**CLASSIFICATORE: DecisionTree**

Gli alberi decisionali sono una delle tecniche di classificazione maggiormente utilizzate.

Hanno una **struttura gerarchica** costituita da un insieme di nodi, correlati da archi orientati e etichettati, che permette di rappresentare un insieme di **regole di classificazione**.

Perciò per classificare un nuovo record in questi tipo di classificatori è necessario semplicemente seguire la regola di classificazione giusta e quindi il giusto percorso partendo dalla radice fino ad arrivare ad una foglia.

Abbiamo scelto questo tipo di classificatori perché:

* Sono classificatori **molto semplici da capire e spiegare** inoltre la loro struttura permette chiaramente di mostrare come si sviluppa il processo decisionale.
* Non richiede normalizzazione o standardizzazione dei dati poiché le divisioni sono basate su soglie specifiche delle caratteristiche.
* Possono catturare relazioni non lineari tra gli attributi e la classe dei record.

Ovviamente sempre consapevoli dei contro:

* Decision boundary di limitata espressività.
* Rischio di overfitting a causa di un troppo adattamento ai dati di training.
* Rischio di frammentazione dei dati.

Sarà quindi necessario determinare i migliori iperparametri proprio per evitare i problemi appena citati.

**REALIZZAZIONE E TEST CLASSIFICATORE**

Per la sua realizzazione e ottimizzazione si sono utilizzate le funzioni offerte da scikit-learn mentre per quanto riguarda la scelta degli iperparametri si è deciso di ottimizzare la **massima profondità** dell’albero e il **massimo numero di nodi foglia**. Questi ultimi se ottimizzati evitano che l’albero si adattarsi troppo ai dati o di raggiunga una complessità eccessiva che ridurebbe la sua capacità di generalizzazione. Per quanto riguarda la misura della purezza dei nodi, si è stabilito a priori che sarò calcolata con **l’indice di GINI**.

In pratica quindi una volta creato l’albero, per il tuning si è scelto di utilizzare la funzione di scikit\_learn ‘GridSearch’ la quale permette di ottenere i migliori iperparametri a seguito di una procedura di ricerca esaustiva e molto efficace la quale utilizza, per ridurre eventuali rischi di overfitting, una tecnica di cross-validation per la verifica delle performance.

Infine, abbiamo addestrato un nuovo modello con i migliori iperparametri individuati ed effettuato le predizioni sul test set calcolandone l’accuratezza. Un ulteriore verifica delle performance è stata poi effettuata sul training set attraverso una cross validation a 5 fold.

Così facendo otteniamo sia il risultato effettivo sul test set, utile per capire quanto è efficace il modello su dati nuovi, e sia una verifica sul training set che ci permette anche di capire se l’errore non sia l’overfitting ma magari il modello in sé che non riesce a adattarsi ai dati.

**RISULTATI**

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

**ANALISI RISULTATI**

Dai risultati quindi gli iperparametri migliori sembrano essere una profondità massima di 16 e un numero massimo di nodi foglia di 24.

Il risultato sul test set è abbastanza pessimo e ciò, ricordando che si sta utilizzando il dataset grezzo senza alcun preprocessing, può essere dato da diversi fattori tra cui:

1. Il dataset essendo sbilanciato favorisce le classi con maggior numero di record, il che vuoldire che l’influenza delle classi maggioritarie porta le regole di classificazione a pendere a loro favore a discapito delle classi con meno record con un conseguente aumento nella difficoltà di classificare adeguatamente proprio i record di quest’ultime.
2. Il dataset avendo attributi estremamente correlati tra loro può portare l’albero a crescere in complessità o nel numero di nodi senza però un effettivo guadagno nelle performance.

L’unica nota positiva sembra essere la resistenza degli alberi decisionali agli attributi con scale diverse (nel nostro dataset è presente una grossa differenza di scale di valori tra gli attributi).

Dai risultati si nota poi che il risultato ottenuto nel training set è molto simile a quello ottenuto sul test set, si ha quindi un **underfitting** che in parte può essere dato dai problemi del dataset ma in parte probabilmente è dato dal fatto che il modello non riesce a adattarsi ai dati e, di conseguenza, non riesce dividere adeguatamente i record nelle loro classi.

Una possibilità è che l’albero decisionale non sia adatto a questo tipo di problema; noi però crediamo che la problematica sia, più che il classificatore, il fatto che gli attributi non trasmettano abbastanza informazioni ai fini della predizione.

Tutto ciò, come fanno notare anche gli iperparametri migliori individuati, porta l’albero a cercare di compensare questo problema crescendo in numero di nodi e soprattutto in profondità (16 è una profondità esagerata per un problema di questo tipo) portando infine a un maggiore **overfitting**.

Vedremo se con il preprocessing sarà possibile migliorare le performance di questo classificatore.