Il Machine Learning per determinare la qualità dell'acqua

Presentato da Martino Corti

Introduzione

In questo notebook condurrò un'analisi predittiva sul dataset "Copy of water_potability", seguendo il seguente approccio:

- Analisi esplorativa del dataset per comprenderne la struttura e le caratteristiche principali.
- Data Manipulation, applicando eventuali trasformazioni o pulizia dei dati, se necessario.
- Selezione delle feature: creerò un subset contenente le variabili più significative. Da questa fase in poi, i modelli predittivi verranno applicati sia al dataset completo che a questo subset.
- Scelta del modello predittivo e definizione della metrica di valutazione.
- Ottimizzazione degli iperparametri per migliorare le performance dei modelli più promettenti.
- Applicazione dei modelli ottimizzati e confronto finale delle performance ottenute.

L'obiettivo è individuare il modello più efficace per prevedere la potabilità dell'acqua. 🚀

Descrizione del Dataset

Il dataset analizzato contiene 3.276 campioni d’acqua, ciascuno descritto da 10 caratteristiche. Di queste, 9 features numeriche (di tipo float) rappresentano le proprietà chimico-fisiche dell'acqua, mentre l'ultima è una variabile binaria (0 o 1) che indica la potabilità del campione. Quest'ultima feature fungerà da variabile target (y) per le analisi predittive, determinando se un campione d'acqua è potabile o meno.

EDA e Data Manipulation

01

Dalle analisi emerge che le features *ph, Sulfate* e *Trihalomethanes* contengono al loro interno valori nulli. Sostituirli con la media o la mediana potrebbe “appiattire” la varianza, procedo quindi a sostituirli con nuovi valori determinati con un *RandomForestRegressor*

02

Per ogni feature, con il metodo dei quantili stabilisco e individuo gli *outliers*. Una volta identificati procedo a eliminarli, filtrando le features con una maschera appositamente creata.

03

La feature *Solids* ha un range di valori molto più ampio delle altre features. Prima di applicare la scalatura verifico la sua distribuzione con un istogramma: appare piuttosto gaussiana, requisito necessario per applicare il metodo *StandardScaler* (scalatura).

Creazione del subset

Procedimento

Dopo aver constatato con una matrice di correlazione che non esistono correlazioni lineri tra le features, mi concentro su un altro quesito: quali variabili indipendenti sono maggiormente legate alla variabile target (y)? Per rispondere a questa domanda ho calcolato il chi-squared test.

Risultato

Servendomi della funzione *chi2_contingency* stampo per ogni feature i valori *chi2* e *p-value*. Ne emerge che solo le colonne *ph* e *Sulfate* presentano un valore di *p-value* minore di 0.05 e quindi significativamente correlate. Il subset che servirà di confronto al dataset, presenterà queste due features.

Scelta dei modelli predittivi

01

Dato lo sbilanciamento tra le due classi e il fatto che l'*F1-macro* assegna lo stesso peso a entrambe, ho scelto questa metrica come criterio di valutazione per i modelli.

02

Scelgo due modelli da “battere”: il primo è un modello che assegna le classi in modo random, il secondo predice sempre la classe maggioritaria.

03

Importo i seguenti modelli che verranno testati: *LogisticRegression*, *RandomForestClassifier* e *KNeighborsClassifier*.

04

Utilizzando la tecnica della cross-validation, applico tutti i modelli sia al dataset di addestramento completo che al subset selezionato.

05

Per ogni modello viene calcolata la media del punteggio di f1-macro ottenuto nel cross-validation.

06

Nel dataset completo e nel subset, il **RandomForestClassifier** e il **KNeighborsClassifier** si confermano come i migliori modelli, superando anche le baseline definite nel punto 2.

Validazione degli iperparametri

Obiettivo

Identificati i modelli più performanti, il passo successivo consiste nell'ottimizzare i loro iperparametri. Questa fase di validazione consentirà di valutare se un aumento della complessità del modello può tradursi in un miglioramento delle prestazioni.

Svolgimento

Per ogni modello imposto un lista di valori, ovvero un elenco di vari iperparametri da testare. Ogni combinazione di iperparametro sarà testata grazie alla *GridSearch*, utilizzando come metrica di valutazione la *f1-macro*.

Risultato

Per ogni set (dataset e subset) avrò due modelli (RandomForestClassifier KNeighborsClassifier) con i relativi iperparametri ottimizzati.

Fase di test e conclusioni

I modelli ottimizzati vengono infine testati sulla porzione di test dei due dataset. Utilizzando la metrica f1-macro, vengono confrontate le performance di ciascun modello. I risultati evidenziano che il miglior punteggio è stato ottenuto dal *RandomForestClassifier* applicato al dataset completo. Questo suggerisce che il subset con le feature più significative non è stato in grado di catturare sufficienti pattern predittivi, risultando meno efficace rispetto al dataset con tutte le variabili.

Grazie per l’attenzione!