

Come Funzionano le Foreste Casuali: Struttura Interna e Logica di Apprendimento

Prof. Fedeli Massimo

1 Introduzione

2 Dal singolo albero alla foresta

Un albero decisionale prende decisioni tramite una sequenza di regole.

Ad esempio:

- Temperatura $\geq 15^{\circ}\text{C}$?
- C'è vento?

Ogni nodo divide i dati in due gruppi.

L'obiettivo è rendere ogni gruppo il più omogeneo possibile.

Questo processo è chiamato:

partizionamento dello spazio delle feature.

3 Il problema dell'instabilità

Piccoli cambiamenti nei dati possono produrre alberi completamente diversi.

Questo accade perché:

- ogni nodo dipende dai dati disponibili
- ogni divisione influenza tutte le successive

Di conseguenza:

un albero decisionale ha alta varianza

Cioè è molto sensibile ai dati di training.

4 Overfitting

Se lasciamo crescere completamente l'albero:

- imparerà perfettamente i dati
- ma anche il rumore

Questo produce regioni decisionali molto frastagliate.

Il modello diventa preciso sui dati visti ma incapace di generalizzare.

5 L'idea della foresta

La soluzione non è costruire un albero migliore.

È costruire molti alberi diversi.

Una foresta casuale è quindi:

Un insieme di alberi che apprendono in modo indipendente

La decisione finale nasce dalla loro combinazione.

6 Cosa succede durante l'addestramento

Il funzionamento interno può essere diviso in quattro fasi fondamentali.

6.1 1. Campionamento Bootstrap

Per ogni albero viene creato un nuovo dataset.

Questo avviene estraendo casualmente dati con reinserimento.

Se il dataset originale ha 100 esempi:

ogni albero riceve un nuovo dataset di 100 esempi, ma con duplicati.

Alcuni dati non verranno utilizzati.

Questo genera alberi diversi.

6.2 2. Costruzione indipendente degli alberi

Ogni albero viene addestrato separatamente.

Non esiste comunicazione tra gli alberi.

Ogni albero cerca la migliore divisione locale dei dati.

7 Come un albero divide i dati

Ogni nodo deve scegliere:

- quale feature usare
- quale soglia scegliere

La scelta si basa sulla riduzione dell'impurità.

7.1 Indice di Gini

L'impurità è misurata tramite:

Dove p_i è la proporzione della classe i .

Se tutte le osservazioni appartengono alla stessa classe:

[$G = 0$]

Il nodo è puro.

L'albero sceglie la divisione che riduce maggiormente il Gini.

8 3. La casualità nelle feature

Qui nasce la vera differenza rispetto agli alberi classici.

In ogni nodo:

non vengono considerate tutte le feature.

Solo un sottoinsieme casuale.

Questo significa che due alberi:

- vedono dati diversi
- considerano variabili diverse

Anche se partono dallo stesso dataset.

9 4. Crescita completa

Gli alberi nella foresta:

- non vengono potati
- crescono profondamente

Ogni albero diventa altamente specializzato.

10 Perché la foresta funziona

Ogni albero commette errori.

Ma non gli stessi errori.

La casualità introduce:

- decorrelazione

Gli errori non sono sincronizzati.

11 Aggregazione delle decisioni

Dopo l'addestramento:

ogni albero produce una previsione.

11.1 Classificazione

Si usa la maggioranza.

11.2 Regressione

Si usa la media.

Questo riduce la varianza del modello.

12 Interpretazione geometrica

Ogni albero suddivide lo spazio delle feature in regioni.

Una foresta costruisce molte suddivisioni diverse.

La decisione finale è la combinazione di queste partizioni.

Il risultato è una superficie decisionale più stabile.

13 Ruolo del numero di alberi

Aumentando il numero di alberi:

- diminuisce la varianza
- aumenta la stabilità

Ma aumenta il costo computazionale.

14 Bagging

Come spiegato nel documento , questo processo prende il nome di:

Bootstrap Aggregation

Bagging significa:

- addestrare più modelli su dati diversi
- aggregare le loro decisioni

15 Perché riduce l'overfitting

Un singolo albero apprende il rumore.

La media di molti alberi riduce l'effetto del rumore.