Algoritmo dei K-Nearest Neighbours

Davide Peccioli

31 marzo 2021

Dato un punto, per mezzo di questo algoritmo è possibile classificarli, associandovi una classe di appartenenza.

Il codice presentato, consente, dato un insieme di punti di cui si conosce la classe, ed un insieme di punti da testare, ma di cui comunque si conosce la classe, di stabilire per quali valori k l'algoritmo funziona meglio.

1 Funzionamento dell'algorimo

L'algoritmo funziona sulla base di un insieme di punti di partenza (denominato **train set**), ciascuno associato alla propria classe: dato un punto da analizzare (denominato **A**), il risultato sarà la classe di appartenenza del punto stesso.

All'algoritmo viene passato un parametro k. Verranno presi i k punti del **train set** più vicini ad **A**, e la classe più frequente tra questi punti sarò la classe assegnata ad **A** stesso.

2 Codice

Come già detto, il codice presentato non è semplicemente lo svolgimento dell'algoritmo, bensì un codice che permette di stabilire per quale parametro k l'algoritmo funziona meglio sui punti considerati.

Il codice è visualizzabile su GitHub.

3 Risultati

L'esecuzione su terminale del codice presentato è disponibile qui.

I due grafici che sono stati prodotti dal codice sono:

Nella figura 1 possiamo vedere in nero i punti del **train set**, e in magenta i punti che verranno processati dall'algoritmo.

Nella figura 2 è rappresentato l'andamento dell'accuratezza (in percentuale) dell'algoritmo al variare di k, per valori da 0 a 100. Possiamo notare come contrariamente da quello che ci si può aspettare, i valori di k per cui l'accuratezza è massima sono relativamente bassi:

$$k \in K = \{3; 4; 5; 7; 8; 9; 10\}$$

Per tutti i valori di $k \in K$ l'accuratezza è pari al 91%, mentre il valore minimo, toccato con $k = 100$ è del 50%.

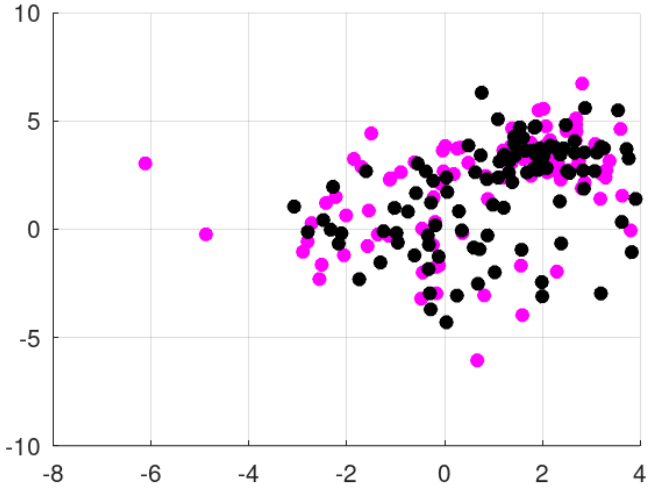

Figura 1: Punti di train e di test

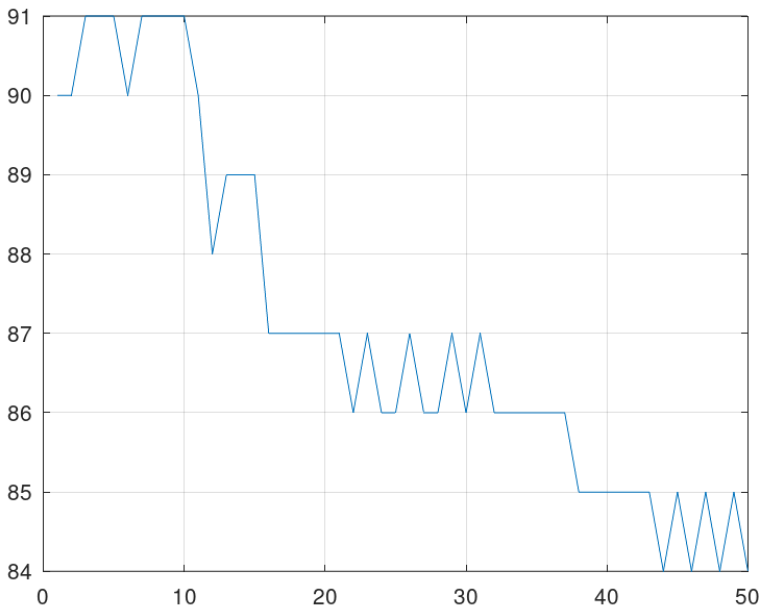

Figura 2: Accuratezza al variare di k

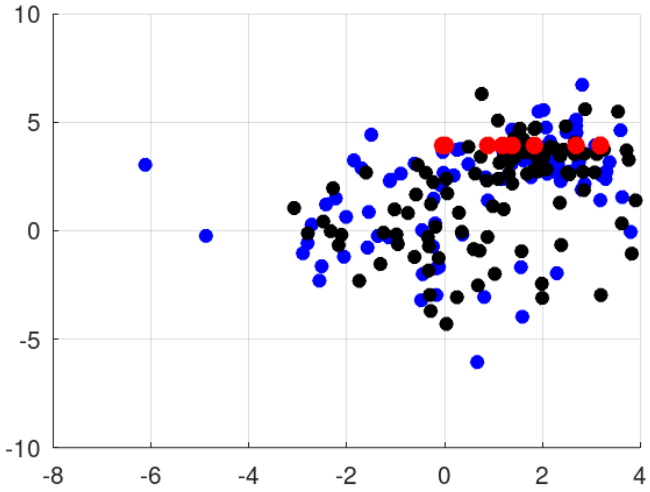

Figura 3: Punti di train e di test

Con una piccola modifica al codice, è possibile stampare i punti dell'insieme di test per cui l'algoritmo (con $k = 4$) non ha funzionato: il risultato è mostrato nella figura 3 (punti in rosso).

Possiamo quindi notare come l'algoritmo abbia fallito in una zona sul piano particolarmente densa di punti, e in cui quindi, probabilmente, la precisione dell'algoritmo non è sufficiente.