

Progetto Metodi Avanzati di Programmazione sugli Alberi di Regressione

Introduzione ai Regression Tree

I Regression Tree vengono usati nell'ambito del Data Mining, il cui scopo è l'estrazione di conoscenza nascosta in databases voluminosi; in particolare, tra le diverse aree di applicazione, incontriamo i Regression Tree per l'appunto, quando si parla di regressione.

La regressione punta a predire il valore di un attributo numerico associato ad un esempio sulla base di valori osservati per altri attributi dello stesso esempio.

Vengono quindi usati alberi di regressione per effettuare la predizione di un valore (che può essere, ad esempio, la previsione di vendita di un determinato settore in un'azienda a seguito di determinate condizioni).

Un Regression Tree viene costruito attraverso un processo chiamato "recursive partitioning", un processo iterativo che divide i dati in partizioni o rami e continua a dividere ogni partizione in un gruppo più piccolo ogni volta che la funzione si sposta su un nuovo ramo.

Avremo quindi che ogni nodo interno rappresenta una variabile ed un arco verso un nodo figlio rappresenta un valore per quella proprietà.

Avremo anche i nodi "foglia", che rappresentano il valore predetto per la classe a partire dai valori delle altre proprietà che nell'albero sono rappresentati dal cammino dalla radice al nodo foglia.

Introduzione al Progetto

Il progetto è composto da 3 applicativi:

- Server
- Client da console
- Client con interfaccia grafica

Il server esegue le funzioni per la creazione del regression tree e la predizione dei valori acquisendo i dati da un database SQL ed invia il risultato al client.

Server

Quando il server viene lanciato rimane in ascolto sulla porta 8080 attendendo che un client si connetta.

A connessione avvenuta, il server lascerà decidere al client se acquisire i dati dal database o caricare il contenuto di un file relativo ad una precedente esecuzione.

Client

Come accennato prima esistono due versioni del client: una da console, che permette comunque un uso completo di tutte le funzionalità dell'applicazione ed una da interfaccia grafica; quest'ultima è stata realizzata usando Scene Builder nell'ecosistema di Java FX e risulta essere più user friendly rispetto al client da console.

L'interfaccia grafica è costituita da due parti:

- la prima scritta in Java;
- la seconda, invece, costituita da file FXML che descrivono il comparto grafico dell'interfaccia.

Il client richiede all'utente l'inserimento di indirizzo ip e port number del server per poi lasciar decidere sempre all'utente se caricare i dati da database o file. Verranno inoltre richiesti input per direzionare l'esecuzione della funzione; gli input dati dall'utente influenzeranno quindi il risultato della funzione.

Testing

Per testare l'applicazione sono stati utilizzati unit tests realizzati tramite il framework JUnit. La percentuale finale di coverage è 78.7%.

src	78,7 %	1.986	538	2.524
server	0,0 %	0	269	269
ServerOneClient.java	0,0 %	0	224	224
MultiServer.java	0,0 %	0	41	41
UnknownValueException.java	0,0 %	0	4	4
tree	83,8 %	1.082	209	1.291
RegressionTree.java	74,6 %	464	158	622
SplitNode.java	83,0 %	224	46	270
ContinuousNode.java	97,0 %	161	5	166
DiscreteNode.java	100,0 %	79	0	79
LeafNode.java	100,0 %	53	0	53
Node.java	100,0 %	101	0	101
database	92,3 %	456	38	494
DbAccess.java	76,2 %	48	15	63
TableData.java	93,6 %	191	13	204
TableSchema.java	96,5 %	110	4	114
DatabaseConnectionException.java	0,0 %	0	3	3
Example.java	96,1 %	74	3	77
Column.java	100,0 %	30	0	30
EmptySetException.java	100,0 %	3	0	3
data	95,3 %	448	22	470
Data.java	95,7 %	405	18	423
DiscreteAttribute.java	81,0 %	17	4	21
Attribute.java	100,0 %	18	0	18
ContinuousAttribute.java	100,0 %	5	0	5
TrainingDataException.java	100,0 %	3	0	3

Autori

- Grimaldi Regina
r.grimaldi8@studenti.uniba.it
- Lamorgese Gianfranco
g.lamorgese5@studenti.uniba.it

Riferimenti

https://en.wikipedia.org/wiki/Decision_tree_learning