Concept Tree Based Clustering Visualization with Shaded Similarity Matrices

Jun Wang Bei Yu Les Gasser
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign
501 E. Daniel St., Champaign, IL 61820, USA
{junwang4, beiyu, gasser}@uiuc.edu

Abstract

One of the problems with existing clustering methods is that the interpretation of clusters may be difficult. Two different approaches have been used to solve this problem: conceptual clustering in machine learning and clustering visualization in statistics and graphics. The purpose of this paper is to investigate the benefits of combining clustering visualization and conceptual clustering to obtain better cluster interpretations. In our research we have combined concept trees for conceptual clustering with shaded similarity matrices for visualization. Experimentation shows that the two interpretation approaches can complement each other to help us understand data better.

Keywords: Clustering Visualization, Conceptual Clustering, Shaded Similarity Matrix, Concept Tree

1. Introduction

One of the problems with existing clustering methods is that the interpretation of clusters produced may be difficult. To address this interpretation problem, on the one hand, people from statistics and graphics have focused on visualization approaches[4, 5]. On the other hand, researchers in machine learning (or artificial intelligence) have developed conceptual clustering[1, 10]. Clustering visualization can help users visually perceive the clusterings, and sometimes even hidden patterns in data. Conceptual clustering aims at representing the clusterings using symbolic knowledge. Clustering visualization utilizes people's perceptual ability (low-level information processing), while conceptual clustering exploits human inference ability (high-level information processing). However, these two different approaches have not previously been combined. The purpose of this paper is to combine conceptual clustering with visualization in order to obtain better intrepretations of clusterings.

Our approach is to use shaded similarity matrices[3] for visualization and concept trees[7] for conceptual clustering. Since there are exponentially many ways to order a set of objects, the key problem for the shaded similarity matrix approach is how to order the data or objects in a matrix so that similar objects are adjacent. Heuristic strategies are needed for generating a near-optimal ordering. Happily, concept trees provide not only an approach to conceptual clustering, but also a potential approach to solve the ordering problem, because *the more specific the concept shared by two objects, the more similar the two objects*. Our experiments (presented later) do show that concept trees are effective.

2. Shaded Similarity Matrices

Over the past forty years, shaded similarity matrices have been used in visual cluster anaylsis[8, 3, 11]. In a shaded similarity matrix[1], similarity in each cell is represented using a shade to indicate the similarity value: greater similarity is represented by dark shading, and lesser similarity by light shading. The dark and light cells may initially be scattered over the matrix. To reveal the potential clusterings visually, the rows and columns need to be re-organized so that similar objects are put in adjacent positions. If "real" clusters exist in the data, they should appear as symmetrical dark squares along the diagonal.

Here we will briefly show how shaded similarity matrices are constructed and how one looks through an example. The data used in the example is part of the *Iris* data from the UCI repository[9]. The *Iris* data set contains 150 instances, evenly distributed in 3 classes. We fetch 5 instances from each class, and thus obtain 15 instances (Table 1). The similarity matrix was computed based on Euclidean distance (Table 2).

The shaded similarity matrix is illustrated in Fig. 1. The right figure in Fig. 1 is generated from the original similarity matrix using the seriation algorithm which was pro-

[1] Some researchers use the term *shaded distance matrix, shaded proximity matrix*, or *trellis diagram*.

Table 1. Data matrix extracted from the Iris data set. *Abbreviations:* **sl: sepal-length, sw: sepal-width, pl: petal-length, pw: petal-width.**

Instance	sl	sw	pl	pw
e_1	5.1	3.5	1.4	0.2
e_2	6.3	2.9	5.6	1.8
e_3	7.0	3.2	4.7	1.4
...	...	...	...	...
e_{15}	6.5	2.8	4.6	1.5

Table 2. Similarity matrix corresponding to Table 1.

1.0	0.1	0.2	0.8	0.2	0.1	...	0.2
0.1	1.0	0.3	0.2	0.3	0.8	...	0.3
0.2	0.3	1.0	0.2	0.9	0.3	...	0.8
...	...						
0.2	0.3	0.8	0.2	0.8	0.3	...	1.0

posed in ClustanGraphics [11]. It works by weighting each similarity using the distance of the similarity cell from the diagonal. The algorithm tries to minimize the sum of the weighted similarities in the similarity matrix by reordering the pre-computed clusters in an agglomerative hierarchical clustering such as a dendrogram.

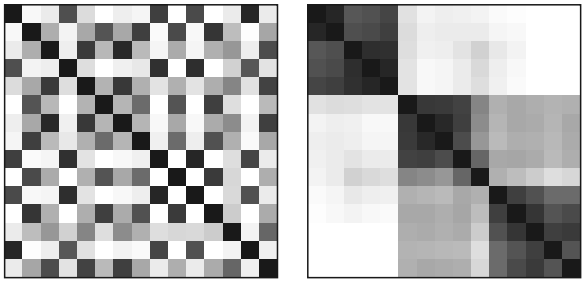

Figure 1. LEFT: *Randomly ordered* **shaded similarity matrix;** RIGHT: *Reordered* **shaded similarity matrix using a seriation algorithm.**

To display a similarity matrix of n objects, we need n^2 cells or $\frac{n^2}{2}$ cells (in the case of half matrix). In practice, usually it is not necessary to display all cells in a matrix. In this paper, only those cells are displayed whose similarity values are over a pre-specified threshold.

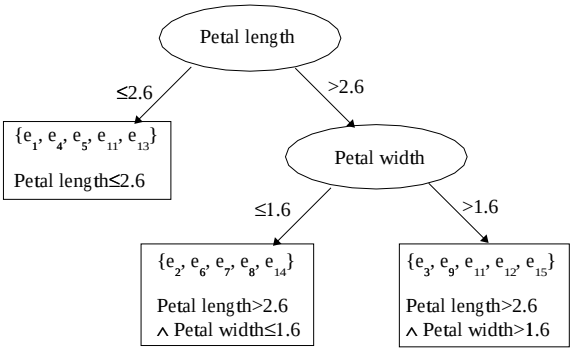

Figure 2. A concept tree on the Iris data set.

3. Concept Tree Based Clustering

3.1. Concept tree

A *concept tree* (also known as *concept hierarchy*) is composed of nodes and links with each node representing a concept[7]. The links connecting a node to its children specify an 'IS-A' or 'subset' relation. Fig.2 is an illustration of a concept tree generated by using the algorithm in Table 3 on the 15 *Iris* instances mentioned above.

3.2. Concept tree construction algorithm

Table 3 is a general algorithm for constructing a concept tree. The key part of the algorithm is how to select a *best* attribute using some split measurement. In this paper, we take the *within-group average similarity*[2] as the measurement. The best attribute is the one with the maximum within-group average similarity. Suppose an attribute a with k possible values splits a data set S into k subsets: $\{S_1, \cdots, S_k\}$. Let $\sigma(e_1, e_2)$ be the similarity of two instances e_1 and e_2. Then the *within-group average similarity* is defined as:

$$WAS(a) = \sum_{i=1}^{k} \frac{|S_i|}{|S|} \frac{\sum_{e_1,e_2 \in S_i} \sigma(e_1, e_2)}{|S_i|^2} \qquad (1)$$

3.3. Concept tree based ordering for shaded similarity matrices

Given that *objects sharing the same concept are probably similar*, if we put together all objects belonging to a concept, the ordering problem for shaded similarity matrices will be heuristically solved. Concretely speaking, given a tree with k leaves from left to right: $< L_1, L_2, \cdots, L_k >$. Each leaf represents a concept which covers a set of objects:

Table 3. Concept tree construction algorithm.

Inputs: The current node N of the concept tree,
an instance set S,
and an attribute set A .
Output: A concept tree.
Procedure: $CTree(N, S, A)$
If both S and A are not empty,
Then:
Select a best attribute $a \in A$ using some metric,
For each possible value v_i of a:
Form a node C, corresponding to the test $a = v_i$, *
Let S_{v_i} be the subset of S that have value v_i for a,
Make node C a child of Node N,
$CTree(C, S_{v_i}, A - \{a\})$.

* For numberical attributes, the test is represented as $a \leq v_i$ or $a > v_i$.

$L_i = \{e_{i_1}, \cdots, e_{i_n}\}$. Then the ordering of all objects will be:

$$< \{e_{1_1}, \cdots, e_{1_n}\}, \cdots\cdots, \{e_{k_1}, \cdots, e_{k_n}\} >$$

We don't care about the internal object ordering within a leaf. We assume that the objects within a leaf are similar enough, and if not the leaf node can be partitioned into smaller leaves until the objects within a leaf do become similar enough. Note that this ordering can only produce a partial order of the objects due to limitations of the tree structure.

4. Experimentation

To demonstrate the effectiveness of our approach, we have tested it on some UCI datasets. Here we use the full *Iris* dataset (150 instances) to generate a concept tree. Table 4 shows 3 concepts which are visualized as three square blocks along the diagonal from the left-top to right-bottom in the Fig. 3. Note that the concept tree here is a little bit different from the one in Fig. 2 because of the different dataset size.

Table 4. Concepts shown in Fig. 3.

Concept name	Square block	Concept description
concept-1	left-top	$pl \leq 2.5$
concept-2	center	$pl > 2.5 \wedge pw \leq 1.8$
concept-3	right-bottom	$pl > 2.5 \wedge pw > 1.8$

Among these 3 concepts, the concept-1 is the most clearly separated from other concepts. The concept-2 and concept-3 are not perfectly separated. There are some instances covered by concept-2 which have high similarity[2] with the instances in concept-3 and vice versa.

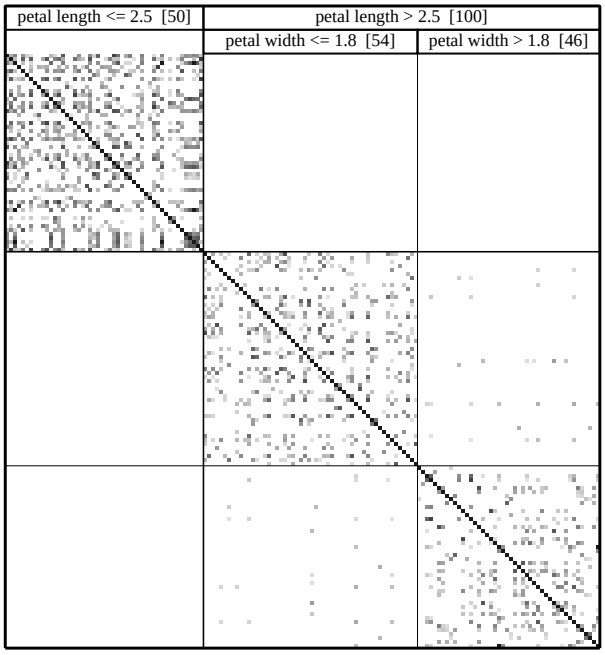

Figure 3. Concept tree based clustering visualization on the Iris dataset.

Overall, the visualization result in the Fig. 3, allows us to discover two properties of the *Iris* dataset: 1) the dataset is naturally divided into three groups or clusters (there are three self-similar blocks); 2) each group can be described by a simple concept. In other words, we can say that both the visualization and the acquired concepts complement each other to help us better understand the data.

5. Discussion

First, it is apparent that the effectiveness of our approach depends on the definition of similarity, which is also a general problem for clustering methods. Second, our approach has an assumption that the data can be described conceptually, which does not always hold. However, if the data cannot be described conceptually, it is hard to say that the data are fully understood. Therefore, the problem becomes how to choose the right concept representation. Concept trees are only one kind of representation, and are not appropriate for all data.

Third, visualization based on shaded similarity matrices

[2] In the Fig. 3, if the similarity between two instances is over a threshold, a spot will be displayed for these two instances. In this situation we say that these two instances have high similarity.

has a scalability limitation. One solution is to use sampling and ensemble approaches. Using small sample sizes such as 100 or 200, we have tested the sampling approach on some Statlog datasets, including the *Shuttle* dataset which contains $43,500$ instances[6]. The results are promising.

6. Summary

This paper proposes a new approach for getting better interpretations for clustering results by combining visualization and conceptual clustering. This is achieved by using concept trees as a heuristic ordering strategy for shaded similarity matrices. Our experiment shows that the two clustering interpretation approaches can complement each other to help us better understand the data.

Acknowledgments

This work was partially supported by the Information Systems Research Lab (ISRL) of the Graduate School of Library and Information Science at University of Illinois at Urbana-Champaign. We are thankful to Dr. David Dubin, Professor Jiawei Han, and reviewers for valuable comments.

References

[1] F. Douglas. Knowledge acquisition via incemental conceptual clustering. *Machine Learning*, 2:139–172, 1987.

[2] B. S. Everitt. *Cluster Analysis*. John Wiley & Sons, Inc., New York, 1974.

[3] N. Gale, W. Halperin, and C. Costanzo. Unclassed matrix shading and optimal ordering in hierarchical cluster analysis. *Journal of Classification*, 1:75–92, 1984.

[4] P. E. Hoffman and G. G. Grinstein. A survey of visualizations for high-dimensional data mining. In U. Fayyad, G. Grinstein, and A. Wierse, editors, *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann, 2001.

[5] D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics (Special Section on Information Visualization and Visual Data Mining)*, 8(1):1–8, 2002.

[6] R. King, C. Feng, and A. Shutherland. Statlog: comparison of classification algorithms on large real-world problems. *Applied Artificial Intelligence*, 9(3):259–287, 1995.

[7] P. Langley. *Elements of Machine Learning*. Morgan Kaufmann Publishers, 1996.

[8] R. L. Ling. A computer generated aid for cluster analysis. *Communications of the ACM*, 16(6):355–361, 1973.

[9] C. J. Merz, P. M. Murphy, and D. W. Aha. UCI repository of machine learning databases, http://www.ics.uci.edu/˜mlearn/mlrepository.html, 1997. Dept. of Information and Computer Science, University of California at Irvine.

[10] L. Talavera and J. Bejar. Generality-based conceptual clustering with probabilistic concepts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):196–206, 2001.

[11] D. Wishart. ClustanGraphics3: Interactive graphics for cluster analysis. In W. Gaul and H. Locarek-Junge, editors, *Classification in the Information Age*, pages 268–275. Springer-Verlag, 1999.