

EPMiner

1 Introduzione del progetto

EPMiner è un'applicazione, con interfaccia a linea di comando, che permette, mediante l'utilizzo dell'algoritmo Apriori, di scoprire pattern frequenti e pattern emergenti partendo da due tabelle differenti.

Con il termine **pattern frequenti** si intende l'insieme di item o pattern che occorrono con una frequenza minima chiamata **minimo supporto**, mentre con il termine **pattern emergenti** si intende l'insieme di pattern o item che occorrono con una frequenza minima e che hanno un **grow rate** o tasso di crescita superiore rispetto al passato.

2 Algoritmo Apriori

L'algoritmo utilizzato per la scoperta dei pattern è l'algoritmo Apriori. L'algoritmo Apriori è un classico algoritmo di ricerca delle associazioni. È utilizzato per la generazione dei pattern frequenti, per approssimazioni successive, a partire dagli itemset con un solo elemento. In sintesi, il presupposto teorico su cui si basa l'algoritmo parte dalla considerazione che, se un insieme di oggetti (itemset) è frequente, allora anche tutti i suoi sottoinsiemi sono frequenti, ma se un itemset non è frequente, allora neanche gli insiemi che lo contengono sono frequenti.

Un ambito dove questo algoritmo trova grande applicabilità è il market/basket problem. Per ricavare le associazioni viene impiegato un approccio bottom up, dove i sottoinsiemi frequenti sono costruiti aggiungendo un item per volta (generazione dei candidati); i gruppi di candidati sono successivamente verificati sui dati e l'algoritmo termina quando non ci sono ulteriori estensioni possibili.

I dati in input del problema sono i seguenti:

- un database di transizioni target DTarget
- un database di transizioni di background Dbackground
- un valore minimo di supporto ($0 < \text{minS} \leq 1$)
- un valore minimo di grow rate ($\text{minGr} \geq 1$)

L'obiettivo è trovare i pattern che siano frequenti (cioè con supporto maggiore o uguale a minS) in DTarget e emergenti (grow rate maggiore o uguale a minGr) rispetto a Dbackground. L'algoritmo si divide in due fasi: nella prima fase si scoprono i pattern di lunghezza k a partire dai pattern frequenti di lunghezza k-1, utilizzando la tabella DTarget. Nella seconda fase, si scoprono i pattern emergenti, calcolando i grow rate dei pattern frequenti in DTarget e in Dbackground e, confrontando i valori ottenuti, si selezionano i pattern con grow rate superiore alla soglia prefissata.

Di seguito viene riportato lo pseudo-codice dell'algoritmo:

```

frequentPatternDiscovery(DTarget,minS) → FP
begin
  FP= ∅
  L1= {1-item che compaiono in minS×|D| transazioni di DTarget}
  K=2
  while LK-1 ≠ ∅ do
    begin
      CK= candidati generati da LK-1 aggiungendo un nuovo item
      LK=∅
      for each (p ∈ CK) do
        if (supporto(p, DTarget) ≥ minS) then
          LK=LK∪p
      FP=FP∪LK
      K=K+1
    end
  return FP
end

```

Figura 1: Pseudo-codice per la scoperta di pattern frequenti.

```

EPDiscovery(DBackground,FP,minGr) → EP
begin
  EP= ∅
  for each (p ∈ FP) do
    begin
      if (growrate(p,DBackground) ≥ minGr) then
        EP=EP∪ p
      end
    end
  return EP
end

```

Figura 2: Pseudo-codice per la scoperta di pattern emergenti.

3 Architettura del progetto

Il progetto EPMiner è stato sviluppato con l'IDE IntelliJ IDEA. Esso presenta una architettura di tipo client-server, in cui il client può mandare richieste al server per i seguenti servizi:

- scoprire dei nuovi pattern nel database
- caricare pattern salvati nell'archivio

Il client deve specificare i valori di minimo supporto, grow rate e le tabelle di target e background.

Il server elabora le richieste accedendo al database tramite il DBMS MySQL. Il server viene avviato sulla porta 8080, pertanto il client deve essere avviato con i parametri "localhost" e la porta 8080. È possibile avviare server e client con parametri diversi.

4 Guida all'installazione

Di seguito sono riportate le linee guida da seguire per il corretto funzionamento del server:

1. installare la Java Runtime Environment, versione 16 o superiore
2. installare il DBMS MySQL, versione 5.7 o superiore
3. dopo aver avviato il server mysql, eseguire lo script "mysqlScript" presente nel seguente percorso: "EPMiner\Versione base\Server\mysqlScript.sql"

Le linee guida da seguire per il funzionamento del client sono:

1. installare la Java Runtime Environment, versione 16 o superiore
2. avviare il server prima di avviare il client

5 Guida utente

5.1 Avvio del server tramite batch

Per il corretto funzionamento del programma è necessario avviare prima il server. Per avviare il server bisogna eseguire il file serverBatch.bat, presente nel seguente percorso: "EPMiner\Versione base\Server\serverBatch.bat", tramite uno dei seguenti metodi:

- fare doppio clic con il mouse sul file batch
- eseguire il file batch tramite riga di comando (es. cmd o PowerShell)

In questo modo il server verrà avviato senza parametri sulla porta 8080. Se si ha necessità di avviare il server con **parametri** si può modificare il file serverBatch tramite un qualsiasi editor di testo, aggiungendo i parametri come segue:

```
1 cd build/libs
2
3 java -jar Server-1.0-SNAPSHOT.jar 45621
4 PAUSE
```

In questo modo il server verrà avviato sulla porta 45621 (se libera).

5.2 Avvio del client tramite batch

Per avviare il client bisogna eseguire il file clientBatch.bat presente nel seguente percorso: "EPMiner\Versione base\Client\clientBatch.bat", tramite uno dei seguenti metodi:

- fare doppio clic con il mouse sul file batch
- eseguire il file batch tramite riga di comando (es. cmd o PowerShell)

In questo modo il client verrà avviato senza parametri ed utilizzerà come indirizzo del server "localhost" e come porta "8080". Se si ha necessità di avviare il client con **parametri** si può modificare il file clientBatch tramite un qualsiasi editor di testo, aggiungendo i parametri come segue:

```
1 cd build/libs
2
3 java -jar Client-1.0-SNAPSHOT.jar 195.42.3.5 45621
4 PAUSE
```

In questo modo il client verrà avviato con i parametri "195.42.3.5" e "45621".

5.3 Esempi di test

Di seguito sono riportati esempi di test.

Se la connessione con il client è andata a buon fine, il server visualizzerà i seguenti messaggi:

```
Server avviato
Connessione di socket Socket[addr=/127.0.0.1,port=61974,localport=8080]
Nuovo client connesso
in attesa dell'operazione da eseguire..
```

Il client invece, dopo aver mostrato i dati relativi alla connessione con il server, chiederà all'utente di effettuare una scelta:

```
addr = localhost/127.0.0.1
port=8080
Socket[addr=localhost/127.0.0.1,port=8080,localport=61974]

Scegli una opzione:
1:Nuova scoperta
2:Risultati in archivio
```

Scegliendo l'opzione "Nuova scoperta", all'utente verrà chiesto di inserire i dati per la scoperta dei pattern emergenti:

```
Scegli una opzione:
1:Nuova scoperta
2:Risultati in archivio
1
Inserire valore minimo supporto (minsup>0 e minsup<=1):
0.3
Inserire valore minimo grow rate (minGr>=1):
1
Tabella target:
playtennistarget
Tabella background:
playtennisbackground
Frequent patterns:
1:(outlook=rain)[0.375]
2:(outlook=sunny)[0.375]
3:(temperature in [0.0,6.06[)][0.375]
4:(temperature in [24.24,30.3[)][0.375]
5:(umidity=normal)[0.375]
6:(wind=strong)[0.375]
7:(outlook=sunny) AND (umidity=high)[0.375]
8:(outlook=sunny) AND (play=no)[0.375]
9:(temperature in [0.0,6.06[) AND (umidity=normal)[0.375]
10:(temperature in [24.24,30.3[) AND (umidity=high)[0.375]
11:(umidity=high) AND (outlook=sunny)[0.375]
12:(umidity=high) AND (temperature in [24.24,30.3[)][0.375]
13:(umidity=high) AND (play=no)[0.375]
14:(umidity=normal) AND (temperature in [0.0,6.06[)][0.375]
15:(wind=weak) AND (play=yes)[0.375]
```

```

16:(play=no) AND (outlook=sunny)[0.375]
17:(play=no) AND (umidity=high)[0.375]
18:(play=yes) AND (wind=weak)[0.375]
19:(outlook=sunny) AND (umidity=high) AND (play=no)[0.375]
20:(outlook=sunny) AND (play=no) AND (umidity=high)[0.375]
21:(umidity=high) AND (outlook=sunny) AND (play=no)[0.375]
22:(umidity=high) AND (play=no) AND (outlook=sunny)[0.375]
23:(play=no) AND (outlook=sunny) AND (umidity=high)[0.375]
24:(play=no) AND (umidity=high) AND (outlook=sunny)[0.375]
25:(play=no)[0.5]
26:(play=yes)[0.5]
27:(umidity=high) AND (wind=weak)[0.5]
28:(wind=weak) AND (umidity=high)[0.5]
29:(umidity=high)[0.625]
30:(wind=weak)[0.625]

Emerging patterns
1:(outlook=rain)[0.375][1.125]
2:(outlook=sunny)[0.375][1.125]
3:(wind=weak)[0.625][1.25]
4:(umidity=high)[0.625][1.875]
5:(temperature in [0.0,6.06[)[0.375][2.25]
6:(temperature in [24.24,30.3[)[0.375][2.25]
7:(temperature in [0.0,6.06[) AND (umidity=normal)[0.375][2.25]
8:(umidity=high) AND (play=no)[0.375][2.25]
9:(umidity=normal) AND (temperature in [0.0,6.06[)[0.375][2.25]
10:(play=no) AND (umidity=high)[0.375][2.25]
11:(play=no)[0.5][3.0]
12:(outlook=sunny) AND (umidity=high)[0.375][Infinity]
13:(outlook=sunny) AND (play=no)[0.375][Infinity]
14:(temperature in [24.24,30.3[) AND (umidity=high)[0.375][Infinity]
15:(umidity=high) AND (outlook=sunny)[0.375][Infinity]
16:(umidity=high) AND (temperature in [24.24,30.3[)[0.375][Infinity]
17:(play=no) AND (outlook=sunny)[0.375][Infinity]
18:(outlook=sunny) AND (umidity=high) AND (play=no)[0.375][Infinity]
19:(outlook=sunny) AND (play=no) AND (umidity=high)[0.375][Infinity]
20:(umidity=high) AND (outlook=sunny) AND (play=no)[0.375][Infinity]
21:(umidity=high) AND (play=no) AND (outlook=sunny)[0.375][Infinity]
22:(play=no) AND (outlook=sunny) AND (umidity=high)[0.375][Infinity]
23:(play=no) AND (umidity=high) AND (outlook=sunny)[0.375][Infinity]
24:(umidity=high) AND (wind=weak)[0.5][Infinity]
25:(wind=weak) AND (umidity=high)[0.5][Infinity]

Vuoi ripetere?(s/n)

```

Scegliendo l'opzione "**Risultati in archivio**", verrà chiesto all'utente di inserire i dati per la ricerca dei dati in archivio:

```

Scegli una opzione:
1:Nuova scoperta
2:Risultati in archivio
2
Inserire valore minimo supporto (minsup>0 e minsup<=1):
0.3
Inserire valore minimo grow rate (minGr>=1):
2
Tabella target:
playtennistarget
Tabella background:
playtennisbackground
Frequent patterns:
1:(outlook=rain)[0.375]
2:(outlook=sunny)[0.375]
3:(temperature in [0.0,6.06[)[0.375]
4:(temperature in [24.24,30.3[)[0.375]
5:(umidity=normal)[0.375]
6:(wind=strong)[0.375]
7:(outlook=sunny) AND (umidity=high)[0.375]
8:(outlook=sunny) AND (play=no)[0.375]
9:(temperature in [0.0,6.06[) AND (umidity=normal)[0.375]
10:(temperature in [24.24,30.3[) AND (umidity=high)[0.375]
11:(umidity=high) AND (outlook=sunny)[0.375]
12:(umidity=high) AND (temperature in [24.24,30.3[)[0.375]
13:(umidity=high) AND (play=no)[0.375]
14:(umidity=normal) AND (temperature in [0.0,6.06[)[0.375]
15:(wind=weak) AND (play=yes)[0.375]
16:(play=no) AND (outlook=sunny)[0.375]
17:(play=no) AND (umidity=high)[0.375]
18:(play=yes) AND (wind=weak)[0.375]
19:(outlook=sunny) AND (umidity=high) AND (play=no)[0.375]
20:(outlook=sunny) AND (play=no) AND (umidity=high)[0.375]
21:(umidity=high) AND (outlook=sunny) AND (play=no)[0.375]
22:(umidity=high) AND (play=no) AND (outlook=sunny)[0.375]
23:(play=no) AND (outlook=sunny) AND (umidity=high)[0.375]
24:(play=no) AND (umidity=high) AND (outlook=sunny)[0.375]
25:(play=no)[0.5]
26:(play=yes)[0.5]
27:(umidity=high) AND (wind=weak)[0.5]
28:(wind=weak) AND (umidity=high)[0.5]
29:(umidity=high)[0.625]
30:(wind=weak)[0.625]

```

```

Emerging patterns
1:(temperature in [0.0,6.06[)[0.375][2.25]
2:(temperature in [24.24,30.3[)[0.375][2.25]
3:(temperature in [0.0,6.06[) AND (umidity=normal)[0.375][2.25]
4:(umidity=high) AND (play=no)[0.375][2.25]
5:(umidity=normal) AND (temperature in [0.0,6.06[)[0.375][2.25]
6:(play=no) AND (umidity=high)[0.375][2.25]
7:(play=no)[0.5][3.0]
8:(outlook=sunny) AND (umidity=high)[0.375][Infinity]
9:(outlook=sunny) AND (play=no)[0.375][Infinity]
10:(temperature in [24.24,30.3[) AND (umidity=high)[0.375][Infinity]
11:(umidity=high) AND (outlook=sunny)[0.375][Infinity]
12:(umidity=high) AND (temperature in [24.24,30.3[)[0.375][Infinity]
13:(play=no) AND (outlook=sunny)[0.375][Infinity]
14:(outlook=sunny) AND (umidity=high) AND (play=no)[0.375][Infinity]
15:(outlook=sunny) AND (play=no) AND (umidity=high)[0.375][Infinity]
16:(umidity=high) AND (outlook=sunny) AND (play=no)[0.375][Infinity]
17:(umidity=high) AND (play=no) AND (outlook=sunny)[0.375][Infinity]
18:(play=no) AND (outlook=sunny) AND (umidity=high)[0.375][Infinity]
19:(play=no) AND (umidity=high) AND (outlook=sunny)[0.375][Infinity]
20:(umidity=high) AND (wind=weak)[0.5][Infinity]
21:(wind=weak) AND (umidity=high)[0.5][Infinity]

Vuoi ripetere?(s/n)

```

Nel caso in cui venga inserita una tabella inesistente, verrà visualizzato un messaggio di errore:

```

Scegli una opzione:
1:Nuova scoperta
2:Risultati in archivio
1
Inserire valore minimo supporto (minsup>0 e minsup<=1):
0.3
Inserire valore minimo grow rate (minGr>=1):
2
Tabella target:
tab1
Tabella background:
tab2
Errore
Errore
Vuoi ripetere?(s/n)

```

5.4 Avvio del server tramite IDE

È possibile avviare il server tramite l'IDE IntelliJ IDEA. Di seguito sono riportate le linee guida da seguire per avviare il server tramite IDE:

1. importare il progetto Server, presente nella cartella Versione base, in IntelliJ IDEA
2. eseguire il task "build" presente nella cartella "Tasks\build" nella visualizzazione dei task gradle
3. avviare il server cliccando col tasto destro sulla classe "MultiServer" e scegliere l'opzione "Run MultiServer.main()"

È possibile avviare il client tramite l'IDE IntelliJ IDEA. Di seguito sono riportate le linee guida da seguire per avviare il client tramite IDE:

- ## 6 Diagrammi UML

[illegible]

Diagramma delle classi del package Data:

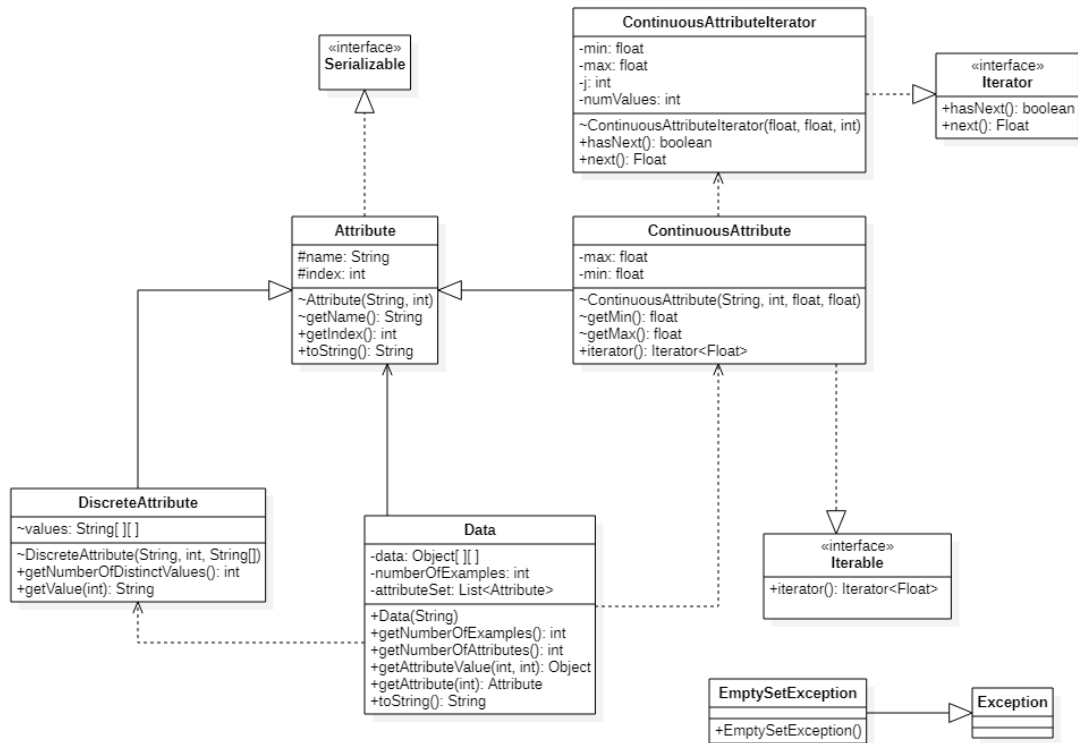


Diagramma delle classi del package Utility:

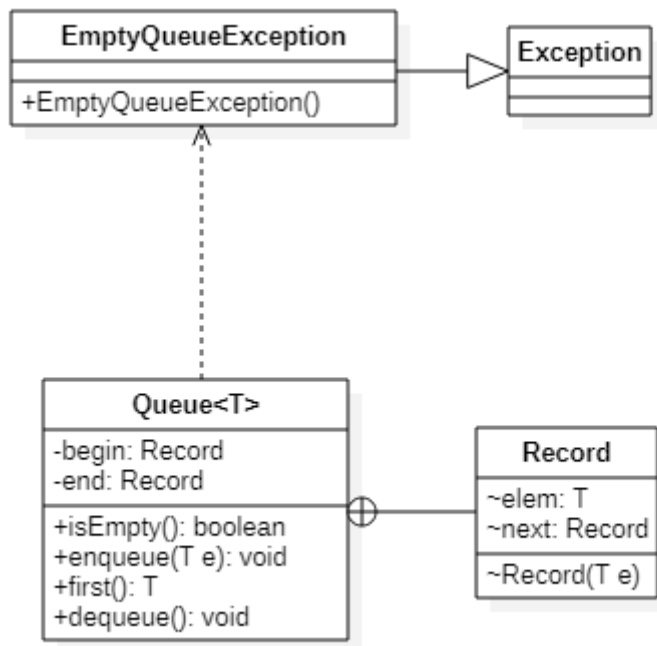


Diagramma delle classi del package Database:

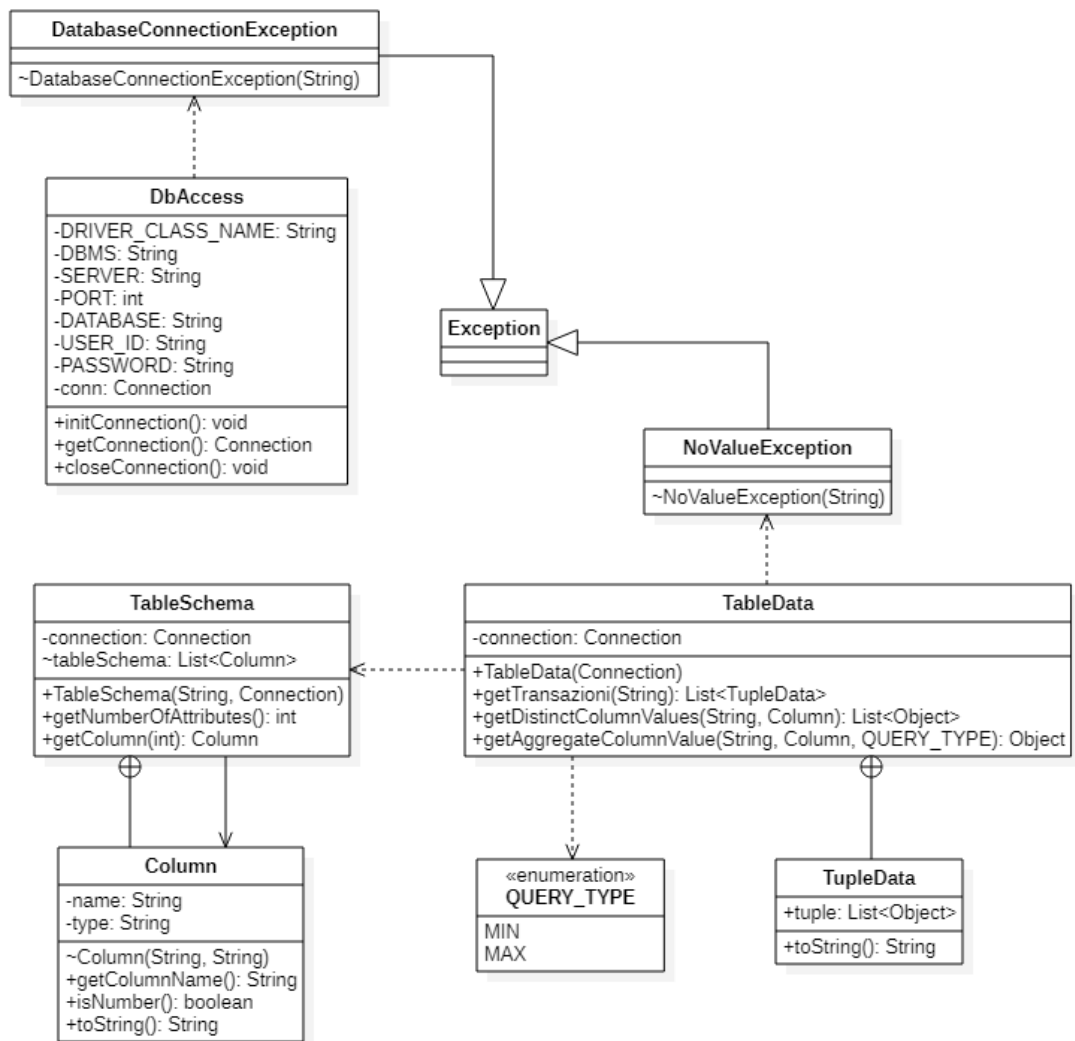


Diagramma del package Client:

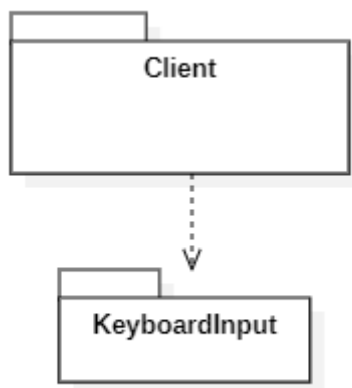
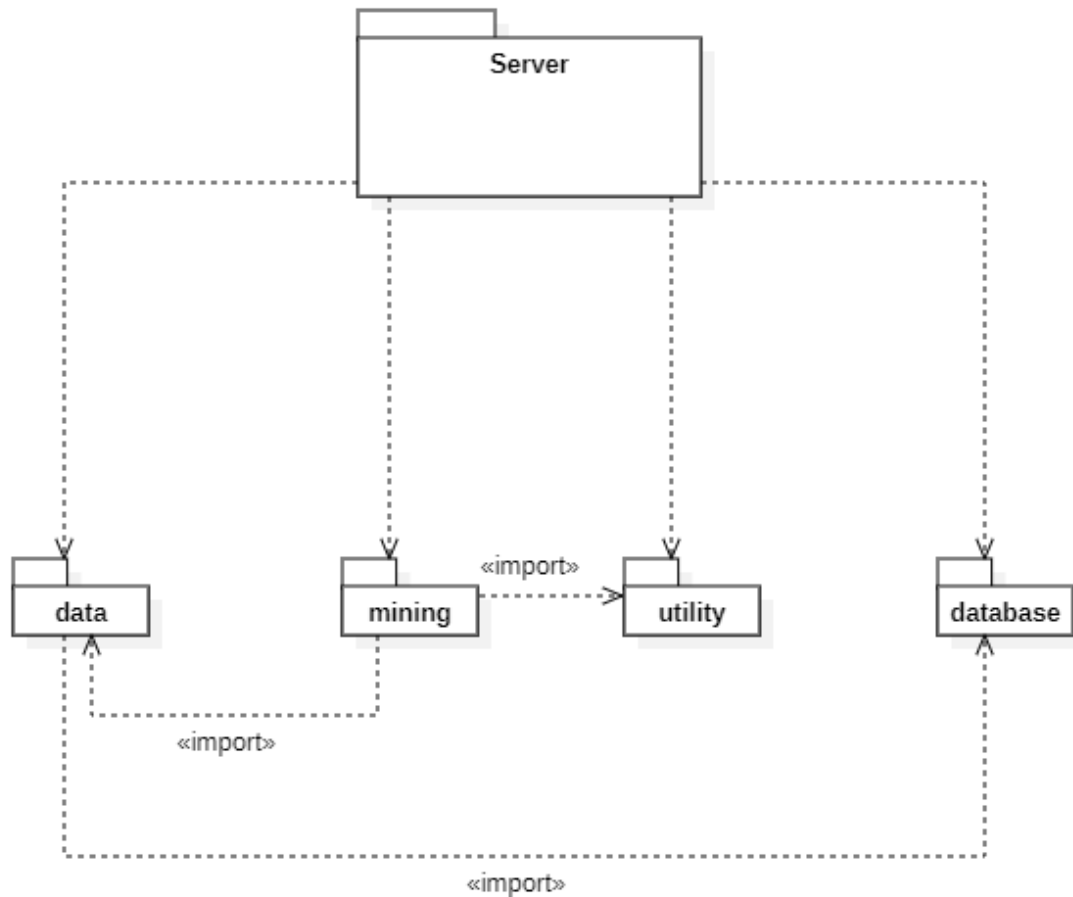


Diagramma del package Server:



7 Javadoc

È possibile visualizzare la documentazione del client e del server tramite le pagine presenti nei seguenti percorsi:

"EPMiner\Versione base\Client\build\docs\javadoc\allclasses-index"

"EPMiner\Versione base\Server\build\docs\javadoc\allclasses-index"

8 Studenti del gruppo

Il progetto è stato realizzato da:

- Marco Angelo Lillo - MAT: 717683 - m.lillo21@studenti.uniba.it
- Daniele Cecca - MAT: 718588 - d.cecca1@studenti.uniba.it
- Ferrulli Francesco - MAT: 716836 - f.ferrulli14@studenti.uniba.it

¹

¹È possibile scaricare il progetto EPMiner dalla repository <https://github.com/Ferru2000/EPMiner>