

User Guide

Prerequisiti

Per il corretto funzionamento è necessario avere un computer con Windows e i due applicativi (client e server). Il client è anche dotato di un'interfaccia grafica per l'utente.

Al fine di eseguire l'applicazione, si rende necessaria l'installazione (se non sono già presenti nel sistema) di MySQL e la JRE 8.

E possibile effettuare il download di entrambi dai seguenti link:

MySQL:

<https://dev.mysql.com/get/Downloads/MySQLInstaller/mysql-installercommunity-8.0.18.0.msi>

JRE 8:

<https://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

Server

Prima dell'esecuzione del server è necessario creare database, utente e popolare le tabelle del db attraverso uno script SQL chiamato "MAPDB.SQL" contenuto nel percorso \Regression Tree Learner\Distribution\server

Client

Il client necessita della JRE 8 installata sulla macchina e del Server già avviato e in attesa di una connessione.

GUI-Client

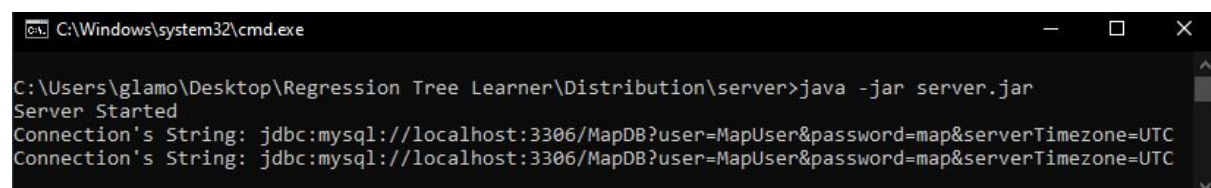
E possibile usare il client anche tramite GUI; sarà sempre necessaria la JRE 8.

N.B. Per il corretto funzionamento dell'applicazione server e client devono essere collegati sotto la stessa rete ed il server deve essere avviato per primo.

Funzionamento Server

Il server viene avviato usando il file batch server.bat, situato in \Regression Tree Learner\Distribution\server.

Una volta avviato il server rimarrà in attesa di connessione da parte di un client. Il server mostrerà a video un messaggio "Server started" e una volta connesso, mostrerà informazioni sulla connessione avvenuta.



```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\server>java -jar server.jar
Server Started
Connection's String: jdbc:mysql://localhost:3306/MapDB?user=MapUser&password=map&serverTimezone=UTC
Connection's String: jdbc:mysql://localhost:3306/MapDB?user=MapUser&password=map&serverTimezone=UTC
```

Funzionamento Client

Il client viene avviato da un file batch denominato "client.bat" che si trova in \Regression Tree Learner\Distribution\client. Di default il client viene avviato con determinati parametri specificati nel file batch.

Una volta avviato, il client mostrerà a video la seguente schermata:

```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\client>java -jar client.jar 127.0.0.1 8080 ^
Socket[addr=/127.0.0.1,port=8080,localport=55468]
Learn Regression Tree from data [1]
Load Regression Tree from archive [2]
```

Da qui potremo scegliere se effettuare il caricamento da database o da file. Scegliendo l'opzione [1], ci verrà chiesto di inserire il nome della tabella.

```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\client>java -jar client.jar 127.0.0.1 8080 ^
Socket[addr=/127.0.0.1,port=8080,localport=55738]
Learn Regression Tree from data [1]
Load Regression Tree from archive [2]
1
Table or File name:
Starting data acquisition phase!
```

Una volta inserito il nome, il programma inizierà la learning phase e ci verrà chiesto di direzionare l'esecuzione mediante l'inserimento del branch.

```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\client>java -jar client.jar 127.0.0.1 8080 ^
Socket[addr=/127.0.0.1,port=8080,localport=55738]
Learn Regression Tree from data [1]
Load Regression Tree from archive [2]
1
Table or File name:
Starting data acquisition phase!
provaC
Starting learning phase!
Starting prediction phase!
0:X=A
1:X=B
```

In base alla scelta e al dataset utilizzato, l'inserimento potrebbe essere richiesto più volte. Infine avremo l'output e la possibilità di ripetere il processo, rispondendo "y" a "would you repeat?", generando così nuovamente l'albero.

Nel caso in cui la risposta sia "n" invece, il client terminerà.

```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\client>java -jar client.jar 127.0.0.1 8080
Socket[addr=/127.0.0.1,port=8080,localport=55738]
Learn Regression Tree from data [1]
Load Regression Tree from archive [2]
1
Table or File name:
Starting data acquisition phase!
provaC
Starting learning phase!
Starting prediction phase!
0:X=A
1:X=B

0
0:Y<=2.0
1:Y>2.0

1
Predicted class:1.5
Would you repeat ? (y/n)
```

Invece, se la scelta dovesse essere la [2], ci verrà chiesto di inserire il nome di un file relativo ad una precedente esecuzione presente su disco.

```
C:\Windows\system32\cmd.exe
C:\Users\glamo\Desktop\Regression Tree Learner\Distribution\client>java -jar client.jar 127.0.0.1 8080
Socket[addr=/127.0.0.1,port=8080,localport=56050]
Learn Regression Tree from data [1]
Load Regression Tree from archive [2]
2
Table or File name:
provaC
Starting prediction phase!
0:X=A
1:X=B

1
Predicted class:10.0
Would you repeat ? (y/n)
```

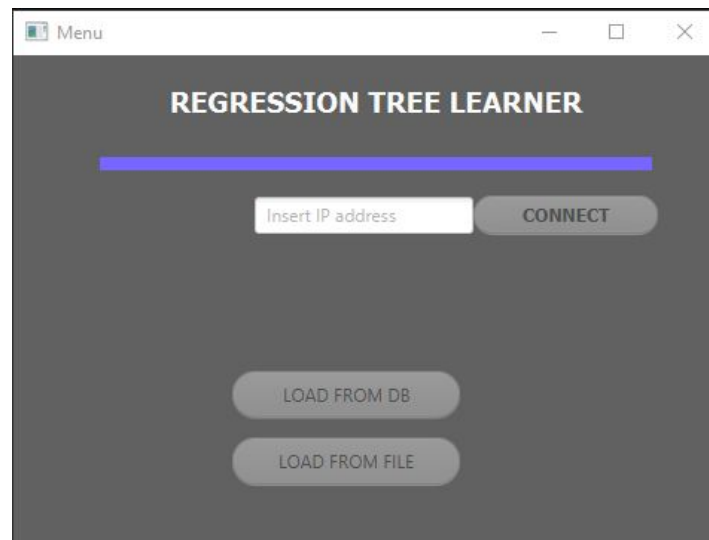
Successivamente, verranno richiesti gli stessi input e mostrato l'output nelle stesse modalità del caso 1.

N.B. Il programma non ammette l'inserimento di table name relativi ad una tabella non esistente o ad un file mancante. L'inserimento verrà ripetuto fin quando il nome del file/tabella non sarà valido.

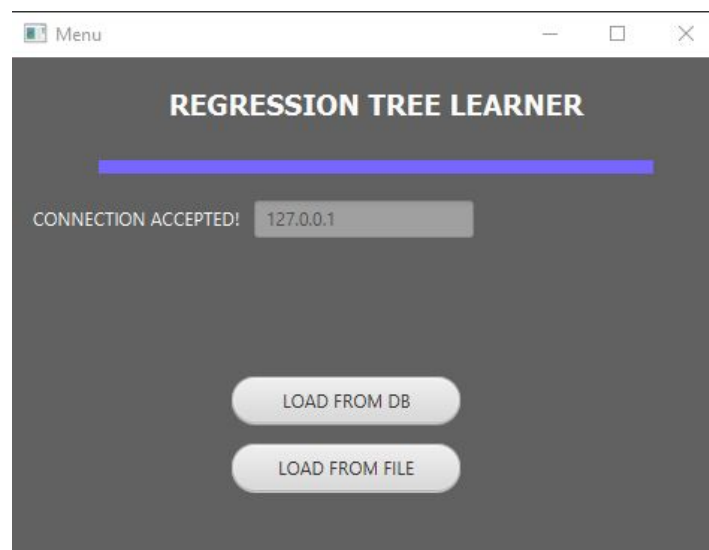
Funzionamento GUI

L'interfaccia grafica viene avviata dal file JAR denominato "GUI.jar" presente in \Regression Tree Learner\Distribution.

Una volta avviata, apparirà una schermata nella quale si potrà inserire l'indirizzo IP del server. Una volta inserito l'IP e premuto il tasto connect, verrà mostrato un messaggio di conferma



L'utente avrà quindi la possibilità di caricare i dati da database o da un file relativo ad una precedente esecuzione



Nel primo caso, una volta inserito il nome della tabella e premuto il tasto “execute”, verrà mostrata una schermata come quella sottostante.

P.S. Anche in questo caso, il programma controllerà la correttezza dell’input e nel caso fosse necessario, farà ripetere l’inserimento fin quando l’input non risulterà corretto.

The screenshot shows a window titled "Load Data From Database". It has a dark grey background. At the top, there are two input fields: "TABLE NAME" with the value "provaC" and "BRANCH" which is empty. To the right of the "BRANCH" field is a small "OK!" button. Below these fields is a large white rectangular area containing the text "GENERATE DECISION TREE" followed by "=====
0:X=A
1:X=B". At the bottom of the window, there are four buttons: "EXECUTE", "NEW EXECUTION", "EXIT", and "MENU". The "EXECUTE" button is highlighted with a light grey border.

A questo punto, toccherà nuovamente all’utente direzionare l’esecuzione del programma ed effettuare le scelte riguardanti i branch. Verrà infine mostrato l’output della funzione.

The screenshot shows the same "Load Data From Database" window. The "TABLE NAME" field still contains "provaC". The "BRANCH" field now contains the text "Insert branch". The "OK!" button is still present. The large white rectangular area now displays "GENERATE DECISION TREE" followed by "=====
0:X=A
1:X=B

1
Predicted class Value : 10.0". At the bottom, the "NEW EXECUTION" button is highlighted with a light blue border.

Ora l'utente potrà decidere di uscire dal programma tramite il tasto "exit", effettuare una nuova esecuzione premendo "new execution" o tornare al menu principale mediante il tasto "menu".

Se invece viene scelta la seconda opzione, il caricamento da file, l'utente sarà chiamato ad inserire un nome (valido) di un file relativo ad una precedente esecuzione del programma. Le modalità di esecuzione, per quanto concerne le azioni dell'utente, saranno analoghe a quelle del primo caso.

P.S. Nel caso venga inserito un valore non valido nel campo "branch", il sistema segnalerà il fatto all'utente e ne richiederà l'inserimento fino a quando il valore non sarà valido.

Load From File

TABLE NAME: provaC

BRANCH: Insert branch OK!

Start prediction phase

0:X=A
1:X=B

0
0:Y<=2.0
1:Y>2.0

0
Predicted class Value : 1.0

EXECUTE NEW EXECUTION EXIT MENU

Autori

Lamorgese Gianfranco
Grimaldi Regina