Relazione su Random Forest

Niccolò Niccoli

1 Descrizione esperimento

Il lavoro svolto consiste nell'implementare l'algoritmo random forest e poi eseguire un confronto tra questa implementazione e quella disponibile in scikit-learn. Le due implementazioni sono state testate su tre dataset disponibili nella UCI Machine Learning Repository.

1.1 Dataset

I dataset selezionati sono:

1. Bank Marketing Data Set[1], dove date informazioni sui clienti di una banca è possibile stabilire se questi abbiano sottoscritto il prodotto offerto loro.
2. Breast Cancer Diagnostic Data Set[2], dove date delle informazioni riguardanti delle cellule tumorali è possibile determinare se il tumore è benigno o maligno.
3. Wine Data Set[4], dove date informazioni sulle caratteristiche del vino è possibile determinare il tipo di uva utilizzato.

I dataset vengono importati utilizzando pandas e poi, dopo aver rimosso eventuali valori nulli, vengono convertiti in un array di numpy. A questo punto i dati che non sono numerici e quelli relativi alla classe di appartenenza di ciascun esempio vengono codificati utilizzando il metodo fit_transform della classe LabelEncoder situata all'interno di sklearn.preprocessing.
Successivamente il dataset iniziale viene suddiviso in due sottoinsiemi utilizzando il metodo train_test_split: uno formato dall'67% degli esempi su cui allenare il modello e uno formato dal restante 33% per testarne le capacità classificatorie.

1.2 Alberi di decisione

Nell'implementazione degli alberi di decisione viene utilizzato il Gini index come funzione per calcolare l'impurità di un nodo e di conseguenza per calcolare il guadagno di informazione che comporta scegliere un determinato attributo ed una determinata soglia al momento di dividerlo.

La suddivisione dei nodi termina quando questi superano una certa profondità (che può essere scelta arbitrariamente e che negli esperimenti che svolti è impostata a 40) o quando il numero di esempi che devono essere classificati scende sotto una certa soglia (anche questa può essere scelta liberamente, in questo caso è stata scelta pari a 2).
Per quanto riguarda il numero di features da considerare ad ogni split è stato scelto di attenersi alla quantità *standard* e quindi vengono considerate ogni volta $\sqrt{n}$ features (n = numero di features totali).

1.3 Random Forest

Per l'implementazione di random forest viene utilizzata come guida lo pseudocodice fornito in The Elements of Statistical Learning [3]: nell'istante in cui avviene l'inizializzazione della foresta è necessario scegliere quanti alberi creare (in questo caso 200) e poi per ciascun albero viene creato un *bootstrapped dataset* che viene utilizzato per allenarlo.
Nel momento in cui si deve ottenere la predizione ultima vengono considerate le predizioni di tutti gli alberi della foresta e poi viene classificato il campione come appartenente alla classe che compare nella maggior parte di esse.

2 Risultati

Le due implementazioni sono state confrontate considerandone la precisione, quindi $\frac{\#\text{campioni classificati correttamente}}{\#\text{campioni totali}}$. Per calcolare questo valore si utilizza il metodo accuracy_score di sklearn.metrics.
Per ciascun insieme di dati viene ripetuta la misurazione 10 volte cambiando ogni volta il seed al fine di ridurre la possibilità di avere dati relativi esclusivamente ad uno specifico seed, quindi i seguenti risultati sono la media aritmetica di queste misurazioni.

	Implementazione dell'autore	Implementazione di scikit-learn
Wine Data Set	0.9085	0.9780
Bank Marketing Data Set	0.8832	0.8945
Breast Cancer Diagnostic Data Set	0.9584	0.9677

3 Conclusioni

Dai risultati degli esperimenti è possibile concludere che l'implementazione realizzata dall'autore performi bene nei casi testati. In particolare questo è visibile negli esperimenti riguardanti il secondo e il terzo dataset.
Per quanto riguarda il primo dataset, sebbene ci sia un distacco maggiore tra la precisione delle due implementazioni, dal momento che il valore ottenuto è sufficientemente alto, si può reputare riuscito l'esperimento.

Riferimenti

[1] *bank.zip, bank.csv.* URL: https://archive.ics.uci.edu/ml/machine-learning-databases/00222/.

[2] *breast-cancer-wisconsin.data.* URL: https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/.

[3] R. Tibshirani T. Hastie and J. Friedman. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction. 2nd edition.* 2017, p. 588.

[4] *wine.data.* URL: https://archive.ics.uci.edu/ml/machine-learning-databases/wine/.