza uno motivo apparente? | | screening\_score (result) | Integer (0-10) | ***Screening Score:***  Il punteggio finale ottenuto sulla base dell'algoritmo di punteggio del metodo di screening utilizzato. Questo è stato calcolato in modo automatizzato | | class/asd | Binary (0 / 1) | **class/asd**:  Con screening score ≤ 6 no tratti asd, con screening score > 6 tratti asd |  * 1. **Osservazione grafica dei dati:**   Abbiamo svolto delle osservazioni grafiche che ci permettessero di valutare la correlazione dei dati e soprattutto per capire in che modo la diagnosi fosse influenzata da essi. Qui di seguito riportiamo un graficoi che abbiamo realizzato in linguaggio python:   * Il grafico descrive la differenza nella media dello score tra chi è risultato dislessico e chi no. |

**2. Ontologia:**

Un’ontologia è un modello di conoscenza che consente di rappresentare in modo univoco le informazioni relative ad un determinato dominio. Durante l’analisi del dominio si indentificano i concetti principali, le relazioni e le proprietà che caratterizzano il dominio.

Successivamente questi concetti, relazioni e proprietà possono essere formalizzate mediante un linguaggio di rappresentazione formale come, ad esempio, OWL (Ontology Web Language).

**2.1 Analisi Dominio:**

Dato il set di dati utilizzato e le informazioni sugli attributi, abbiamo deciso di dividere gli attributi del dataset in:

* *”ciò che deve essere rappresentato”*: gli attributi fondamentali e cruciali che devono essere rappresentati per catturare le informazioni essenziali dal dataset. Nel contesto dell'autismo, include: persone, dislessia, difficoltà e prestazioni
* *”ciò che caratterizza ciò che deve essere rappresentato”*: le proprietà delle entità rappresentate, (nel caso delle perone ad esempio possono essere età, sesso...)
* *”ciò che può essere ricavato”*
* *”ciò che può essere scartato”*.

**2.1.1 Classi:**

Analizzando il dominio abbiamo deciso di rappresentare come classi dell’ontologia:

***• Persona***: La classe Paziente, rappresenta l’insieme dei pazienti sottoposti al test. A questa classe abbiamo associato diverse proprietà:

* ètà: rappresenta l’età del paziente al momento del test.
* genere: rappresenta il genere del paziente.
* linguaMadre: vera se paziente è inglese, falso altrimenti.
* conosceAltraLingua: vera se il paziente conosce altre lingue, falso altrimenti.

***• Prestazioni:*** La classe Prestazioni, rappresenta l’insieme delle prestazioni sulla base dei test svolti dai pazieni. A questa classe sono associati cinque proprietà:

* valoreAccuracy: Indica la percentuale di previsioni corrette rispetto al totale.
* valoreHits: Numero di previsioni corrette (classificazioni giuste).
* valoreMisses: Numero di previsioni errate (classificazioni sbagliate).
* valoreMissrate: Percentuale di errori rispetto al totale delle previsioni.
* valoreScore: Punteggio complessivo del modello basato su una metrica specifica.

***• Dislessia:*** rappresenta l’insieme dei pazienti che sono affetti dal disturbo della dislessia.

***•Difficoltà:***

Immagine che contiene testo, Carattere, schermata, design

Il contenuto generato dall'IA potrebbe non essere corretto.

**2.1.2 Object property:**

Una object property permette di mettere in relazione due individui, siano essi di classi distinte o della stessa classe.

Tra le classi abbiamo definito delle relazioni che rappresentano come queste interagiscono tra di loro. Le relazioni definite sono:

* ***haDifficoltà(Paziente) -> Difficoltà*** : permette di individuare le difficoltà svolte da un paziente.
* ***haPrestazione(Persona) ->Prestazione***: permette di individuare a prestazione di un paziente sulla base dei test effettuati.
* ***haDislessia(Test) ->Domanda***:Permette di capire se un paziente è affetto dal disturbo della dislessia o meno.

Immagine che contiene testo, Carattere, schermata, Blu elettrico

Il contenuto generato dall'IA potrebbe non essere corretto.

**2.1.3 Data property:**

Una data property permette di mettere in relazione un individuo con un valore di tipo primitivo.

Immagine che contiene testo, schermata, Carattere, numero

Il contenuto generato dall'IA potrebbe non essere corretto.

**2.1.4 Individuals:**

Per alcune entità si sono create delle istanze, ad esempio per individuare istanze della classe “persona” a cui attribuire una prestazione e le relazioni con essa. Individui inseriti nella nostra ontologia:

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

**2.2 Software per la realizzazione dell’Ontologia:**

L’ontologia è stata create mediante il software Protege, che è uno strumento per creare e gestire ontologie e che permette di definire concetti, classi e relazioni in ontologie basate su standard come OWL.

**3. Query:**

Successivamente sono state formulate delle query per interrogare l’ontologia.

Immagine che contiene testo, schermata, Carattere, numero

Il contenuto generato dall'IA potrebbe non essere corretto.**3.1 DL Query:**

Una query che restituisce tutti i pazienti con età maggiore di 8 anni, scritta in linguaggio DL (Descriptions logics), che è un linguaggio formale utilizzato principalmente per la modellazione concettuale e l'ontologia nelle discipline dell'intelligenza artificiale e della rappresentazione della conoscenza.

**3.2 OwlReady:**

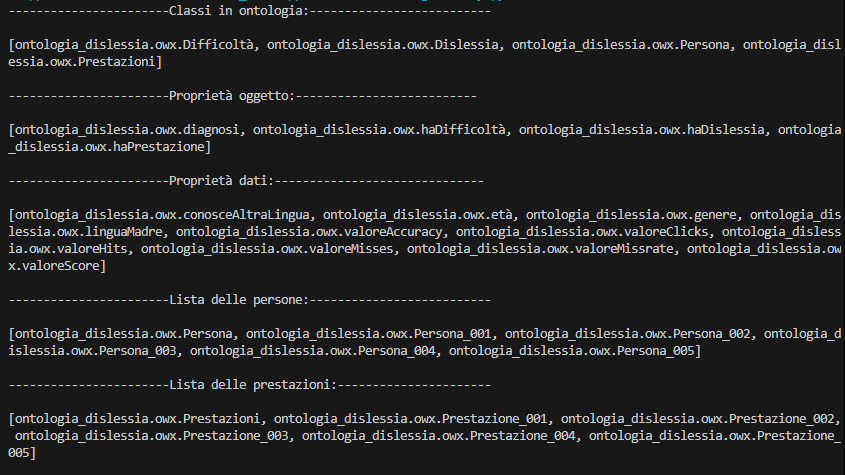
È stato poi creato il file “query.py” con il quale è possibile consultare l’ontologia direttamente in Python grazie alla libreria OwlReady2 per la manipolazione di ontologie e il ragionamento.

* Con il seguente codice è quindi possibile estrarre dall’ontologia la lista delle classi, delle object property, dei data property e degli individui che appartengono alle relative classi.

Immagine che contiene testo, schermata, Carattere, software

Il contenuto generato dall'IA potrebbe non essere corretto.

Il codice precedente produce i seguenti risultati:



* Attraverso il seguente codice è stato invece possibile interrogare l’ontologia.

Immagine che contiene testo, schermata, software, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

Il codice precedente restituisce come risultato la lista delle persone che hanno effettuato una prenotazione e la lista delle persone dislessiche

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

**4. Apprendimento supervisionato:**

L'apprendimento supervisionato è una tecnica di apprendimento automatico che mira a istruire un sistema informatico in modo da consentirgli di elaborare automaticamente previsioni sui valori di uscita di un sistema rispetto ad un input sulla base di una serie di esempi ideali, costituiti da coppie di input e di output, che gli vengono inizialmente forniti.

**4.1 Decisioni di progetto:**

In questo progetto, l'obiettivo dell'apprendimento supervisionato è risolvere un problema di classificazione, in particolare utilizzando la colonna 'Class/ASD' come variabile target da predire basandosi sulle altre caratteristiche relative ai dati dei pazienti.

Inizialmente è stato necessario decidere quali modelli utilizzare. Sono stati scelti gli algoritmi con le migliori prestazioni, in quanto la sensibilità del tema necessitava una grande precisione nelle valutazioni. Per questo motivo è stato scelto ad esempio random forest piuttosto che decision tree, in quanto è stata preferita la precisione del random forest alla minore complessità del decision tree, che potrebbe garantire prestazioni inferiori nel caso, ad esempio, in cui l'albero diventa più profondo e complesso.

Sono stati scelti 4 altri modelli (implementati con le librerie *‘sklearn’* e *‘keras’*) e ne è stata valutata l'efficacia e l'efficienza:

* **K-nearest-neighbors (KNN)**
* **Random Forest**
* **Support Vector Machine (SVM)**
* **Neural Network**

**4.2 Metriche di valutazione:**

Andiamo adesso ad analizzare le metriche con cui sono stati valutati i modelli ed i risultati. Per ogni algoritmo implementato sono statati prodotti 4 tipologie di grafici differenti:

* **ROC Curve (Receiver Operating Characteristic Curve):** La curva ROC è un grafico in cui l'asse delle ordinate rappresenta il tasso di veri positivi (True Positive Rate), mentre l'asse delle ascisse rappresenta il tasso di falsi positivi (False Positive Rate).
* **Precision-Recall Curve:** Questa curva rappresenta il trade-off tra la precisione e il recall di un modello di classificazione binaria.
* **Bar Chart di Varianza e Deviazione Standard:** Questo tipo di grafico a barre rappresenta la varianza e la deviazione standard dei punteggi di cross-validation. La varianza e la deviazione standard aiutano a comprendere quanto i risultati siano consistenti o variabili su diverse iterazioni della cross-validation.
* **Matrice di Confusione:** La matrice di confusione è una tabella che mostra il numero di previsioni corrette e errate fatte da un modello di classificazione in termini di veri positivi (TP), falsi positivi (FP), veri negativi (TN) e falsi negativi (FN). .

La **cross-validation** è un’altra tecnica utilizzata per valutare le prestazioni dei modelli in modo accurato e affidabile. Abbiamo scelto di dividere i dati in 5 "fold" uguali, in modo da addestrare e testare il modello 5 volte, ognuna delle quali usa una fold diversa come set di test e l'unione delle rimanenti come set di addestramento.

**4.3 Selezione delle Feature:**

Nella selezione delle feature abbiamo scelto di selezionare:

* Le risposte al questionario *(‘A1\_Score’, ‘A2\_Score’, ..., ‘A10\_Score’*) sono le nostre feature principali, in quanto rappresentano il metro di giudizio più evidente.
* Età (*‘age’*) e Genere (*‘gender’*) sono feature che potrebbero essere rilevanti, poiché l'età e il genere possono influenzare la prevalenza e la manifestazione dell'autismo.
* Il punteggio di Screening (*‘screening\_score’*), al pari delle risposte al questionario, è una delle feauture principali in quanto rappresenta la somma delle risposte al questionario.
* *‘PDD\_parent’* può essere un’altra feature importante per comprendere la trasmissibilità dello spettro autistico, quindi come un genitore affetto da autismo possa influenzare la possibilità che questa diagnosi si ripercuota sul proprio figlio.

*‘Ethnicity’, ‘contry\_of\_res’, ‘used\_app\_before’* e *‘test\_compiler’* sono i campi del nostro dataset che non sono stati inseriti tra le nostre feature in quanto nonostante potessero avere un impatto sulla diagnosi, non le abbiamo ritenute rilevanti per l’addestramento del modello a tal punto da considerarle feature.

**4.4 Preprocessing dei Dati:**

L’addestramento di ogni modello inizia con l’esecuzione del preprocessing dei dati.

**4.4.1 : Dummification**

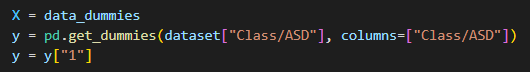
Questa operazione include la normalizzazione delle feature attraverso la funzione *‘get\_dummies()’* per eseguire la codifica one-hot delle variabili categoriche in quanto questi modelli non supportano questo tipo di variabili.

La codifica one-hot è un processo che converte le variabili categoriche in una rappresentazione binaria, in modo che ciascuna categoria venga rappresentata come una colonna binaria separata



Dato che il target della nostra predizione è Class/ASD, la colonna relativa a questo campo viene eliminata dal dataframe con il metodo *‘drop()’*.

La colonna Class/ASD viene inserita invece in una nuova variabile.



**4.4.2**: **SMOTE**

È stata utilizzata la funzione «SMOTE», implementata grazie all’uso della libreria imblearn, con lo scopo di ridimensionare la classe di esempi. SMOTE è utile quando si lavora con dataset sbilanciati, dove una classe è rappresentata in modo significativamente minore rispetto all'altra classe.



Applichiamo quindi la tecnica SMOTE ai parametri `X` e `y` che rappresentano le feature e il target. X1 conterrà le nuove feature sintetiche create da SMOTE per bilanciare le classi e y1 conterrà le corrispondenti etichette target.

**4.5 Divisione dei Dati:**



Nel contesto del problema specifico, abbiamo scelto di suddividere il nostro dataset in 75% per l’addestramento, e il restante 25% per il test. Questa suddivisione è sembrata un compromesso ragionevole tra la necessità di avere abbastanza dati per l'addestramento e la necessità di avere abbastanza dati nel set di test per valutare accuratamente le prestazioni del modello.

Se avessimo scelto una proporzione più alta per il set di test, avremmo avuto meno dati disponibili per l'addestramento del modello, ciò avrebbe potuto comportare una minore capacità del modello di apprendere le complessità dei dati e avrebbe potuto portare a una generalizzazione meno accurata.

D'altro canto, se avessimo scelto una proporzione più alta per il set di addestramento, saremmo potuti incorrere in un rischio di overfitting, acquisendo anche il rumore nei dati e risultando meno in grado di generalizzare su nuovi dati.

**4.6 K-Nearest Neighbors (KNN):**

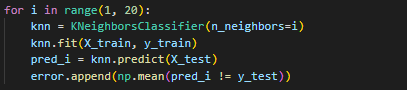
Il KNN è un algoritmo di classificazione che mira a determinare la classe di appartenenza di un dato di input cercando tra tutti gli esempi di addestramento quello più vicino al dato di input in base alla metrica desiderata, come ad esempio la distanza euclidea.

**4.6.1 Decisioni di progetto:**

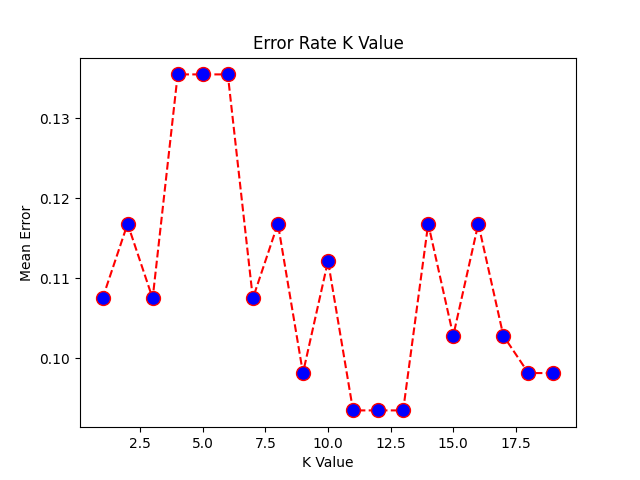
Questo modello è stato scelto perché semplice da implementare, versatile, in quanto funziona bene con dataset di qualsiasi dimensione, e robusto anche in presenza di dati rumorosi o valori anomali. Dato che il modello si basa sui vicini più prossimi, un singolo valore anomalo avrà meno impatto sull'output finale.

**4.6.2 Ottimizzazione numero di vicini:**

L'obiettivo di questo codice è quello di valutare come l'errore varia al variare del numero di vicini utilizzati in un range di valutazione da 1 a 20. Si sta cercando il valore ottimale di numero di vicini che minimizza l'errore di previsione e migliora la capacità di generalizzazione del modello.



L'obiettivo è scegliere il valore di k che equilibra la complessità del modello con la sua capacità di generalizzazione.



In questo caso l’errore è più basso quando k è tra 11 e 13 perciò un k intorno a 11 potrebbe essere una scelta ragionevole per ottenere un buon equilibrio tra overfitting e underfitting, dato che un k troppo piccolo potrebbe causare overfitting, mentre un k troppo grande potrebbe causare underfitting.

**4.6.3 Addestramento e predizione:**

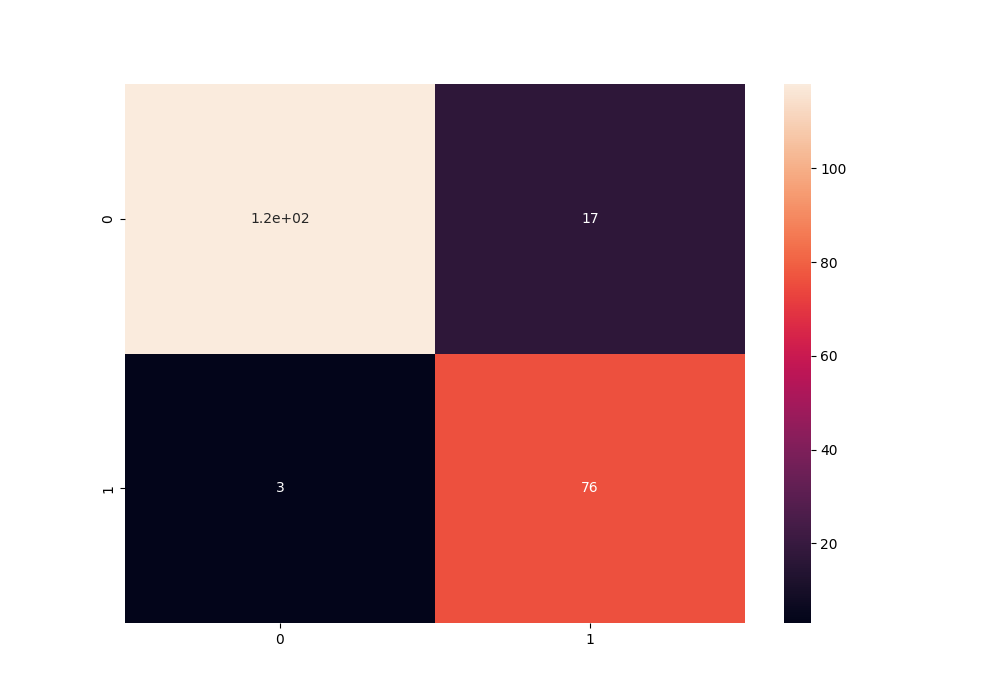
Scelto il numero di vicini, inizia l’addestramento del modello. Durante questo processo, il modello regola i suoi parametri in modo che le previsioni siano il più vicine possibile ai valori reali.



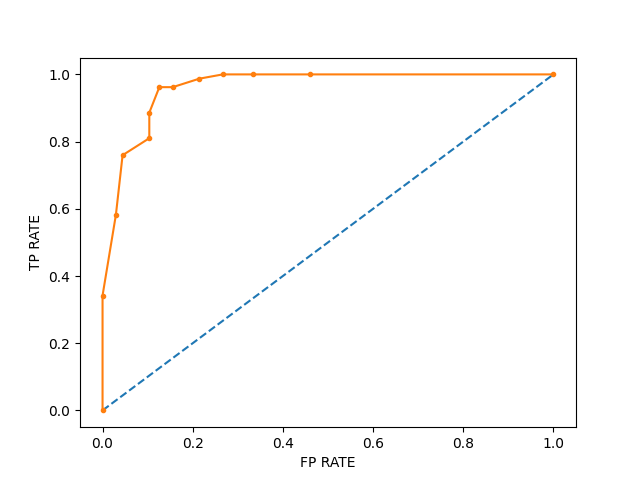
Il modello viene addestrato cercando di trovare la relazione tra le feature e le etichette di classe. Il codice crea quindi un modello di classificazione addestrato pronto per essere utilizzato per fare previsioni su nuovi dati, cercando di assegnare loro l'etichetta di classe appropriata in base alle sue vicinanze rispetto ai dati di addestramento.

Inserimento dell'immagine in corso...

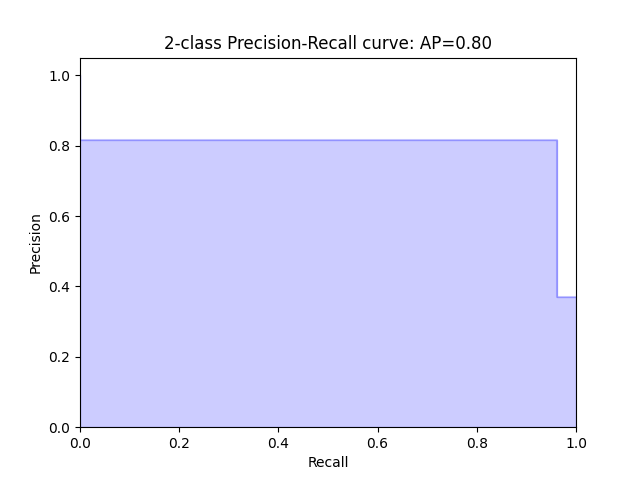
Il classificatore k-NN addestrato viene quindi usato per effettuare previsioni sul set di dati di test. Il metodo `predict` restituisce un array contenente i valori delle previsioni, che vengono confrontate con le etichette di classe effettive per calcolare il valore dell’accuracy,

**4.6.4 Valutazione Finale:**

**Matrice di Confusione:** rivela che il modello ha una buona capacità di fare previsioni corrette per entrambe le classi. Si nota che i falsi positivi (17) sono leggermente più elevati dei falsi negativi (3), ciò potrebbe suggerire che il modello è leggermente incline a effettuare previsioni corrette quando la classe reale è negativa.



**ROC Curve:** L'Area Under the Curve (AUC) nella ROC Curve è estremamente positiva, con un valore di 0.962. Questo indica che il modello KNN ha un'eccellente capacità di discriminare tra le classi positive e negative**.**



**Precision-Recall Curve:** Il grafico mostra delle imperfezioni evidenti dall’analisi delle metriche di precision, recall. La precisione relativamente bassa potrebbe significare che il modello sta etichettando erroneamente alcuni campioni.

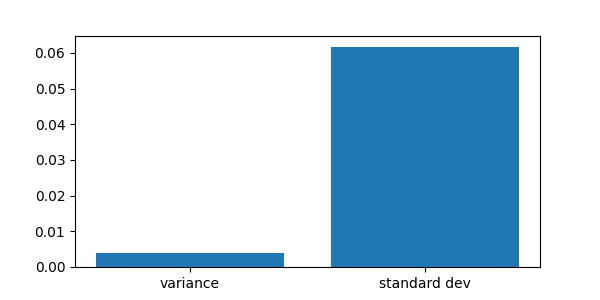
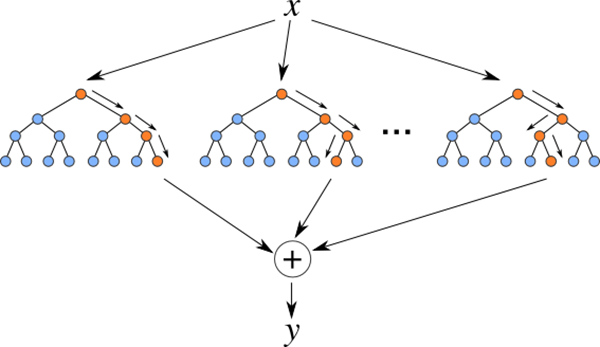
**Bar Chart di Varianza e Deviazione Standard:** La bassa varianza nelle iterazioni di cross-validation indica che il modello è stabile e coerente nelle sue prestazioni. Inoltre, la deviazione standard suggerisce che il modello non è soggetto a fluttuazioni significative nelle performance durante le diverse iterazioni. Questo è un indicatore positivo della coerenza del modello e della sua capacità di mantenere risultati affidabili.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

**4.7 Random Forest:**

L'algoritmo Random Forest rappresenta un tipo di modello che combina molti alberi decisionali in un unico modello. Individualmente, le previsioni fatte dagli alberi decisionali potrebbero non essere accurate, ma combinate insieme, le previsioni saranno in media più vicine al risultato.



**4.7.1 Decisioni di progetto:**

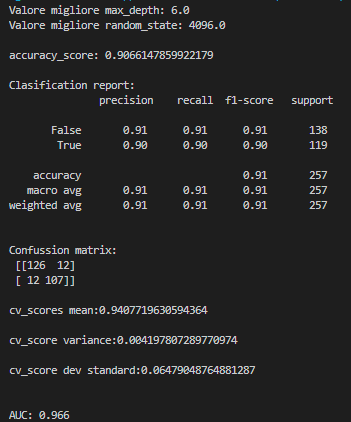
L'algoritmo Random Forest è stato scelto poichè tende ad avere una buona precisione e stabilità nei problemi di classificazione. Inoltre, la sua capacità di addestrare gli alberi in modo parallelizzato consente una rapida esecuzione anche con risorse computazionali limitate.

* ***‘n\_estimators=20’***: si è scelto di impostare a 20 questo parametro, che indica il numero di alberi decisionali che vengono creati nell'ensemble, in modo da trovare un equilibrio tra prestazioni e efficienza computazionale, dato che un numero maggiore di alberi può portare a prestazioni migliori, ma aumenta anche il tempo di addestramento.
* ***‘max\_depth=30’:*** questo parametro indica la profondità massima degli alberi decisionali all'interno dell'ensemble. È stato scelto 30 in quanto è un valore che permette agli alberi di adattarsi ai dati diminuendo il rischio di overfitting che si avrebbe avuto con una maggiore profondità.
* ***‘random\_state=0’***: questo parametro controlla la generazione dei numeri casuali all'interno del modello. Fornendo un valore specifico ci si assicura che l'addestramento e la previsione del modello siano riproducibili.

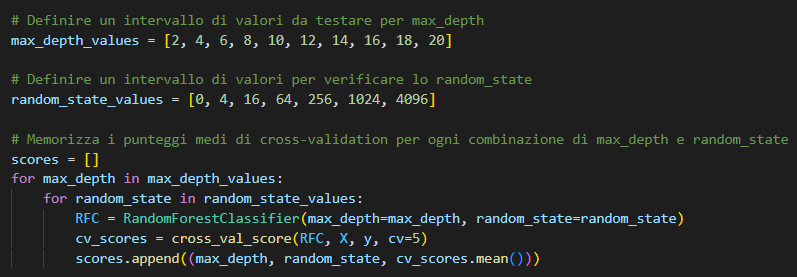


**4.7.2 Ottimizzazione scelta numero alberi decisionali:**

L'esecuzione del modello con i parametri impostati come in precedenza ha portato ad ottenere i seguenti risultati:



Abbiamo poi capito che scegliendo i valori più adatti per le variabili *‘max\_depth’* e *‘random\_state’*, i risultati, molto validi già di per sé, potevano essere migliorati ulteriormente. Per questo abbiamo deciso di usare un algoritmo che ci aiutasse proprio a scegliere questi valori.

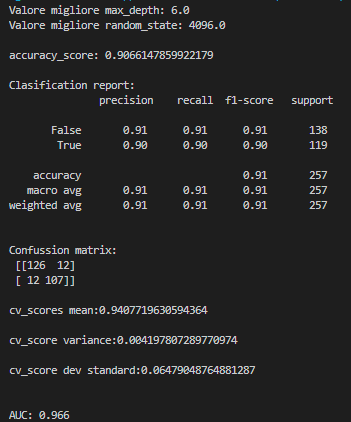
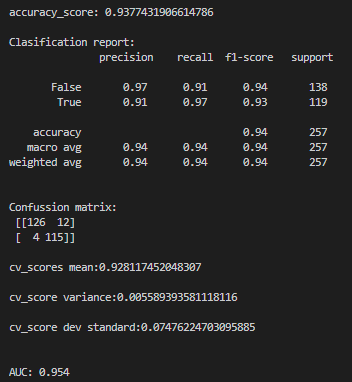


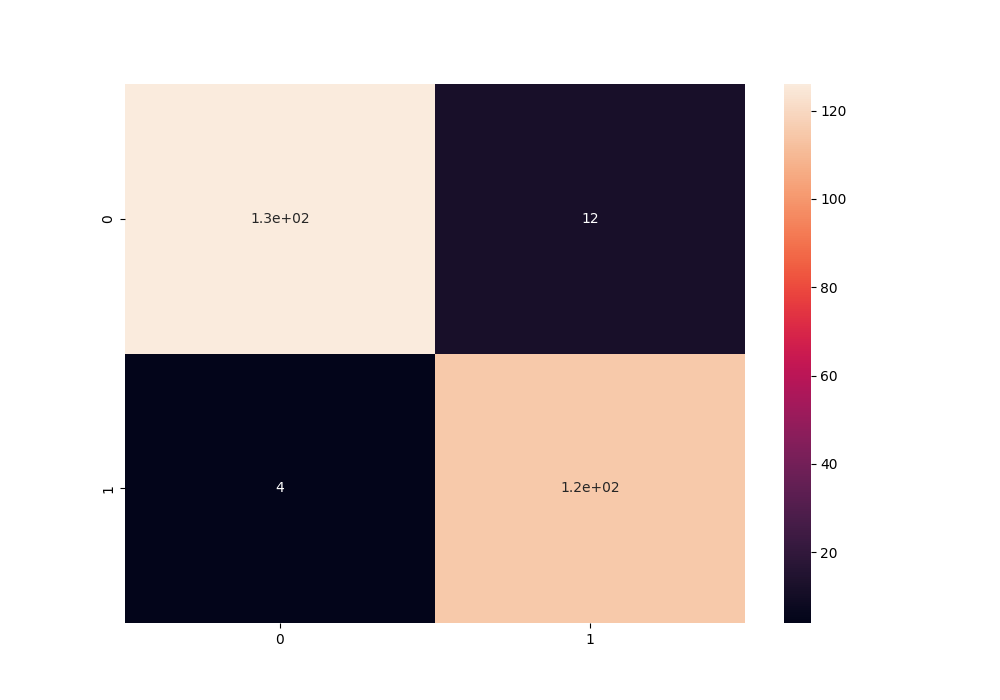
L'algoritmo parte definendo un intervallo di valori da testare per max\_depth e random\_state. Viene quindi eseguita una valutazione di cross-validation su ogni combinazione di valori di ‘max\_depth’ e ‘random\_state' e viene calcolata la media delle valutazioni ottenute, scegliendo la combinazione che restituisce i risultati migliori.



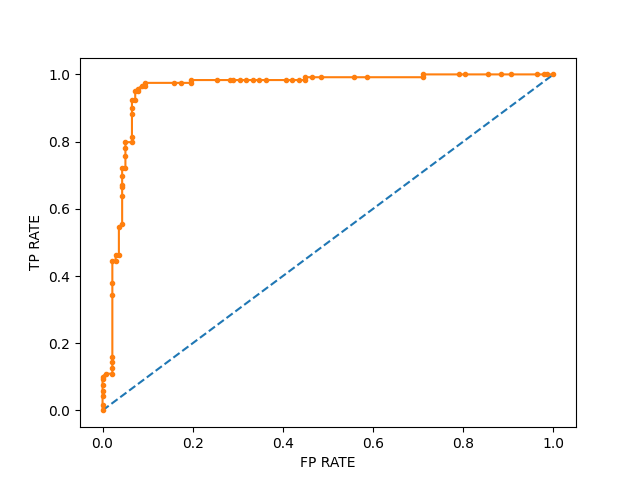
Infine, viene addestrato un classificatore random forest, utilizzando i valori di *‘max\_depth’* e *‘random\_state’* appena ottenuti, e viene eseguita una valutazione di cross-validation sul nuovo classificatore.

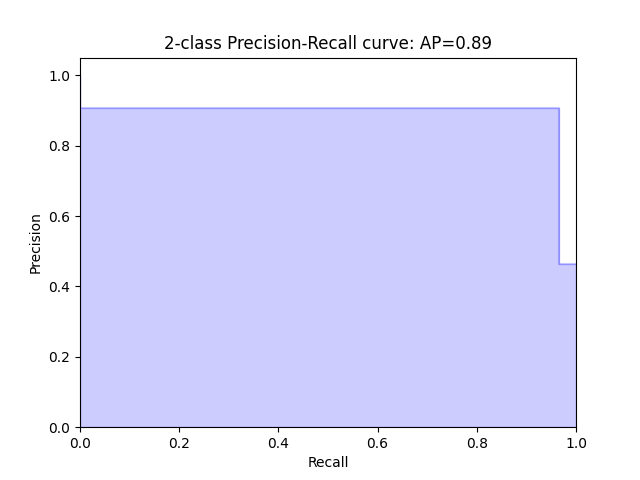
L'immagine a sinistra riporta i valori precedenti all’uso dell’algoritmo, mentre quella a destra mostra i valori ottimizzati.

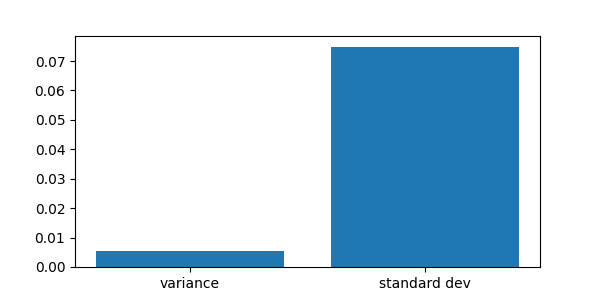
**4.7.3 Valutazione Finale:**

**Matrice di confusione**: mostra che ci sono pochi falsi positivi, solo 12 a fronte di 126 classificazione corrette, e ancor meno falsi negativi, solo 4 su un totale di 119 previsioni, il che indica un'eccellente capacità del modello nel limitare gli errori di classificazione.

**ROC curve:** L'Area Under the Curve (AUC) indica una forte capacità del modello di discriminare tra le classi. Questo punteggio suggerisce che il modello ha una buona capacità di separazione delle classi e un'ottima performance complessiva.



**Precision-recall curve**: l'analisi della precision-recall curve presenta dei valori molto bilanciati. Questi risultati indicano che il modello è abile a prevedere le classificazioni, suggerendo che la quasi totalità delle previsioni sia stata corretta.



**Bar Chart di Varianza e Deviazione Standard:** l'analisi delle metriche di validazione incrociata (cv\_scores mean, cv\_score variance, cv\_score dev standard) suggerisce che il modello mantiene un livello di coerenza nelle prestazioni quando viene applicato a diverse parti del dataset. La bassa varianza e la deviazione standard delle cv\_scores indicano una certa stabilità nelle prestazioni.

**4.8 Support Vector Machines (SVM):**

L’SVM è basato sull’idea di trovare un iperpiano che divida al meglio un set di dati in due classi, i Support Vector sono i punti dati più vicini all’iperpiano e tali punti dipendono dal set di dati che si sta analizzando.

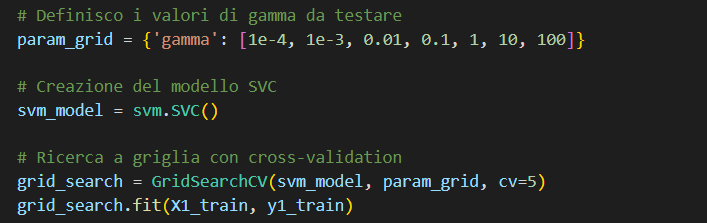
**4.8.1 Decisioni di progetto:**

Questo modello è stato scelto in quanto potente per la classificazione binaria, in particolare quando le classi sono ben separate. È adatto per problemi con un numero moderato di feature e può gestire relazioni non-lineari attraverso l'uso di kernel.

* Abbiamo scelto di usare il **kernel RBF** (Radial Basis Function) per questo modello perché con il kernel RBF è possibile comprendere strutture complesse e non lineari che potrebbero essere presenti nei dati, migliorando le probabilità di ottenere un modello di classificazione più accurato e generalizzabile.

**4.8.2 Ottimizzazione parametro gamma:**

L'iperparametro gamma nel kernel RBF controlla quanto l'influenza di un singolo esempio di addestramento si estenda. Valori più alti di gamma rendono le decisioni più locali, mentre valori più bassi danno una maggiore influenza alle istanze circostanti. L'obiettivo del seguente codice è quello di trovare il miglior valore per il parametro gamma.



Per fare ciò, si utilizza la tecnica della "grid search" insieme alla cross-validation. Si inizia definendo una griglia di valori di gamma, e successivamente, dopo aver creato un modello SVM, si esegue la ricerca a griglia che addestrerà e valuterà il modello su diverse combinazioni di parametri gamma utilizzando la cross-validation con 5 fold.

Infine, con il metodo fit, si otterrà il modello SVM ottimizzato con i migliori parametri gamma individuati dalla grid search. Il risultato del codice è il seguente:

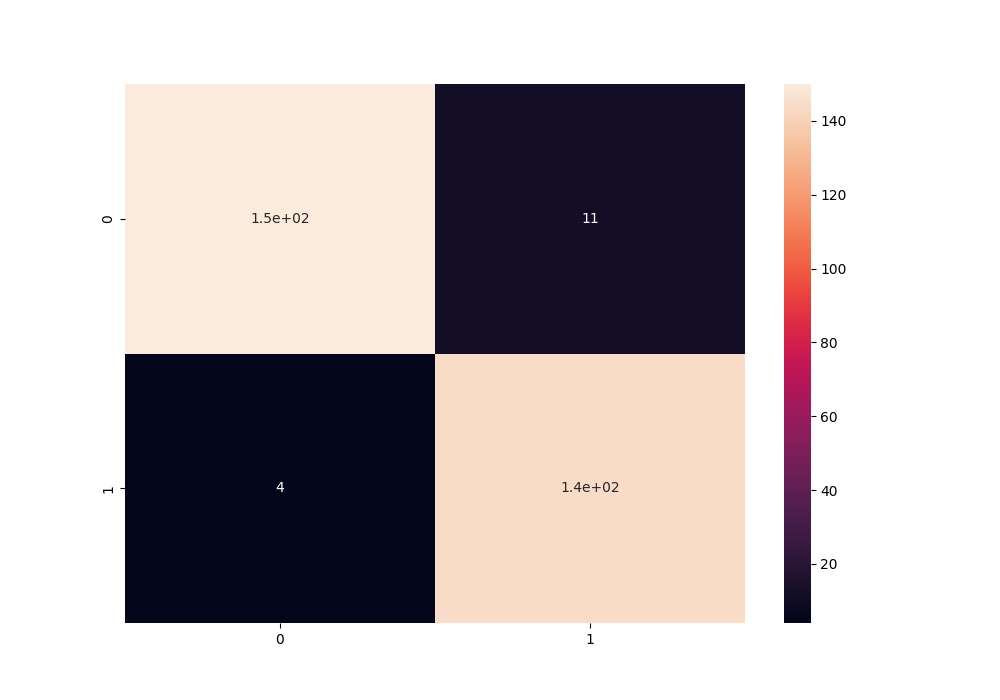
Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

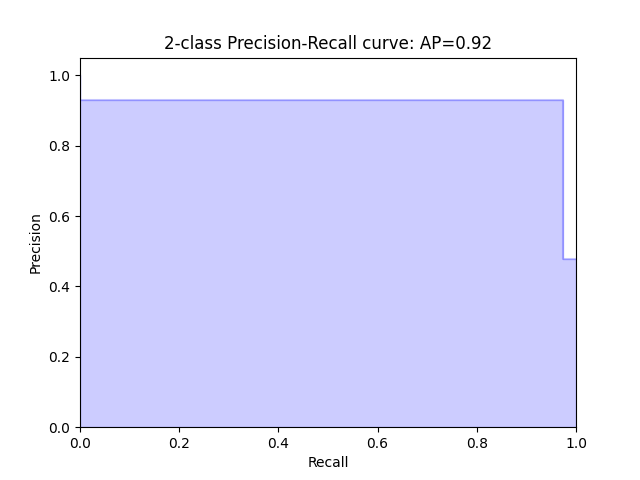
Questi risultati ci fanno capire come il modello risulta ottimizzato con il valore di gamma impostato a 0.01.



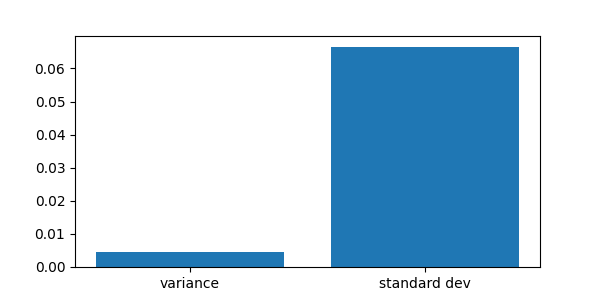
**4.8.3 Valutazione Finale:**



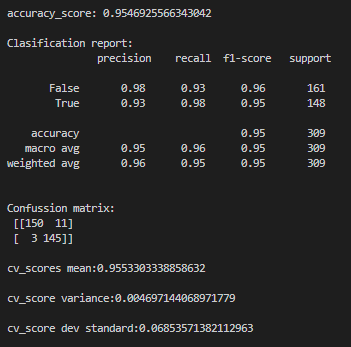
**Matrice di Confusione:** mostra come il modello abbia effettuato nel complesso delle ottime classificazioni, nonostante qualche errore, giustificabile in parte però dalla quantità di previsioni effettuate.



**Precision-Recall Curve:** il grafico mostra un ottimo bilanciamentotra precision e recall, confermando l’affidabilità delle previsioni.



**Bar Chart di Varianza e Deviazione Standard:** i risultati evidenziano una bassa varianza e una leggermente maggiore deviazione standard. Ciò suggerisce coerenza nelle prestazioni medie del modello, ma con alcune variazioni significative in alcuni fold.



**4.9 Neural Network**

Le reti neurali artificiali simulano il funzionamento dei neuroni biologici attraverso neuroni artificiali collegati tra loro. Questi neuroni elaborano gli input assegnando loro pesi, e la loro attivazione dipende dal superamento di una soglia prestabilita.

* I pesi determinano l'importanza di ciascun input nel processo decisionale.
* L'attivazione avviene se il valore elaborato supera una soglia definita.

Questo modello è stato scelto per la sua capacità di affrontare problemi complessi di apprendimento e per l'elevata accuratezza nelle previsioni.

**4.9.1 Decisioni di Progetto**  
Il modello di rete neurale artificiale è stato scelto per la sua capacità di affrontare problemi complessi di apprendimento e per l'elevata accuratezza nelle previsioni. Abbiamo optato per una rete neurale sequenziale con due livelli principali:

* **Primo livello (Hidden Layer):** 30 neuroni nascosti con funzione di attivazione ReLU, che introduce non-linearità e migliora la capacità di apprendimento del modello senza causare overfitting.
* **Secondo livello (Output Layer):** un singolo neurone con funzione di attivazione sigmoide, che permette di ottenere un output compreso tra 0 e 1 per la classificazione binaria.

Questa configurazione consente al modello di apprendere in maniera efficace, distinguendo con precisione tra le due classi del dataset.

**4.9.2 Creazione Modello e Addestramento**  
Il modello di rete neurale è stato implementato utilizzando il framework Keras.

Il codice seguente mostra la definizione della rete:

Immagine che contiene testo, schermata, software, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

**Spiegazione del codice:**

* **Sequential()**: definisce una rete neurale sequenziale in Keras.
* **Dense(30, activation="relu")**: aggiunge un livello completamente connesso con 30 neuroni e attivazione ReLU.
* **Dense(1, activation='sigmoid')**: aggiunge un'unità di output con attivazione sigmoide per la classificazione binaria.
* **Compilazione del modello**: utilizza la funzione di perdita binary\_crossentropy, ideale per la classificazione binaria, e l'ottimizzatore adam, scelto per la sua efficienza in vari scenari.

Una volta definito il modello, si procede con l’addestramento:

Immagine che contiene testo, Carattere, schermata, linea

Il contenuto generato dall'IA potrebbe non essere corretto.

* **Epochs = 30**: il numero di epoche stabilisce quante volte il modello analizzerà l'intero dataset di training.
* **Batch size = 64**: definisce il numero di istanze processate prima dell’aggiornamento dei pesi, bilanciando precisione ed efficienza computazionale.

**4.9.3 Valutazione Finale**

Dopo l'addestramento, il modello è stato valutato sui dati di test e validazione.

* **Validation Accuracy:** 88.45%, con una loss di 0.3047.
* **Test Accuracy:** 86.04%, con performance stabili rispetto alla validazione.

**Metriche di Classificazione:**

Immagine che contiene testo, schermata, Carattere, numero

Il contenuto generato dall'IA potrebbe non essere corretto.

**Interpretazione:**

* **Precisione Classe 0 (0.93)**: pochi falsi positivi per la classe 0.
* **Recall Classe 1 (0.93)**: buona identificazione della classe 1 con pochi falsi negativi.
* **F1-score medio (0.86)**: equilibrio ottimale tra precisione e recall.

**Grafici di Valutazione**

**Confusion Matrix:**

Immagine che contiene testo, schermata, design

Il contenuto generato dall'IA potrebbe non essere corretto.

* **681 veri positivi** (classe 0 correttamente classificata).
* **718 veri negativi** (classe 1 correttamente classificata).
* **172 falsi positivi** (erroneamente classificati come classe 1).
* **55 falsi negativi** (erroneamente classificati come classe 0).

Il modello mostra una leggera difficoltà nella distinzione della classe 0, evidenziata da un numero maggiore di falsi positivi rispetto ai falsi negativi.

**ROC Curve**:

Immagine che contiene linea, Diagramma, diagramma, pendio

Il contenuto generato dall'IA potrebbe non essere corretto.

* La curva ROC è **ben al di sopra della diagonale** (baseline), indicando che il modello ha una **buona capacità predittiva**.
* **Il rapido aumento iniziale** suggerisce un modello con un’alta sensibilità e pochi falsi negativi.
* La curva **si appiattisce vicino al punto (1,1)**, indicando che il modello è in grado di classificare correttamente quasi tutte le istanze positive senza molti falsi positivi.

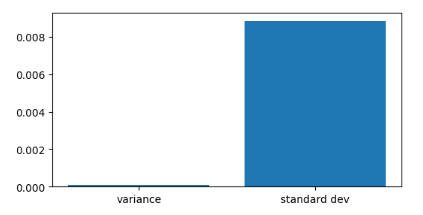
Il grafico della **ROC Curve** mostra quindi che il modello possiede **un'ottima capacità di discriminazione** tra le due classi, bilanciando **sensibilità (recall) e specificità**, e riducendo al minimo il numero di classificazioni errate.

* **Precision-Recall Curve**: dimostra che il modello è efficace nel rilevare le istanze positive reali.

Immagine che contiene testo, schermata, Rettangolo, diagramma

Il contenuto generato dall'IA potrebbe non essere corretto.

* **Bar Chart di Varianza e Deviazione Standard**: evidenzia un’ottima coerenza nei risultati, sottolineando la stabilità della rete neurale.



**4.10 Conclusioni:**

Come metriche per confrontare i nostri modelli abbiamo usato la media di accuratezza, varianza e deviazione standard generata attraverso il processo di cross-validation, la media pesata del f1-score, la precisione media e l’AUC.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **MODELLO** | **ACCURATEZZA** | **VARIANZA** | **DEV.STANDAD** | **F1-SCORE** | **AVERAGE-PRECISION** | **AUC** |
| **KNN** | 0.884 | 0.0038 | 0.062 | 0.91 | 0.80 | 0.962 |
| **RANDOM FOREST** | 0.928 | 0.0056 | 0.075 | 0.94 | 0.89 | 0.954 |
| **SVM** | 0.955 | 0.0047 | 0.069 | 0.95 | 0.92 | / |
| **NEURAL NETWOK** | 0.951 | 0.0002 | 0.014 | 0.96 | 0.93 | 0.992 |

Analizzando le prestazioni generali dei vari algoritmi di apprendimento senza entrare nei dettagli di ogni singolo campo, possiamo osservare le seguenti tendenze:

* ***Neural Network*** presenta deivalori che indicano un alto livello di precisione e coerenza nei risultati. L'F1-Score, l'Average-Precision e l'AUC sono i più alti tra tutti i modelli considerati, dimostrando una notevole capacità predittiva e discriminativa del modello.
* ***Support vector machines (SVM)*** presenta un'accuratezza ancora più elevata dimostra un'elevata competenza nel classificare i dati. La bassa varianza sottolinea la capacità di generalizzare su diverse partizioni dei dati, ma la leggermente maggiore deviazione standard potrebbe indicare alcune variazioni nelle prestazioni tra i fold.
* ***Random Forest*** ha dimostrato generalmente prestazioni quasi eccellenti. L'accuratezza del modello, è un risultato notevole e suggerisce che il modello sta facendo un buon lavoro nella classificazione complessiva. L’algoritmo è coerente nel corso delle esecuzioni e non dimostra sostanziali disuguaglianze di prestazioni tra le classi.
* ***K-Nearest Neighbor (KNN)*** i risultati indicano che questo modello ha una buona capacità di discriminare tra le classi, garantendo una buona performance globale, ci sono però segnali di disuguaglianza di prestazioni tra le classi e una variazione nelle prestazioni quando viene applicato a diversi set di dati. La varianza e la deviazione standard confermano la mancanza di piena coerenza del modello nelle iterazioni di cross-validation.

In conclusione, i dati evidenziano un chiaro ordine di performance tra i modelli considerati. Il Neural Network si pone come il modello più affidabile e preciso, seguito da SVM e Random Forest. KNN si attesta a un livello inferiore di accuratezza e prestazioni complessive.

**5. Clustering:**

Dopo aver esplorato i metodi di apprendimento supervisionato, abbiamo ritenuto che l'applicazione del clustering potesse offrire un ulteriore approccio per comprendere meglio la struttura intrinseca dei dati sulla dislessia. Questa tecnica ci consente di identificare naturali raggruppamenti nei dati, anche senza etichette di classe, e ci permette di rivelare relazioni sottostanti tra le prestazioni nei test cognitivi e comportamentali, fornendo una prospettiva diversa sulla distribuzione dei dati. Il clustering può essere di due tipi:

* **Clustering rigido (Hard clustering)**: assegna ciascun esempio a un unico cluster specifico.
* **Clustering morbido (Soft clustering)**: utilizza distribuzioni di probabilità per assegnare le classi associate a ciascun esempio.

**5.1 Decisioni di progetto:**

Abbiamo scelto il clustering rigido, in particolare il k-means, per raggruppare i soggetti in categorie basate sulle caratteristiche presenti nel file Dyslexia\_dataset.csv. L'obiettivo è individuare dei cluster con centroidi calcolati automaticamente, creando una nuova colonna nel dataset con l'assegnazione dei cluster. Il nuovo dataset, Dyslexia\_Dataset\_Clustered.csv, potrebbe aprire nuove prospettive sulla dislessia e la sua identificazione.ù

**5.2 K-means:**

K-means è un algoritmo di clustering che suddivide un insieme di dati in **K cluster**, dove K è un valore predefinito. Inizia posizionando casualmente K centroidi nel dataset e assegna ciascun punto al centroide più vicino. Successivamente, aggiorna la posizione dei centroidi in base alla media dei punti assegnati e ripete il processo fino a convergenza. L'obiettivo è minimizzare la somma dei quadrati delle distanze tra i punti e i centroidi assegnati, creando cluster di dati simili.

Dato che parte delle caratteristiche nel dataset sono di natura categorica e l'algoritmo k-means accetta solo caratteristiche numeriche, è stato necessario convertirle. A tal fine, abbiamo scelto di utilizzare la tecnica di **one-hot encoding**, che crea una nuova variabile binaria univoca per ogni categoria.

Una volta codificate le variabili categoriche, abbiamo dovuto determinare il numero ottimale di cluster. Abbiamo utilizzato la tecnica del **metodo del gomito (elbow method)**, in cui sull'asse delle ordinate sono rappresentati i valori delle somme dei quadrati intra-cluster (**WCSS**), mentre sull'asse delle ascisse sono rappresentati i K cluster.

Osservando il grafico del gomito, abbiamo notato che il punto di svolta netto si trova intorno a **K=4**, suggerendo che questo sia il numero ottimale di cluster.

Abbiamo testato diversi valori di **K**:

* Con **K=3**, il **Silhouette Score** ottenuto è stato **0.102**.
* Con **K=4**, il **Silhouette Score** è salito a **0.115**, indicando una migliore separazione tra i cluster.
* Con **K=5**, il **Silhouette Score** è sceso a **0.082**, suggerendo una minore qualità dei cluster.

Poiché **K=4** ha ottenuto il miglior punteggio in termini di separazione dei cluster, è stato scelto come valore ottimale per il clustering.

Le metriche scelte per la configurazione del modello sono state:

* **n\_clusters=4**: il numero di cluster in cui il dataset deve essere diviso.
* **random\_state=42**: questo parametro controlla la riproducibilità dei risultati. Fissando un valore per random\_state, il modello fornirà sempre gli stessi risultati quando viene addestrato con gli stessi dati.
* **n\_init=10**: specifica il numero di volte che l'algoritmo viene eseguito con diverse inizializzazioni casuali dei centroidi. L'algoritmo seleziona la soluzione migliore tra i tentativi basandosi sulla somma dei quadrati delle distanze. È stato scelto un valore non troppo alto, ma sufficiente per massimizzare la qualità dell'output.

**Grafico:**

Immagine che contiene testo, linea, diagramma, Diagramma

Il contenuto generato dall'IA potrebbe non essere corretto.

**5.3 Valutazione Finale:**

Sono state valutate le prestazioni del modello utilizzando due metriche:

* **WCSS (Within-Cluster Sum of Squares)**: calcola la somma dei quadrati delle distanze di ogni punto rispetto al proprio centroide di cluster. Un valore più basso di WCSS indica una maggiore coesione all'interno dei cluster.
* **Silhouette Score**: valuta la coesione all'interno dei cluster e la separazione tra i cluster. È calcolato per ciascun punto e rappresenta il rapporto tra la distanza media al proprio cluster e la distanza media ai cluster più vicini. Un punteggio più alto del Silhouette Score indica una migliore separazione dei cluster.

Nel nostro caso, il **Silhouette Score ottenuto è pari a 0.115**, il che suggerisce che i cluster identificati sono ben separati e possono fornire indicazioni utili sulla struttura dei dati.

**5. Conclusioni:**

In questo progetto, abbiamo voluto esplorare il problema della dislessia attraverso un approccio basato sull'intelligenza artificiale, applicando tecniche di clustering per analizzare e segmentare i dati raccolti dai test cognitivi e comportamentali. L'obiettivo principale era individuare gruppi omogenei di soggetti, mettendo in luce eventuali pattern nascosti nei dati, e contribuire così a una migliore comprensione della dislessia.

L'applicazione dell'algoritmo K-Means ha permesso di suddividere il dataset in cluster significativi, fornendo nuove prospettive su come le caratteristiche cognitive e linguistiche possano influenzare la categorizzazione dei soggetti. Sebbene il **Silhouette Score** non abbia evidenziato una separazione perfetta tra i cluster, i risultati ottenuti dimostrano il potenziale di questa tecnica nell’analisi di dati complessi legati ai disturbi dell’apprendimento.

**6.1 Sviluppi futuri:**

Questo progetto apre la strada a molteplici sviluppi futuri. Un'evoluzione naturale potrebbe essere l’integrazione di tecniche di **clustering morbido** per identificare meglio le sovrapposizioni tra i gruppi e migliorare la comprensione delle differenze individuali tra i soggetti.

Inoltre, potrebbe essere interessante combinare il clustering con algoritmi di **apprendimento supervisionato**, utilizzando le informazioni emerse per costruire modelli predittivi più raffinati. Un'altra possibilità è ampliare il dataset con nuove variabili o integrare dati provenienti da diverse fonti per migliorare l’accuratezza delle analisi.

Infine, la metodologia sviluppata potrebbe essere estesa ad altri disturbi dell’apprendimento o condizioni neurologiche, come la discalculia o il disturbo da deficit di attenzione e iperattività (ADHD), contribuendo alla creazione di strumenti diagnostici innovativi basati sull’intelligenza artificiale.