KDB2000: Uno strumento per la scoperta di conoscenza nelle Basi di Dati.

Annalisa Appice, Michelangelo Ceci, Donato Malerba

Dipartimento di Informatica, Università degli Studi di Bari
Via Orabona, 4 – 70126 Bari
{Appice,Ceci,Malerba}@di.uniba.it

Sommario. Con la crescita esplosiva nelle capacità di generare e collezionare dati cresce l'importanza di automatizzare il processo di scoperta di conoscenza nelle basi di dati. I sistemi per la scoperta di conoscenza devono affrontare i problemi tipici dei dati disponibili nelle basi di dati del mondo reale. Infatti, questi sono spesso incompleti, dinamici, ridondanti, incerti. Questo articolo offre una panoramica del processo KDD (knowledge discovery in databases). In particolare, si descrive KDB2000, un sistema di Data Mining che assiste l'utente in tutte le fasi del processo KDD.

1 Il processo KDD

Il termine *knowledge discovery in databases* (*KDD)*, si riferisce all'intero processo, interattivo ed iterativo, di scoperta della conoscenza che consiste nell'identificazione di *relazioni tra dati* che siano *valide*, *nuove*, potenzialmente *utili* e *comprensibili* [3]. Il processo KDD può essere scomposto in più fasi tra le quali il *data mining* riveste un ruolo centrale [1].

Il primo passo è certamente quello di *individuare gli obiettivi dell'utente finale*. Esso ha molto in comune con le fasi iniziali di un qualunque progetto commerciale, durante le quali gli analisti del business e gli analisti dei dati collaborano per definire gli obiettivi da raggiungere. Il punto di partenza è la definizione dell'*aspettativa* in funzione della quale si potrà stabilire il successo o meno del progetto.

In relazione agli obiettivi dell'utente, è necessario identificare l'insieme di dati da analizzare *selezionando* un campione o un sottoinsieme delle variabili disponibili nella sorgente dei dati. Oltre le variabili selezionate, è necessario acquisire le corrispondenti informazioni semantiche (*metadati*), indispensabili per interpretare il significato di ciascuna variabile. I metadati potrebbero includere la definizione dei dati, la descrizione dei tipi, i valori potenziali, il loro sistema sorgente e formato.

I dati selezionati devono essere *pre-elaborati* al fine di ridurre l'effetto del rumore (*noise*), eliminare casi limite (*outlier*), valori obsoleti ed eventuali inconsistenze, decidere le strategie per la gestione dei valori nulli.

[1] È ormai invalso l'uso del termine Data Mining come sinonimo di KDD. In questo lavoro, tuttavia, si preferisce distinguere le due espressioni in accordo a quanto emerso durante la prima conferenza Internazionale su Knowledge Discovery in Database (KDD'95), Montreal, Agosto 1995.

Spesso potrebbe essere necessario *trasformare i dati* selezionati al fine di isolare caratteristiche utili a rappresentare i dati in base agli obiettivi dell'analisi. In particolare i dati sono trasformati in un formato (modello analitico) compatibile con gli algoritmi disponibili. Questo passo è fondamentale per garantire l'accuratezza dei risultati che dipende da come gli analisti decidono di strutturare i dati di input. Una complessa conversione è la *riduzione dei dati*, il cui obiettivo è ridurre il numero totale di variabili da analizzare combinandone alcune tra loro in modo da ottenerne una nuova. Altre conversioni sono lo *scaling* che consente di ridurre input numerici a specifici intervalli, la *discretizzazione* che converte variabili quantitative in categoriche, la *binarizzazione* che converte una variabile categorica in più variabili binarie ed il *campionamento* che restituisce un sottoinsieme casuale dell'insieme di partenza, di dimensione proporzionale ad una percentuale fissata.

La scelta del task di *Data Mining* (classificazione, regressione, clustering, regole di associazione,ecc..), è una fase fondamentale del processo KDD: essa influenza anche la scelta dell'*algoritmo* per la scoperta di nuove relazioni, in funzione del tipo di rappresentazione che si intende adottare per le relazioni estratte, e conduce a risultati utili solo se i passi precedenti del processo KDD sono stati correttamente eseguiti.

Infine nel processo KDD le relazioni scoperte devono essere *interpretate* alla luce della conoscenza pregressa, *valutate* rispetto all'obiettivo di business, e comunicate alle parti interessate mediante opportuna reportistica.

2 Gli strumenti a supporto del processo KDD

Lo studio avanzato nel campo del *KDD* ha reso possibile lo sviluppo di molti sistemi di data mining, che possono essere catalogati in quattro diverse "generazioni". I sistemi di prima generazione supportano un piccolo gruppo di algoritmi progettati per analizzare vettori di dati. I sistemi di seconda generazione, invece, si interfacciano con basi di dati per elaborare insiemi di dati complessi e di grandi dimensioni e supportano uno schema di data mining ed un linguaggio di interrogazione, garantendo maggiore flessibilità. I sistemi di terza generazione sono capaci di analizzare dati altamente eterogenei, distribuiti in rete locale e non. I sistemi di quarta generazione, infine, interagiscono direttamente con i generatori di dati.

Sono in commercio diversi sistemi di data mining, tra i quali citiamo DataMind [4], WizWhy (http://www.wizsoft.com/why.html) e Clementine [4]. *DataMind* è un sistema di terza generazione composto da due differenti prodotti che si basano sul modello client-server, rispettivamente *DataMind Professional Editing* e *DataMind DataCruncher*. Il primo, cioè il prodotto lato client, è usato singolarmente o da gruppi di persone per comprendere meglio i dati memorizzati su file locali. Il lato server è un motore per il data mining di dati complessi. I modelli creati sono visualizzabili tramite Excel o Word. *WizWhy* è un sistema di seconda generazione basato su un sofisticato algoritmo matematico per la scoperta di regole di associazione. L'utente deve solo selezionare la base di dati ed il campo variabile dipendente su cui vuole effettuare l'analisi. *Clementine* è un sistema di seconda generazione dotato di un'interfaccia visuale che assiste l'utente nelle fasi del processo KDD.

3 Il sistema KDB2000

KDB2000 (www.di.uniba.it/~malerba/software/KDB2000) è un sistema di seconda generazione che supporta in maniera intelligente l'utente nelle diverse fasi dell'intero processo KDD, allo scopo di estrarre e interpretare opportunamente relazioni tra i dati analizzati.

3.1 Architettura software

L'architettura software di KDB2000 è mostrata in Fig 1a. Il Data Banker è responsabile della connessione a una base di dati accessibile tramite fonte dati ODBC, della successiva interrogazione in SQL, e della restituzione dei risultati nel formato atteso. Questa componente consente anche al sistema richiedente di ottenere i metadati sui nomi di tabelle, sui tipi degli attributi e così via. Inoltre il Data Banker mantiene i dati dell'utente (per esempio, le interrogazioni effettuate) parte in un file e parte nel database stesso. La componente di *Data Visualization* (DV) consente di visualizzare i risultati di ciascuna fase KDD. Gli strumenti *ETL* (*Extraction, Transformation* and *Loading*) servono ad estrarre i dati dalla base di dati, a pulirli in modo da renderli consistenti con lo schema, a riepilogarli e a convertirli nel formato compatibile con quello previsto dai metodi di analisi disponibili. Infine, la componente *Data Mining tools* (DM) ospita gli algoritmi di data mining per l'analisi dei dati selezionati.

(a)

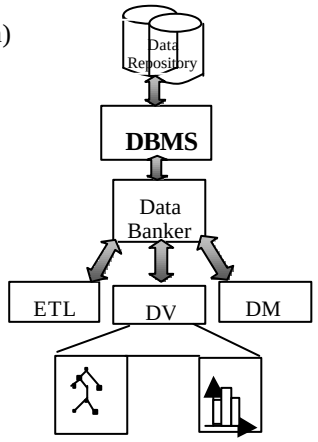

Fig. 1. Architettura software di KDB2000.

3.2 Funzionalità disponibili nel sistema

KDB2000 mette a disposizione dell'utente un insieme di funzionalità che lo assistono nelle varie fasi del processo KDD, dall'acquisizione dei dati alla visualizzazione dei risultati. Nella fase d'acquisizione dei dati, l'utente può selezionare o creare una nuova fonte dati per accedere e gestire le informazioni contenute nella base di dati relativa. L'obiettivo della selezione è identificare le sorgenti di dati disponibili ed estrarre i dati di interesse. I dati possono essere selezionati mediante un'interrogazione SQL definita dall'utente esplicitamente o attraverso l'ausilio di un wizard di creazione della query: tutte le interrogazioni possono essere memorizzate con i relativi risultati per utilizzi futuri in una workspace utente. La fase di pre-eleborazione dei dati utilizza strumenti sia statistici sia di visualizzazione grafica per aumentare la qualità degli stessi. In particolare per dati quantitativi possono essere calcolate le seguenti statistiche: il massimo, il minimo, la media, la moda, la mediana,

la deviazione standard e la correlazione. Le tecniche di visualizzazione grafica, invece, permettono una migliore comprensione del contenuto dei dati tramite istogrammi o diagrammi a torta. La fase di trasformazione converte i dati esistenti in un formato più adeguato alle esigenze dell'utente e compatibile con gli algoritmi di data mining. Le funzionalità implementate sono: *discretizzazione*, *binarizzazione*, *scaling*, *sostituzione dei valori nulli* e *campionamento*.

Le fasi appena descritte sono preparatorie all'applicazione di algoritmi di data mining. Quelli implementati in KDB2000 sono:

- *Induzione di alberi di decisione*: costruisce un albero di decisione che associa opportunamente una collezione di dati con un insieme predefinito di classi [7];
- *Classificazione con k-nearest neighbor*: classifica una osservazione in base alle *k* osservazioni più vicine secondo una misura di distanza fissata a priori [6];
- *Induzione di alberi di modelli*: costruisce una struttura ad albero alle cui foglie sono associati dei modelli di regressione lineare multipla [5];
- *Costruzione di cluster*: cerca di identificare un insieme finito di categorie o raggruppamenti (cluster) per descrivere i dati [2];
- *Generazione di regole di associazione*: ricerca una descrizione compatta di un sottoinsieme di dati. In particolare si generano regole del tipo X->Y dove X e Y sono congiunzioni di caratteristiche binarie [1].

La valutazione dell'accuratezza predittiva degli alberi di decisione e alberi di modelli avviene per mezzo della strategia *k-fold cross-validation* (k-CV). L'applicazione del k-CV prevede una suddivisione casuale dell'insieme di training *D* in k sotto insiemi disgiunti D_1, …, D_k ciascuno contenente approssimativamente lo stesso numero di osservazioni. Si costruisce il modello di classificazione/regressione per ciascun insieme D_i, i= 1…k, usando D-D_i come insieme di addestramento: questo modello è poi testato sull'insieme D_i. Lo stesso processo è ripetuto per i k sottoinsiemi. La stima dell'accuratezza dell'albero costruito su *D* è ottenuta come media delle varie stime ricavate separatemente per ogni D_i.

3.3 Ambiente di Sviluppo e piattaforma

KDB2000 è una applicazione a 32 bit per Windows 95/98/NT. I requisiti minimi per l'installazione di KDB2000 sono: spazio su disco 100 MB, memoria centrale 64MB. Il fabbisogno di memoria centrale è proporzionato alla mole dei dati da analizzare poiché alcuni metodi di data mining caricano in memoria centrale i dati da elaborare.

4 Un esempio di processo KDD con KDB2000.

KDB2000 è un valido strumento per l'estrazione di conoscenza, utilizzabile anche nell'ambito scientifico. Un esempio potrebbe essere uno studio scientifico condotto da un gruppo di biologi, allo scopo di trovare un predittore affidabile per l'età degli abalone (organismi marini dotati di guscio).

Il primo passo è selezionare l'insieme di caratteristiche da analizzare in relazione all'obiettivo proposto. A tal scopo è utile valutare caratteristiche fisiche come: sesso, lunghezza del corpo, diametro, altezza, numero di anelli, peso totale, peso

dell'apparato scheletrico, del guscio e degli organi interni. Ciascun attributo è pre-elaborato tramite tecniche statistiche come l'analisi dei minimi e massimi, lo studio di deviazione standard, e la ricerca di valori nulli, allo scopo di valutare la presenza di outlier o incertezza nei dati. La visualizzazione grafica tramite diagrammi a torta, o istogrammi aiuta a comprendere meglio la distribuzione dei valori. Tali pre-elaborazioni suggeriscono come trasformare i dati selezionati. Per esempio, si può decidere di rimuovere la presenza di valori nulli, sostituendoli con la media o la moda dell'attributo a seconda che esso sia continuo o discreto; oppure di discretizzare o ridurre (scaling) un attributo numerico, o di estrarre dai dati un campione casuale, analizzabile con algoritmi di Data Mining con migliori prestazioni in tempo. L'algoritmo di Data Mining, disponibile in KDB2000 per la scoperta di un predittore dell'età degli abalone, è *Smoti* (*S*tepwise *Mo*del *T*ree *I*nduction) [5] che costruisce un albero di modelli. Il modello costruito è reso disponibile in forma di albero (Fig. 2) o di insieme di regole.

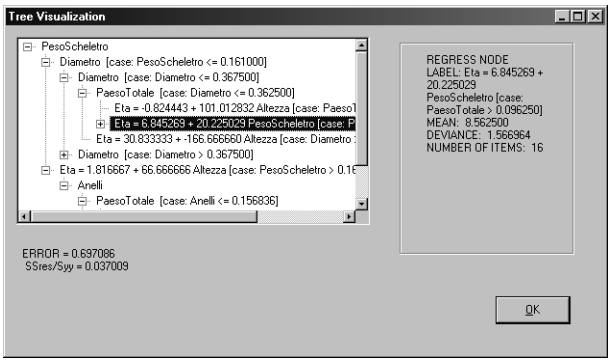

Fig. 2. Visualizzazione di un albero di modelli.

Bibliografia

1. Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proc. of the ACM SIGMOD Conference on Management of Data*, Washington, DC., pp. 207-216.
2. Farnstrom, F., Lewis, J., & Elkan, C. (2000). Scalability for clustering algorithms revisited. *SIKDD Explorations*, Vol. 2, n. 1, pp. 51-57.
3. Fayyad, U.M., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery: an overview. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (eds.), *Advances In Knowledge Discovery And Data Mining*, AAAI Press/The MIT Press, Menlo Park, CA, pp. 1-34.
4. Groth, R. (1998). *Data Mining. A Hands-On Approach for Business Professionals*, Prentice Hall PTR, Upper Saddle River, NJ.
5. Malerba, D., Appice, A., Bellino, A., Ceci, M., & Pallotta, D. (2001). Stepwise induction of model trees. In F. Esposito (ed.), *AI*IA 2001: Advances in Artificial Intelligence*, Lecture Notes in Artificial Intelligence, 2175, Springer, Berlin, Germany.
6. Mitchell, T.M. (1997). *Machine Learning*, McGraw Hill, New York, NY.
7. Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA.