



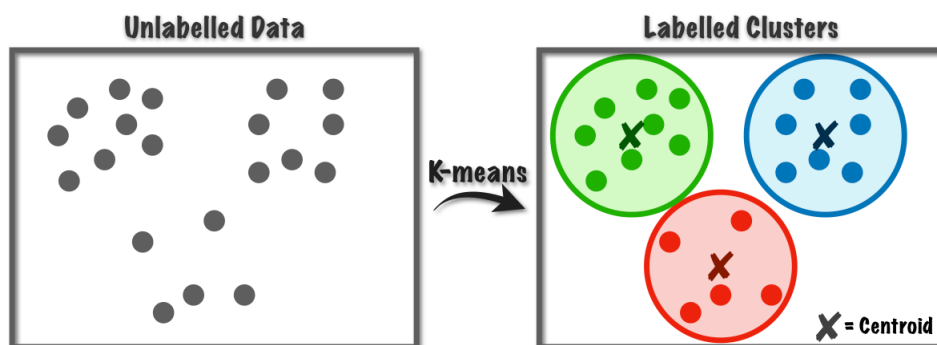
UNIVERSITÀ
DEGLI STUDI DI BARI
ALDO MORO

Dipartimento di Informatica

Corso di laurea in Informatica

Metodi Avanzati di Programmazione

Progetto K-means: Documentazione



Progetto di:

Davide Cirilli (760412) d.cirilli2@studenti.uniba.it

Mattia Curri (758306) m.curri8@studenti.uniba.it

Emanuele Fontana (758344) emanuele.fontana7@studenti.uniba.it

Anno Accademico 2022-2023

Indice

1	Introduzione	2
2	Implementazioni	3
2.1	Progetto Base	3
2.2	Estensione	3
3	Installazione e avvio	4
3.1	Requisiti Server	4
3.2	Requisiti Client (CLI)	4
3.3	Requisiti Client (App)	4
3.4	Installazione APK	4
3.5	Avvio server	4
3.6	Avvio client (CLI)	4
3.7	Avvio client (App)	4
4	Esempi di test (CLI)	5

1 Introduzione

Il programma sviluppato fa uso dell'algoritmo di clustering ***K-means*** per analizzare informazioni estratte da tabelle di un database, sfruttando il servizio MySQL.

Il *K-means* è un algoritmo di clustering, una tecnica di apprendimento non supervisionato utilizzata per suddividere un insieme di dati in gruppi omogenei chiamati *cluster*. L'obiettivo del *K-means* è di assegnare ogni dato al cluster più vicino, in modo che i punti all'interno di ciascun cluster siano simili tra loro e i punti tra cluster diversi siano diversi. Ecco come funziona l'algoritmo *K-means*:

1. **Inizializzazione:** Si inizia scegliendo il numero k desiderato di *cluster*. L'algoritmo crea k partizioni e assegna casualmente k punti come centroidi iniziali, uno per partizione. Un centroide rappresenta il centro del *cluster*.
2. **Assegnazione:** Ciascuna transazione viene assegnata al suo *cluster*. L'appartenenza dipende dalla distanza della transazione dal centroide del *cluster*, l'obiettivo è minimizzare la distanza tra centroidi e transazione.
3. **Aggiornamento dei centroidi:** Una volta assegnati tutti i punti ai *cluster*, i centroidi vengono aggiornati calcolando la media delle posizioni dei punti all'interno di ciascun *cluster*. Questa media diventa il nuovo centroide per il *cluster* corrispondente.
4. **Ripetizione:** I passi 2 e 3 vengono ripetuti fino a quando almeno una transazione cambia *cluster* di appartenenza.
5. **Risultato:** Alla fine delle iterazioni si ottiene un insieme di k *cluster* e ogni transazione sarà stata assegnata al *cluster* più vicino.

È importante notare che il risultato finale può variare a seconda della scelta iniziale dei centroidi.

2 Implementazioni

2.1 Progetto Base

La versione base del progetto consiste in un'architettura client/server che permette all'utente lo studio di cluster. Il server dovrà essere eseguito su una macchina con un database MySQL in esecuzione. Il servizio sarà raggiungibile sulla porta 8080, e potrà comunicare con diversi client contemporaneamente.

I servizi offerti dal server all'utente tramite il client CLI sono i seguenti:

- **scoperta di cluster** fornendo al server il nome della tabella presente nel database ed il numero di cluster da scoprire
- **salvataggio dei cluster** generati nella macchina dove il server viene eseguito. Il salvataggio avverrà in automatico alla generazione di nuovi cluster
- **lettura dei centroidi dei cluster** fornendo al server il path del file in cui sono salvati i centroidi da recuperare

Il server salverà informazioni relativi agli errori nel file di logging mentre nell'interfaccia CLI notificherà l'utente quando la comunicazione si interrompe o se vi sono problemi con gli input dati. Il client da riga di comando permette di collegarsi ad una macchina che sta eseguendo un'istanza del server. L'utente nel menù potrà scegliere tra due opzioni: scoperta di cluster e lettura da file.

2.2 Estensione

3 Installazione e avvio

3.1 Requisiti Server

Per utilizzare il server è necessario:

- Installare MySQL sul proprio computer
- Eseguire lo script sql presente nel percorso ”\KmeansServer\out\artifacts\KmeansServer.jar”
Esempio: \. C:\Users\acurr\Desktop\KmeansBase\KmeansServer\out\artifacts\KmeansServer
Digitando questo comando sulla shell di MySQL sarà creato il database e le tabelle necessarie.
- Installare JRE 8
- Installare JDK 19.0.2(?)

3.2 Requisiti Client (CLI)

Per utilizzare il client è necessario:

- Installare JRE 8
- Server in ascolto

3.3 Requisiti Client (App)

3.4 Installazione APK

3.5 Avvio server

Per avviare il server e' necessario eseguire il file *server.bat* nella stessa cartella di *server.jar*. In alternativa è possibile avviarlo da riga di comando tramite il comando *java -jar server.jar*.

3.6 Avvio client (CLI)

Per avviare il client da riga di comando è necessario eseguire il file *client.bat* nella stessa cartella del file *client.jar*. Questa modalità di avvio conatterà il client ad un server in esecuzione sulla propria macchina sulla porta 8080. Per specificare un altro server a cui connettersi è necessario avviare il client da riga di comando tramite il comando *java -jar client.jar indirizzo_ip porta* o inserendo tale comando nel file batch.

3.7 Avvio client (App)

4 Esempi di test (CLI)

1. Esecuzione del client con server offline

```
PS C:\Users\fonta\Desktop\Kmeans\Kmeans\KmeansClient\src> Java MainTest 127.0.0.1 8080
addr = /127.0.0.1
Connection refused: connect
```

2. **Esecuzione senza parametri:** il programma deve essere avviato fornendo come parametri l'indirizzo IP/DNS del server e la porta logica. Se si lavora con la stessa macchina si può inserire come primo parametro localhost, 127.0.0.1 (indirizzo IPv4 locale) oppure ::1 (indirizzo IPv6 locale).

```
PS C:\Users\fonta\Desktop\Kmeans\Kmeans\KmeansClient\src> Java MainTest
Errore: inserire ip e porta come argomenti
```

3. **Esecuzione con porta errata:** la porta logica è un numero a 16 bit, dunque un decimale compreso tra 0 e 65.535.

```
PS C:\Users\fonta\Desktop\Kmeans\Kmeans\KmeansClient\src> Java MainTest 127.0.0.1 das
Errore: la porta deve essere un numero
```

4. **Esecuzione con IP/DNS inesistente:** in questa esecuzione il server non esiste e dunque il client non riesce a connettersi. In particolare il programma si connette a un server DNS per convertire l'indirizzo fornito in un indirizzo IP ma tale indirizzo non è registrato e dunque si ha errore.

```
PS C:\Users\fonta\Desktop\Kmeans\Kmeans\KmeansClient\src> Java MainTest www.sdaiksdai.it 8080
No such host is known (www.sdaiksdai.it)
```

5. **Esecuzione con parametri corretti:** quando il programma viene eseguito nelle condizioni funzionali (quindi server avviato e parametri validi) vengono stampati a schermo indirizzo IP e porta del server, seguiti dalla porta che sta usando il processo per la comunicazione. Viene poi mostrato all'utente un menù con due opzioni. L'opzione (1) permetterà al client di caricare un cluster di dati che è stato serializzato sul server come un file, mentre l'opzione (2) consente di creare un nuovo cluster di dati mediante l'algoritmo del K-means. In caso di input non validi viene chiesto all'utente di reinserire finché non si avrà un'opzione valida.

```
PS C:\Users\fonta\Desktop\Kmeans\Kmeans\KmeansClient\src> Java MainTest 127.0.0.1 8080
addr = /127.0.0.1
Socket[addr=/127.0.0.1,port=8080,localport=2784]
Scegli una opzione
(1) Carica Cluster da File
(2) Carica Dati
Risposta:
```

```
(1) Carica Cluster da File
(2) Carica Dati
Risposta:dsa
LETTO VALORE NON INTERO, REINSERIRE
43
(1) Carica Cluster da File
(2) Carica Dati
Risposta:
```

- **Opzione (1):** Viene chiesto all'utente di inserire il nome del database, della tabella e del numero di cluster creati. Il server andrà dunque a cercare il file corrispondente a tali informazioni e, se non trovato, manderà un messaggio di errore all'utente (come nell'esempio). Viene chiesto all'utente se vuole tornare al menù oppure terminare l'esecuzione del programma. Se viene digitato *n* il programma termina. Nel caso in cui il file corrispondente alle richieste esista viene inviato al client il contenuto del file, ovvero i centroidi dei cluster.

```
Risposta:1
Nome database:ciao
Nome tabella:das
Numero di cluster:43
Impossibile caricare il salvataggio
Vuoi scegliere una nuova operazione da menu?(y/n)
```

```
Scegli una opzione
(1) Carica Cluster da File
(2) Carica Dati
Risposta:1
Nome database:MapDB
Nome tabella:playtennis
Numero di cluster:9
Centroid=(rain 13.5525 normal weak yes)
Centroid=(overcast 0.1 normal strong yes)
Centroid=(sunny 13.0 high weak no)
Centroid=(sunny 12.5 normal strong yes)
Centroid=(sunny 0.1 normal weak yes)
Centroid=(sunny 30.3 high weak no)
Centroid=(sunny 30.3 high strong no)
Centroid=(rain 6.25 high strong no)
Centroid=(overcast 21.25 high strong yes)
Vuoi scegliere una nuova operazione da menu?(y/n)
```

- **Opzione (2):** Viene chiesto all'utente se vuole usare i valori di default o meno:
 - **Valori di default:** i valori di default sono:
 - localhost → server database
 - 3306 → porta database
 - MapDB → nome database
 - playtennis → nome tabella
 - MapUser → nome utente
 - password di MapUser

```
Scegli una opzione
(1) Carica Cluster da File
(2) Carica Dati
Risposta:2
Vuoi usare dei valori di default per il database? (y/n)
```

Nel caso di risposta affermativa viene chiesto all'utente il numero dei cluster. Una volta confermati (se questi sono validi) il server eseguirà l'algoritmo di K-means e invierà il risultato al client. Viene poi chiesto se si vuole riprendere l'esecuzione sullo stesso dataset o meno.

```
Vuoi usare dei valori di default per il database? (y/n)y
Numero di cluster:5
Clustering output:Numero di iterazioni: 2
0:Centroid=(rain 13.0 high weak no)
Examples:
[rain 13.0 high weak yes] dist=1.0
[sunny 13.0 high weak no] dist=1.0
AvgDistance=1.0

1:Centroid=(rain 8.333333333333334 high strong no)
Examples:
[rain 0.0 normal strong no] dist=1.275027502750275
[overcast 12.5 high strong yes] dist=2.1375137513751374
[rain 12.5 high strong no] dist=0.1375137513751375
AvgDistance=1.1833516685001833

2:Centroid=(overcast 17.802500000000002 normal weak yes)
Examples:
[overcast 30.0 high weak yes] dist=1.4025577557755775
[rain 0.0 normal weak yes] dist=1.5875412541254126
[rain 12.0 normal weak yes] dist=1.1915016501650166
[overcast 29.21 normal weak yes] dist=0.3764851485148515
AvgDistance=1.1395214521452146

3:Centroid=(sunny 30.3 high strong no)
Examples:
[sunny 30.3 high weak no] dist=1.0
[sunny 30.3 high strong no] dist=0.0
AvgDistance=0.5

4:Centroid=(sunny 4.233333333333333 normal strong yes)
Examples:
[overcast 0.1 normal strong yes] dist=1.1364136413641364
[sunny 0.1 normal weak yes] dist=1.1364136413641364
[sunny 12.5 normal strong yes] dist=0.27282728272827284
AvgDistance=0.8485515218188485

Vuoi ripetere l'esecuzione?(y/n)
```

Se l'utente non vuole ripetere l'esecuzione sul dataset ha due scelte:

- (a) Tornare al menù
- (b) Terminare l'esecuzione

```
Vuoi ripetere l'esecuzione?(y/n)n
Vuoi scegliere una nuova operazione da menu?(y/n)y
Scegli una opzione
(1) Carica Cluster da File
(2) Carica Dati
Risposta:|
```

- **Valori non di default:** l'utente può scegliere di non usare i valori di default. In quel caso deve inserire tutte le informazioni necessarie. In caso di valori non validi verrà segnalato l'errore all'utente, il quale sceglierà se proseguire

con i valori di default oppure se riprovare a inserire dei valori personalizzati. L'uso di valori personalizzati permette di utilizzare dataset già presenti nel server del database.

```
Scegli una opzione
(1) Carica Cluster da File
(2) Carica Dati
Risposta:2
Vuoi usare dei valori di default per il database? (y/n)n
Inserisci l'indirizzo ip/DNS del database (localhost / 127.0.0.1 se il database si trova nel server a cui si è connessi):127.0.0.1
Inserisci la porta del database (3306 se il database si trova nel server a cui si è connessi):8076
Inserisci il nome del database:MapDB
Inserisci il nome tabella:playtennis
Inserisci il nome utente del database:MapUser
Inserisci la password del database:map
SI E' VERIFICATO UN ERRORE DURANTE L'INTERROGAZIONE AL DATABASE -> SERVER NON ESISTENTE
Vuoi usare dei valori di default per il database? (y/n)n
Inserisci l'indirizzo ip/DNS del database (localhost / 127.0.0.1 se il database si trova nel server a cui si è connessi):127.0.0.1
Inserisci la porta del database (3306 se il database si trova nel server a cui si è connessi):3306
Inserisci il nome del database:osddsd
Inserisci il nome tabella:playtennis
Inserisci il nome utente del database:MapUser
Inserisci la password del database:map
SI E' VERIFICATO UN ERRORE DURANTE L'INTERROGAZIONE AL DATABASE -> DATABASE NON ESISTENTE
Vuoi usare dei valori di default per il database? (y/n)n
Inserisci l'indirizzo ip/DNS del database (localhost / 127.0.0.1 se il database si trova nel server a cui si è connessi):127.0.0.1
Inserisci la porta del database (3306 se il database si trova nel server a cui si è connessi):3306
Inserisci il nome del database:MapDB
Inserisci il nome tabella:playtennis
Inserisci il nome utente del database:MpaUS
Inserisci la password del database:DSa
SI E' VERIFICATO UN ERRORE DURANTE L'INTERROGAZIONE AL DATABASE -> USER e/o PASSWORD ERRATI
Vuoi usare dei valori di default per il database? (y/n)
```

Ogni volta che l'utente richiede la creazione di un dataset al server, quest'ultimo serializzerà i cluster in un file con nome del tipo: ***NomedatabaseNometabellaNumerocluster.dat***.

```
Inserisci l'indirizzo ip/DNS del database (localhost / 127.0.0.1 se il database si trova nel server a cui si è connessi):127.0.0.1
Inserisci la porta del database (3306 se il database si trova nel server a cui si è connessi):3306
Inserisci il nome del database:ditto
Inserisci il nome tabella:coldaio
Inserisci il nome utente del database:root
Inserisci la password del database:Ciao1234/
Numero di cluster:2
Clustering output:Numero di iterazioni: 2
0:Centroid=(00001 DSM 12kw esterna gas 20.0)
Examples:
{00001 DSM 12kw esterna gas 20.0} dist=0.0
AvgDistance=0.0

1:Centroid=(00002 MFS 13kw esterna pellet 0.0)
Examples:
{00002 MMF 13kw interna pellet 0.0} dist=2.0
{00003 MFS 33kw esterna pellet 0.0} dist=2.0
AvgDistance=2.0

Vuoi ripetere l'esecuzione?(y/n)
```