Concept Tree Based Clustering Visualization with Shaded Similarity Matrices

Bei Yu
University of Illinois at Urbana-Champaign

Jun Wang
University of Illinois at Urbana-Champaign

Les Gasser
University of Illinois at Urbana-Champaign

Follow this and additional works at: https://surface.syr.edu/istpub

Recommended Citation
Concept Tree Based Clustering Visualization with Shaded Similarity Matrices

Description/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

Disciplines
Library and Information Science
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 interpretations 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 analysis[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.

<table>
<thead>
<tr>
<th>Instance</th>
<th>sl</th>
<th>sw</th>
<th>pl</th>
<th>pw</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>5.1</td>
<td>3.5</td>
<td>1.4</td>
<td>0.2</td>
</tr>
<tr>
<td>e2</td>
<td>6.3</td>
<td>2.9</td>
<td>5.6</td>
<td>1.8</td>
</tr>
<tr>
<td>e3</td>
<td>7.0</td>
<td>3.2</td>
<td>4.7</td>
<td>1.4</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>e15</td>
<td>6.5</td>
<td>2.8</td>
<td>4.6</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Table 2. Similarity matrix corresponding to Table 1.

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>0.1</th>
<th>0.2</th>
<th>0.8</th>
<th>0.2</th>
<th>0.1</th>
<th>...</th>
<th>0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>0.3</td>
<td>0.2</td>
<td>0.3</td>
<td>0.8</td>
<td>0.3</td>
<td>...</td>
<td>0.3</td>
</tr>
<tr>
<td>0.1</td>
<td>0.3</td>
<td>1.0</td>
<td>0.2</td>
<td>0.9</td>
<td>0.3</td>
<td>0.8</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0.2</td>
<td>...</td>
<td>...</td>
<td>0.2</td>
<td>0.8</td>
<td>0.3</td>
<td>...</td>
<td>1.0</td>
<td></td>
</tr>
</tbody>
</table>

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 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: \( \{S_1, \ldots, S_k\} \). 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:

\[
WAS(a) = \sum_{i=1}^{k} \left| \frac{|S_i|}{|S|} \sum_{e_1, e_2 \in S_i} \sigma(e_1, e_2) \right| |S_i|^2
\]  

(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: \( < L_1, L_2, \ldots, L_k > \). 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:

1. Select a best attribute \( a \in A \) using some metric.
2. 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 numerical attributes, the test is represented as \( a \leq v_i \) or \( a > v_i \).

\[ L_i = \{e_{i1}, \ldots, e_{in}\} \]

Then the ordering of all objects will be:

\[ < \{e_{11}, \ldots, e_{1n}\}; \ldots; \{e_{k1}, \ldots, e_{kn}\} > \]

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>
<thead>
<tr>
<th>Concept name</th>
<th>Square block</th>
<th>Concept description</th>
</tr>
</thead>
<tbody>
<tr>
<td>concept-1</td>
<td>left-top</td>
<td>( pl \leq 2.5 )</td>
</tr>
<tr>
<td>concept-2</td>
<td>center</td>
<td>( pl &gt; 2.5 \wedge pw \leq 1.8 )</td>
</tr>
<tr>
<td>concept-3</td>
<td>right-bottom</td>
<td>( pl &gt; 2.5 \wedge pw &gt; 1.8 )</td>
</tr>
</tbody>
</table>

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 with the instances in concept-3 and vice versa.

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 2In 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