

# Progetto Quality Treshold - Documentazione

A cura di: Vito Stefano Birardi

26 aprile 2025

## Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
<b>2</b>	<b>Guida all'installazione</b>	<b>4</b>
2.1	Installazione JDK . . . . .	4
2.2	Variabili di ambiente . . . . .	4
2.3	Installazione MySQL . . . . .	5
2.4	Schermata iniziale . . . . .	5
2.5	Download dei file . . . . .	6
2.6	Configurazione del account MySQL . . . . .	6
2.7	Schermata finale . . . . .	7
<b>3</b>	<b>Eseguire il progetto base</b>	<b>8</b>
3.1	Algoritmo . . . . .	8
3.2	Struttura del progetto base . . . . .	8
3.3	Esecuzione di QTB_Server . . . . .	9
3.4	Esecuzione di QTB_Client . . . . .	9
3.5	Test QT_BASE . . . . .	10

## 1 Introduzione

Il progetto in questione utilizza l'algoritmo **Quality Threshold** per l'analisi dei dati.

Tali dati possono essere estratti da un file, oppure possono essere estratti dalle tabelle di un database MySQL.

L'algoritmo quality threshold è un algoritmo di clustering che si basa sulla minimizzazione della distanza tra i punti e il centroide del cluster.

L'algoritmo è stato sviluppato per essere utilizzato in contesti in cui i dati sono rumorosi e non lineari, e si basa su un approccio gerarchico per la creazione dei cluster.

Rispetto ad altri algoritmi di partizionamento dei dati, come il k-means, richiede una potenza di calcolo maggiore, ma a differenza di altri, non richiede di specificare il numero di cluster da creare, ma solo il raggio massimo. Inoltre, l'algoritmo è in grado di restituire sempre lo stesso risultato, nonostante esso venga eseguito ripetutamente, a differenza di altri algoritmi che possono restituire risultati diversi a seconda della loro inizializzazione.

## 2 Guida all'installazione

Prima di essere in grado di eseguire il programma, è necessario eseguire il file `risorse.bat` contenuto nella cartella Risorse.

Una volta eseguito il file, si aprirà una pagina di Powershell e seguirà un download.



Figura 1: Schermata di download

Al termine del download del file compresso delle risorse, verranno estratti i file necessari per l'esecuzione del programma.

Per il progetto in questione è necessario installare il Java Developer Kit (JDK) nella versione 22.0.1 e il software di gestione del database MySQL nella sua versione 8.0.39.

La prima scheda di installazione che apparirà è quella del JDK.

Verrà chiesto se si vuole eseguire l'installazione del JDK tramite permessi di amministratore, cliccare su Sì.

### 2.1 Installazione JDK

Una volta confermata l'esecuzione come amministratore, si procederà all'installazione del JDK.



Cliccare su **Avanti**, nuovamente **Avanti** e infine, nel caso in cui l'installazione sia andata a buon fine, si dovrà cliccare sul tasto **Chiudi**.

### 2.2 Variabili di ambiente

è consigliato creare una variabile di ambiente per il JDK. Per farlo, si dovrà cercare la voce **variabili di ambiente** nella barra di ricerca di Windows e cliccare su Modifica le variabili di ambiente per il tuo account.

Successivamente si dovrà cliccare sul pulsante **Variabili di ambiente** e si aprirà una schermata con le variabili di ambiente.

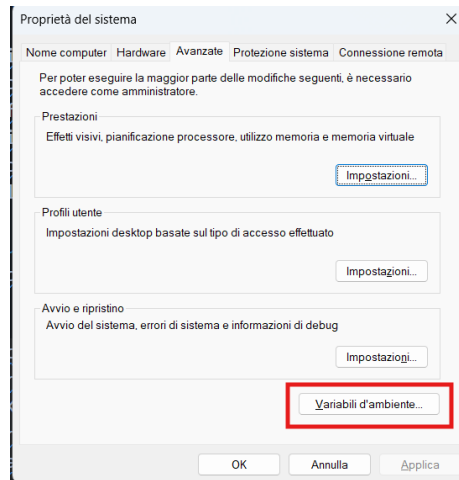


Figura 2: Schermata delle variabili di ambiente

Nella schermata, si dovrà cliccare sulla variabile **PATH** e modificarla in modo da aggiungere il percorso del JDK, inserendo il percorso in cui è stato installato il JDK.

Una volta completata l'installazione del JDK, si procederà automaticamente con l'installazione del MySQL.

## 2.3 Installazione MySQL

Anche qui sarà necessario fornire i permessi di amministratore, cliccare dunque su Sì.

## 2.4 Schermata iniziale

Dopodiché si aprirà la schermata di installazione di MySQL.

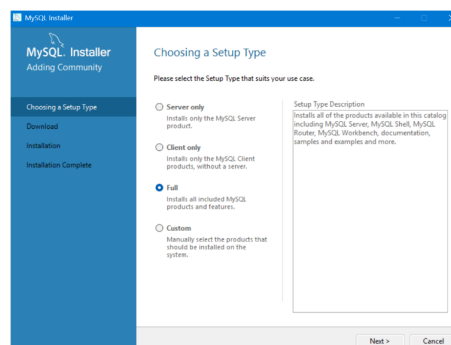


Figura 3: Scelta del setup

Selezionare il tipo di setup **Full** e cliccare su **Next**. Tale scelta permetterà di installare tutti i componenti necessari per l'esecuzione del programma.

## 2.5 Download dei file

Si passerà alla schermata di download dei file necessari per l'installazione, cliccare su **Execute**.

### Avvertenza

Tale schermata potrebbe richiedere un po' di tempo per il download dei file, a seconda della velocità della connessione internet.

Dopo aver scaricato i file, si dovrà cliccare nuovamente su **Execute**.

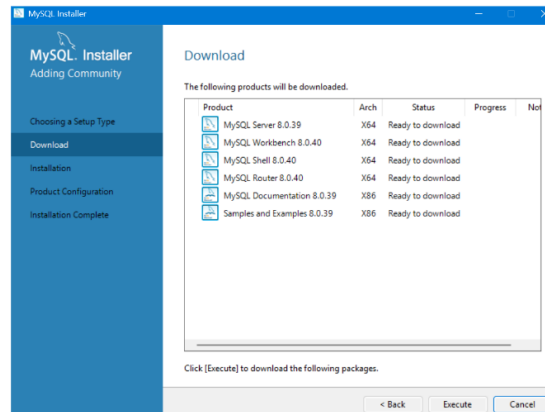


Figura 4: Download MySQL

## 2.6 Configurazione del account MySQL

Una volta terminato il download, si procederà con la configurazione dell'account MySQL. Si dovrà quindi scegliere una password, necessaria per accedere al database.

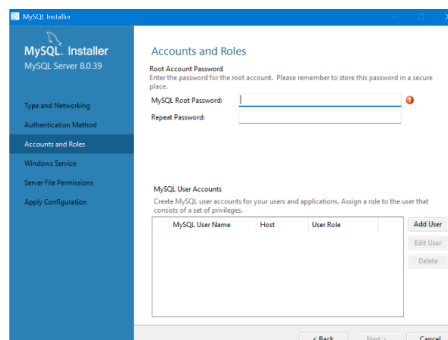


Figura 5: Configurazione account MySQL

Si giungerà infine alla schermata di applicazione della configurazione del database. Cliccare su **Execute** per applicare la configurazione.

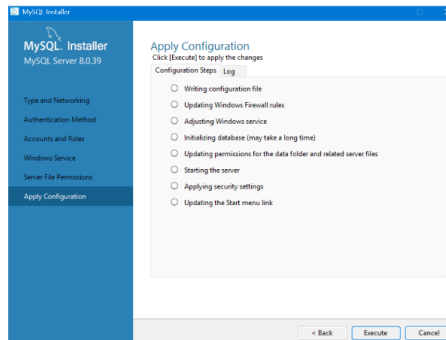


Figura 6: Completamento installazione

Una volta terminata anche l'installazione del MySQL, si dovrà cliccare su **Finish** per chiudere l'installler.

#### Avvertenza

Per un corretto funzionamento sia del JDK che del MySQL, è consigliato riavviare il computer.

## 2.7 Schermata finale

Al termine dell'installazione il programma `resources.bat` mostrerà la seguente schermata.



Figura 7: Completamento installazione

## 3 Eseguire il progetto base

### 3.1 Algoritmo

La base dell'algoritmo permette all'utente di scegliere un raggio massimo per il cluster.

Per ciascun punto interessato, ossia i punti con distanza inferiore al raggio impostato, l'algoritmo calcola i cluster candidati.

Una volta individuati i cluster, quello con il numero maggiore di punti verrà salvato nella cartella `./results/`, ossia nella root del progetto base. Il nome del file indicherà il database di partenza e il raggio massimo utilizzato per la ricerca, quindi scegliendo per esempio il database `gopicnic` con un raggio di 3, il file di output sarà `gopicnic_003.dmp`.

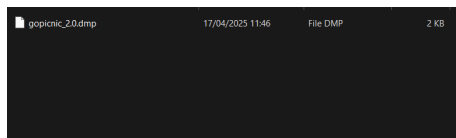


Figura 8: Risultati del salvataggio

Una volta creato il file di output, l'algoritmo provvederà ad eliminare anche i cluster non salvati, per evitare che vengano esaminati nuovamente.

Nel caso in cui più di un cluster abbia il numero massimo di punti, l'algoritmo ripeterà la procedura con il set ridotto di punti.

### 3.2 Struttura del progetto base

Il progetto base consiste in un'applicazione di tipo client/server. Di fatto, nella cartella `Jar+Bat` sono presenti i file `start_server.bat` per l'esecuzione del server e `start_client` per il client, che non sono altri che degli scritti batch per l'esecuzione dei file jar.

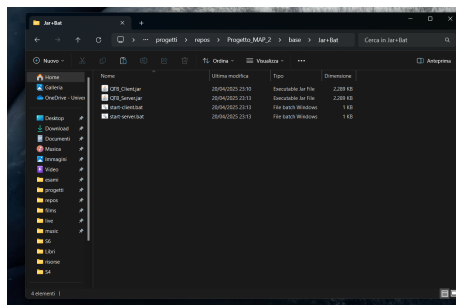


Figura 9: Jar del progetto base



**Avvertenza**

Per avviare il programma correttamente, è necessario eseguire prima il file `.bat` del server e poi quello del client.

### 3.3 Esecuzione di QTB\_Server

Il normale avviamento del file `start_server.bat` mostrerà la seguente schermata, che indica che il server è in ascolto sulla porta 8080, in attesa di una richiesta da parte del client.

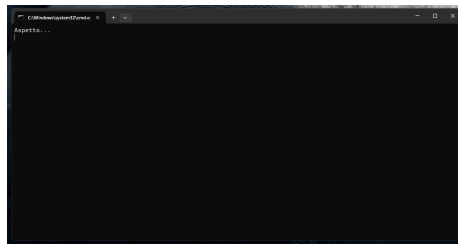


Figura 10: Server in ascolto

### 3.4 Esecuzione di QTB\_Client

Per quanto riguarda il client, una volta avviato, mostrerà la seguente schermata.

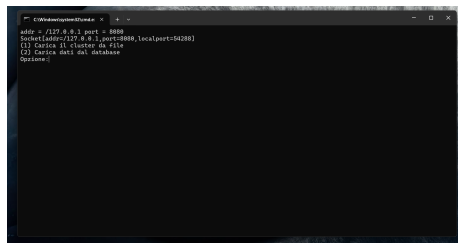
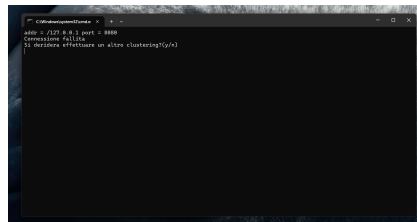


Figura 11: Schermata iniziale del client

**Avvertenza**

Nel caso in cui il server non si sia avviato correttamente, o sia stato avviato il client prima del server, il client mostrerà la seguente schermata, che indicherà appunto che la connessione al server non è avvenuta correttamente.



Una volta assicurati che il client comunichi correttamente con il server, si potrà decidere se caricare i dati da un file di un clustering precedente o da un database.

Se si sceglie di caricare i dati da una tabella del database, si procederà con l'inserimento del nome della tabella da cui si vogliono estrarre i dati. Nel caso in cui tale tabella non sia presente nel database, il client avviserà l'utente con un messaggio di errore e chiedendo di inserire un nome valido.

```
(1) Carica il cluster da file
(2) Carica dati dal database
Opzione:2
Tabella:vpdsin
La tabella inserita non esiste
Tabella:
```

Figura 12: Errore tabella non presente

Una volta scelto correttamente una tabella, si procederà nel determinare il raggio del clustering e, automaticamente, il programma restituirà il risultato.

```
Tabella:gopicnic
Raggio:2
Numero di clusters
1:Centroid=(rain 13.0 high weak no )
Examples:
[rain 13.0 high weak no ] dist=0.0
AvgDistance=0.0
2:Centroid=(sunny 0.1 normal weak no )
Examples:
[sunny 0.1 normal weak no ] dist=0.0
[sunny 40.0 low weak no ] dist=1.9975
AvgDistance=0.99875
3:Centroid=(sunny 30.3 high strong no )
Examples:
[sunny 30.3 high strong no ] dist=0.0
[overcast 12.5 high strong no ] dist=1.445
AvgDistance=0.7225
4:Centroid=(overcast 30.0 high weak yes )
Examples:
[sunny 30.3 high weak yes ] dist=1.0075
[overcast 30.0 high weak yes ] dist=0.0
[sunny 20.0 high weak yes ] dist=1.25
[overcast 29.21 normal weak yes ] dist=1.01975
[overcast 23.0 low weak yes ] dist=1.175
AvgDistance=0.8904500000000001
5:Centroid=(rain 0.0 normal strong no )
Examples:
[rain 40.0 high strong no ] dist=2.0
[rain 0.0 normal weak no ] dist=1.0
[rain 0.0 normal strong no ] dist=0.0
[overcast 0.1 normal strong no ] dist=1.0025
[rain 18.0 normal weak no ] dist=1.45
[sunny 12.5 normal strong no ] dist=1.3125
[rain 12.5 high strong no ] dist=1.3125
AvgDistance=1.1539285714285714
```

Figura 13: Risultato del clustering

### 3.5 Test QT\_BASE

Durante l'esecuzione del programma, può capitare che l'utente inserisca dei valori che quest'ultimo non è in grado di gestire. Chiaramente nonostante la presenza di queste situazioni, deve essere garantita la continuità di esecuzione del programma e la corretta amministrazione di queste situazioni estreme.


#### Gestione del raggio

Nel caso in cui l'utente inserisca un valore del raggio maggiore del massimo raggio possibile, il programma restituirà un messaggio di errore e chiederà

nuovamente il raggio.

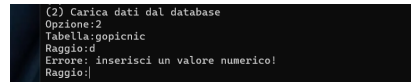
Nel caso invece in cui l'utente inserisca un valore non numerico, il programma restituirà un messaggio di errore e chiederà nuovamente il raggio.

In entrambi i casi, il programma non si fermerà e continuerà a chiedere il raggio fino a quando non verrà inserito un valore corretto.



```
C:\Windows\system32\cmd.exe
addr = /127.0.0.1 port = 8080
Socket[addr=/127.0.0.1,port=8080,localport=55355]
(1) Carica il cluster da file
(2) Carica dati dal database
Opzione:2
Tabella:gopicnic
Raggio:5
Il raggio inserito è troppo grande
Si desidera effettuare un altro clustering?(y/n)
```

(a) Valore del raggio troppo grande



```
(2) Carica dati dal database
Opzione:2
Tabella:gopicnic
Raggio:d
Errore: inserisci un valore numerico!
Raggio:
```

(b) Valore del raggio non numerico

Figura 14: Errori nel valore del raggio

## Uscita dal programma

Una volta calcolato il raggio, il programma chiederà all'utente se ha intenzione di effettuare nuovamente un clustering: in caso positivo chiederà nuovamente la misura del raggio, in caso opposto chiederà all'utente se vuole tornare al menù principale o chiudere il programma.



```
Si desidera effettuare un altro clustering?(y/n)
Si desidera ritornare al menù?(y/n)
Premere un tasto per continuare
```

Figura 15: Gestione finale clustering

Domandando all'utente se si vuole ritornare al menù principale, si gestisce la chiusura del programma e, allo stesso tempo, il ritorno al menù principale, ossia alla situazione iniziale.

Inoltre, tale stratagemma previene lo spiacevole inconveniente per cui si è obbligati a forzare l'interruzione dell'esecuzione.