L'informatique des entrepôts de données

Daniel Lemire

SEMAINE 13
L'exploration des données

13.1. Présentation de la semaine

L'exploration de données (ou data mining) est souvent associée à l'intelligence d'affaires. Nous présenterons cette semaine l'essentiel du domaine. Nous distinguerons aussi le data mining d'une approche guidée par l'utilisateur telle qu'OLAP.

13.2. Qu'est-ce que l'exploration de données?

Il n'y a pas de définition universellement acceptée de l'exploration de données. Cependant, le point de départ est toujours la présence d'une quantité appréciable de données. L'exploration de données repose ensuite sur des méthodes automatiques ou semi-automatiques pour le traitement des données. On cherche généralement à extraire automatiquement des observations générales à partir des données. Pour y arriver, on utilise un mélange de statistiques et d'intelligence artificielle.

A consulter : `http://fr.wikipedia.org/wiki/Exploration_de_donnees`

Il faut cependant distinguer l'exploration de données de la statistique traditionnelle. Bien qu'il n'y ait pas de distinction toujours claire entre les deux, le volume de données est une indication. Ainsi, si on procède à partir d'un échantillon (comme dans le cas d'un sondage), on fait alors des statistiques conventionnelles.

Dans le même ordre d'idée, toute forme d'intelligence artificielle n'est pas pertinente en exploration de données. En effet, l'intelligence

artificielle se construit souvent avec un modèle de la réalité élaboré à partir de règles définies par des experts. Cela n'est pas un cas d'apprentissage machine.

Une caractéristique fréquente en exploration de données est le très grand volume de données. Ainsi, les algorithmes utilisés en exploration de données doivent souvent être très efficaces. Il faut parfois utiliser des moyens considérables pour le traitement de données. Il n'est pas rare de faire appel à des grilles de calcul.

A consulter : `http://fr.wikipedia.org/wiki/Hadoop`

On utilise l'exploration de données à plusieurs fins: pour mieux comprendre les données ou même pour faire des prédictions. Par exemple, la plupart des logiciels de courriel utilisent une classification des courriels selon qu'ils sont légitimes ou qu'ils constituent du pourriel. Le plus souvent, cette classification s'effectue sur la base d'une exploration des données. Par exemple, on peut utiliser la classification bayesienne.

A consulter : `http://fr.wikipedia.org/wiki/Filtrage_bayesien_du_spam`

La différence primordiale entre OLAP et l'exploration des données est que les systèmes OLAP visent à répondre directement aux interrogations de l'analyste. En contraste, l'exploration de données n'est pas généralement entièrement guidée par un analyste. Ce sont plutôt les algorithmes qui déterminent le résultat.

Il y a une multitude de problèmes intéressants en exploration des données. Nous en présenterons brièvement trois exemples importants : le partitionnement, les règles d'association et les arbres de décision.

13.3. Partitionnement des données

Le partitionnement des données consiste à classifier automatiquement des données. Par exemple, un marchand pourrait vouloir classer automatiquement ses clients en ensembles homogènes.

A consulter : `http://fr.wikipedia.org/wiki/Partitionnement_de_donnees`

L'algorithme sans doute le plus utilisé est K-means. Il est plus facile de le comprendre si on imagine des points dans l'espace. Le calcul procède généralement de la manière suivante. On choisit aléatoirement k élements. Ils forment les centres de k noyaux. (Le paramètre k est fixé a priori.) On répartit les points restants dans l'un des k noyaux en choisissant selon la technique des plus proches voisins. Ensuite, pour chaque noyau, on calcule la moyenne des points. Chaque moyenne forme alors un nouveau centre à partir duquel nous allons construire un nouveau noyau. Et ainsi de suite. Au bout d'un certain temps, l'algorithme

se stabilisera. On trouve plusieurs animations intéressantes illustrant l'algorithme K-means.

A visiter : `http://en.wikipedia.org/wiki/K-means_clustering`

Bien que très intéressant, le partitionnement a ses limites. Par exemple, différents algorithmes vont donner différents partitionnements. Comment alors choisir la bonne version?

13.4. Les règle d'association

Une technique très commune en exploration de données est la recherche de règles d'association. Un exemple simple de règle d'association au sein d'un supermarché pourrait être la constatation que la plupart des gens qui achètent de la bière, achètent aussi des couches pour bébé. Sur la base de cette association, trouvée automatiquement, un marchand pourra adopter une disposition des produits visant à en bénéficier.

Formellement, on définit les règles d'association comme suit. Étant donné un ensemble d'événements $A_1, A_2, \ldots, A_i$ et un autre ensemble d'événements $B_1, B_2, \ldots, B_j$, on dit qu'il y a une règle d'association d'un ensemble vers l'autre, $A_1 \wedge \ldots \wedge A_i \rightarrow \wedge B_1 \wedge \ldots \wedge B_j$ si, lorsque les événements $A_1, A_2, \ldots, A_i$ se produisent, on constate aussi l'ensemble des événements $B_1, B_2, \ldots, B_j$. Par exemple, si le fait de prendre l'avion et d'être une femme est associé avec le fait d'être blonde et enceinte, on dira que nous avons la règle d'association (avion, femme) $\rightarrow$ (blonde, enceinte). Évidemment, toutes les règles d'association ne sont pas égales. On détermine la qualité d'une règle à l'aide de son support et de sa confiance. Le support est défini comme étant la probabilité que la règle s'applique (noté $P(B_1 \wedge \ldots \wedge B_j \wedge A_1 \wedge \ldots \wedge A_i)))$. En effet, s'il est rare que les femmes enceintes et blondes prennent l'avion, alors la règle portant sur les femmes qui prennent l'avion n'aura peut-être pas une grande valeur pour un analyste. La confiance d'une règle est définie comme étant la probabilité que le second ensemble d'événements se produira étant donné que le premier ensemble s'est produit : $P(B_1 \wedge \ldots \wedge B_j | A_1 \wedge \ldots \wedge A_i) = \frac{P(B_1 \wedge \ldots \wedge B_j \wedge A_1 \wedge \ldots \wedge A_i)}{P(A_1 \wedge \ldots \wedge A_i)}$. Afin d'illustrer ces mesures, considérons le tableau suivant :

Monster Species	Color	vegetarian?
Ziziz	Red	yes
YiYoz	Blue	yes
Filoufoul	Red	no
Coucou	Red	yes
Passpass	Blue	yes

À partir de cette table, nous pouvons déterminer que la règle selon laquelle les monstres bleus sont végétariens a un support de 40% et une confiance de 100%. La règle selon laquelle les monstres rouges sont végétariens a un support de 40% et une confiance de 66.7%.

Dans la pratique, on fixe souvent des contraintes sur le support et la confiance. Par exemple, on ne s'intéressera qu'aux règles qui ont une confiance d'au moins 80% et un support d'au moins 1%. Une règle d'association qui a à la fois une bonne confiance et un bon support est dite une règle forte.

En pratique, la recherche des règles fortes dans une grande base de données est un problème difficile. Un des algorithmes les plus communs est Apriori [**1**].

A consulter : `http://en.wikipedia.org/wiki/Apriori_algorithm`

En général, les algorithmes visant à trouver des règles d'association fortes fonctionnent de la manière suivante. On cherche d'abord à répondre à une contrainte sur le support. Parmi tous les cas possibles satisfant cette contrainte sur le support, on cherche des règles ayant une forte confiance.

Il n'est cependant pas facile de distinguer automatiquement les règles qui sont utiles de celles qui ne sont d'aucune autre utilité (pour un marchand, par exemple). Ainsi, malgré leur potentiel apparemment immense, les règles d'association demeurent relativement peu utilisées dans le commerce.

Il y a aussi une relation entre les règles d'association et les requêtes de types iceberg. Considérons l'exemple suivant où l'on donne la répartition des achats combinés de pain et de produits laitiers :

	white cheese	yellow cheese	skim milk	1% milk	2% milk	fat milk
whole wheat	12	0	43	13	22	0
white bread	8	16	432	3304	4343	444
brown bread	0	32	2	99	441	4324

On remarque immédiatement la présence d'articles fréquemment associés [**3**] :

- (white bread, 1% milk) : 3304
- (white bread, 2% milk) : 4343
- (brown bread, fatty milk): 4324

On peut mettre à jour de telles associations simples avec une requête iceberg [**2**]. Ils correspondent à des règles d'association (par ex. white bread $\rightarrow$ 1% milk).

13.5. Les arbres de décision

Les arbres de décision peuvent être vus comme une généralisation des règles d'association lorsqu'il y a plus d'une variable. Imaginons un marchand qui découvre la séquence logique suivante. Si un client entre seul, alors il va acheter pour moins de 50$. S'il entre avec quelqu'un d'autre, mais, qu'il a plus de 50 ans, il va acheter pour moins de 100$, mais sinon, il va probablement acheter pour plus de 200$. On voit ici que comme les règles s'appliquent les uns à la suite des autres, nous avons effectivement une forme d'arbre. Wikipedia a un excellent article sur ce sujet :

A consulter : `http://fr.wikipedia.org/wiki/Arbre_de_decision`

Les arbres de décision ont l'avantage d'être facilement compréhensible (par l'humain). Il existe plusieurs algorithmes efficaces pour en effectuer automatiquement le calcul.

13.6. Implantation

Il est parfois relativement facile de programmer ses programmes algorithmes d'exploration de données. Malheureusement, ce n'est pas toujours le cas, surtout si le volume de données est grand. Heureusement, il existe un vaste choix de solutions logicielles :

- Weka (Open Source) `http://www.cs.waikato.ac.nz/~ml/weka/`
- Mahout (Open Source) `http://mahout.apache.org/`
- Data Mining Extensions (Microsoft, SQL Server) `http://msdn.microsoft.com/en-us/library/ms132058.aspx`

13.7. Questions d'approfondissement

(a) Est-ce qu'OLAP est une forme d'exploration de données?
(b) Étant donné l'exemple de monstres, est-ce qu'on pourrait appliquer K-means pour partitionner l'ensemble des monstres?
(c) Étant donné une table comportant deux colonnnes, comment procéderiez-vous pour trouver toutes les règles d'association fortes allant de la première colonne à la seconde?
(d) Étant donné un arbre de décision sur 10 variables, quelle sera la hauteur maximale de l'arbre?

13.8. Réponses suggérées

(a) Non. Dans un système OLAP conventionnel, les requêtes proviennent d'un analyste. S'il y a extraction de règles ou de motifs, c'est plutôt dans l'esprit de l'analyste que le travail se fera.

(b) Certainement, mais il faudrait alors définir une mesure de distance entre les monstres.

(c) Chaque item respectent la contrainte sur le support. Il suffit ensuite fréquent forme une règle d'association. de calculer sa confiance.

(d) Dans un arbre de décision conventionnel, chaque variable n'intervient qu'une seule fois. L'arbre de décision aura donc une hauteur maximale de dix.
On pourrait tout d'abord chercher les items fréquents, c'est-à-dire les paires de valeurs (une par colonne) qui

BIBLIOGRAPHIE

1. R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. *ACM SIGMOD Record*, 22(2):207–216, 1993.
2. K. Beyer and R. Ramakrishnan. Bottom-up computation of sparse and iceberg cube. *SIGMOD Rec.*, 28(2):359–370, 1999.
3. J. Park, M. Chen, and P. Yu. An effective hash-based algorithm for mining association rules. *ACM SIGMOD Record*, 24(2):175–186, 1995.