Abstrakt

Werkzeuge zur interaktiven Clusteranalyse

Alexander Hinneburg
Martin-Luther-Universität Halle/Wittenberg
hinneburg@informatik.uni-halle.de

Einführung und Problemstellung

In vielen Data Mining Anwendungen wird Clusteranalyse als eine wichtige Technik zur Wissensentdeckung eingesetzt. Jedoch die Zielstellung und Anforderungen an eine solche Analyse sind sehr vielfältig und variieren in Abhängigkeit des Anwendungskontext. Demgegenüber stehen eine Vielzahl von automatischen Clusterungstechniken, die jedoch oft eine spezielle Cluster-Definition benutzen. Die Auswahl der Technik hängt somit jeweils auch von der Anwendung ab. Um den Anwender stärker in den Wissensentdeckungsprozeß mit einzubeziehen und dadurch die Ergebnis-Qualität und das Verständnis des Ergebnisses zu verbessern, erforschen wir in unserer Arbeitsgruppe Möglichkeiten automatische Verfahren mit interaktiven, visuellen Techniken zu kombinieren.

Viele Daten, die in KDD Anwendungen vorkommen, enthalten hochdimensionale Eigenschaftsvektoren, die numerische und kategorische Komponenten enthalten. Hier möchte ich mich auf Datenmengen mit Vektoren fester Länge mit numerischen Attributen beschränken ($D = \{x_1, \ldots, x_N\} \subset \mathbb{R}^d$). Bei fast allen realen Datenmengen kann im allgemeinen nicht ausgeschlossen werden, daß sie einen Anteil von Ausreißern (Rauschen) enthalten. Ausreißer sind Objekte, die sich nicht in einen Cluster einordnen lassen.

Viele Verfahren zeigen auf hochdimensionalen Daten ($d > 16$) ein quadratisches Laufzeitverhalten, was für große Datenmengen unakzeptabel ist, oder sie können keine sinnvolle Clusterstruktur in den Daten finden. Der Grund für die erste Beobachtung ist, daß viele Algorithmen [4, 10, 2, 14] nächste Nachbar- oder Bereichsanfragen nutzen und eine Laufzeit von $O(N \cdot T_{query})$ haben. Weber et. al. [13] zeigten, daß alle bekannten Index-Methoden diese Anfragen nicht in sublinearer Laufzeit für beliebig dimensionale Räume und unbekannte Daten Verteilung beantworten können. Zur zweiten Beobachtung, dem Fehlen einer sinnvollen Clusterstruktur in hochdimensionalen Daten, wurden in jüngster Zeit Arbeiten veröffentlicht, die untersuchen, ob der herkömmliche Ähnlichkeitsbegriff (z.B. euklidischer Abstand), auf dem alle Clusterdefinitionen aufbauen, bei zunehmender Dimensionalität sinnvoll bleibt [3, 6, 1]. Erste Resultate zeigten, daß dies nicht der Fall ist, weil die Datenpunkte in einem hochdimensionalen Raum sehr dünn verteilt sind. Beyer et. al zeigten, daß bei zunehmender Dimensionalität der Abstand der Datenpunkte zueinander schneller steigt als die Differenz der Distanzen zum nächsten und weitesten Nachbarn. Daraus folgt, daß ein auf Abstandsmessungen basierendes Ähnlichkeitsmaß in hochdimensionalen Räumen an Selektivität und Unterscheidungsvermögen einbüßt.

In [6] zeigten wir: hochdimensionale Feature-Vektoren beschreiben Objekte sehr genau und können somit auch Informationen über nicht relevante Eigenschaften enthalten. Wir modifizierten das Ähnlichkeitsmaß derart, daß nur die relevanten Attribute zum Messen des Abstands genutzt wurden. Mathematisch entspricht dieses Vorgehen, dem Messen des Abstands in einem projizierten Unterraum des hochdimensionalen Datenraumes. Übertragen auf Clusteralgorithmen muß ein Cluster nur einer bestimmten Projektion des Datenraumes, der Clusterdefinition genügen.

Die Frage ist nun, wie man eine solche Projektion bestimmen kann. Das Problem hat im allgemeinen einen exponentiellen Suchraum (bei der Beschränkung auf achsenparallele Projektionen). In unserer Arbeitsgruppe verfolgen wir den Ansatz eine große Anzahl von Projektionen automatisch von einen Optimierungsalgorithmus (genetischer oder Greedy Algorithmus) generieren zu lassen, diese automatisch auf Relevanz zu testen und zu filtern und dann manuell mit Hilfe von Visualisierungen vom Anwender bewerten zu lassen. Im folgenden Teil möchte ich auf den automatischen Relevanz-Test eingehen.

Ein Relevanz-Test für Projektionen

Die Projektionen die hier betrachtet werden sollen, haben die zusätzliche Eigenschaft, daß die Abstände der Punkte zueinander nach der Projektion nicht größer werden. Da bei einer Projektion Information weggelassen wird, muß eine Clusterstruktur nicht vollständig erkennbar sein. Die Projektion ist aber schon nützlich, wenn sich Teile der Custerstruktur erkennen lassen. Um eine Clusterstruktur zu finden, ist unser neues Paradigma, die Objektmenge in Untermengen zu aufzuteilen, wobei diese bezüglich eines Abstandsmaßes voneinander separiert sind. Basierend auf dem Begriff der Punktdichte (siehe [12, 11]) definieren wir einen Separators, der eine Punktmenge geometrisch (bezüglich einer Projektion) in zwei Untermengen teilt. Die Separations-Qualität ist die maximale Punktdichte auf der Grenze der beiden Mengen und die Teilungs-Qualität mißt wie balanciert der Schnitt ist. Um einen mehrdimensionalen Schnitt effizient und effektiv bestimmen zu können, haben wir eine neue Methode entwickelt, bei der mehrdimensionale Histogramme zur Ausreißerbehandlung [7], die kMeans Variante LBG-U [5] und eine verbesserte Form des konkurrierenden Hebbian-Lernens [9] verarbeitet wurden.

Falls ein Schnitt mit hoher Separations- und Teilungsqualität gefunden wurde und er vom Anwender als sinnvoll angesehen wird, kann man in den Untermengen rekursiv weiter nach geeigneten Schnitten suchen. So entsteht als Ergebnis eine einem Entscheidungsbaum ähnliche Struktur, die sich aber nicht an einer Trainingsmenge sondern an der Datenverteilung und der Aufgabe des Anwenders orientiert. Erste Ergebnisse wurden in [8] vorgestellt.

Literatur

[1] Charu Aggarwal and Phillip Yu. The igrid index: Reversing the dimensionality curse for similarity indexing in high dimensional space. In *SIGKDD 2000, Proceedings 4th Int. Conf. on Knowledge Discovery and Data Mining*, pages 119–129, 2000.

[2] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. In *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadephia, Pennsylvania, USA*, pages 49–60. ACM Press, 1999.

[3] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is nearest neighbors meaningful? In *Proc. of the Int. Conf. Database Theorie*, pages 217–235, 1999.

[4] M. Ester, H.P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD 1996, Proceedings 3rd Int. Conf. on Knowledge Discovery and Data Mining*, pages 226–231. AAAI Press, 1996.

[5] B. Fritzke. The LBG-U method for vector quantization – an improvement over LBG inspired from neural networks. *Neural Processing Letters*, 5(1):35–45, 1997.

[6] A. Hinneburg, C. Aggarwal, and D.A. Keim. What is the nearest neighbor in highdimensional spaces. In *Proc. 26th Int. Conf. on Very Large Data Bases*, 2000.

[7] A. Hinneburg and D. A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *KDD 1998, Proceedings 4th Int. Conf. on Knowledge Discovery and Data Mining*, pages 58–65. AAAI Press, 1998.

[8] A. Hinneburg, D.A. Keim, and M. Wawryniuk. Hd-eye: Visual mining high-dimensional data. *IEEE Computer Graphics and Applications*, 19(5):22–31, 1999.

[9] T.-M. Martinetz. Competitive hebbian learing rule forms perfectly topology preserving maps. In *Proceeding of the Int. Conf. Artificial Neural Networks*, pages 427–434. Springer, Amsterdam, 1993.

[10] J. Sander, M. Ester, H.-P.Kriegel, and X. Xu. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery, an International Journal, Kluwer Academic Publishers*, 2(2):169–194, 1998.

[11] D.W. Scott. *Multivariate Density Estimation.* Wiley and Sons, 1992.

[12] B.W. Silverman. *Density Estimation.* Chapman & Hall, 1986.

[13] R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proc. of 24rd Int. Conf. on Very Large Data Bases (VLDB'98)*, pages 194–205, 1998.

[14] Xiaowei Xu, Martin Ester, Hans-Peter Kriegel, and Jörg Sander. A distribution-based clustering algorithm for mining in large spatial databases. In *Proceedings of the Fourteenth International Conference on Data Engineering, February 23-27, 1998, Orlando, Florida, USA*, pages 324–331. IEEE Computer Society, 1998.