Pattern Recognition 45 (2012) 2633–2644

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Clustering with proximity knowledge and relational knowledge

Daniel Graves [a,*], Joost Noppen [b], Witold Pedrycz [a]

[a] Department of Electrical and Computer Engineering, 9107-116th Street, University of Alberta, Edmonton, Alberta, Canada T6G 2V4
[b] Computing Department, InfoLab21, South Drive, Lancaster University, Lancaster LA1 4WA, UK

ARTICLE INFO

Article history:
Received 23 March 2011
Received in revised form
18 November 2011
Accepted 20 December 2011
Available online 4 January 2012

Keywords:
Relational clustering
Fuzzy clustering
Proximity
Knowledge representation
Software requirements

ABSTRACT

In this article, a proximity fuzzy framework for clustering relational data is presented, where the relationships between the entities of the data are given in terms of proximity values. We offer a comprehensive and in-depth comparison of our clustering framework with proximity relational knowledge to clustering with distance relational knowledge, such as the well known relational Fuzzy C-Means (FCM). We conclude that proximity can provide a richer description of the relationships among the data and this offers a significant advantage when realizing clustering. We further motivate clustering relational proximity data and provide both synthetic and real-world experiments to demonstrate both the usefulness and advantage offered by clustering proximity data. Finally, a case study of relational clustering is introduced where we apply proximity fuzzy clustering to the problem of clustering a set of trees derived from software requirements engineering. The relationships between trees are based on the degree of closeness in both the location of the nodes in the trees and the semantics associated with the type of connections between the nodes.

1. Introduction

Relational clustering problems are increasingly encountered in a number of different applications, cf. [1–13]. It is common to identify clustering problems where data are vectors positioned in a d-dimensional feature space $X=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N\}\subset\Re^d$. An area that is gaining interest is relational clustering, where the entities (i.e., objects) to be clustered are not vectors themselves but rather are described in terms of relational knowledge between objects. Following similar notation as presented in [2], we denote these objects by $X=\{O_1,O_2,\ \ldots,\ O_N\}$, which may or may not have a numerical representation in a feature space. Hathaway, et al., introduced relational knowledge coming in the form of a fuzzy binary relation called $\rho(\mathbf{x}_i,\mathbf{x}_j)$, i.e. $\rho: X\times X\mapsto[0,1]$. There are many interesting applications of relational clustering as well, cf. [2,4–6,8,14–20]. Clustering with proximity information, however, offers a number of benefits over clustering with other forms of relational knowledge including the more common distance-based knowledge. While there are many forms of relational knowledge to choose from, it is important to understand the advantages and challenges of each form of domain knowledge to select the best one for a given problem. This work is largely motivated to that end to gain a better understanding of the advantages and disadvantages of proximity and distance knowledge in the context of clustering.

Let us consider an illustrative example to highlight the essence of the study. Suppose we have a data set with three clusters as shown in Fig. 1.

In order to cluster this data set with commonly encountered algorithms such as Fuzzy C-Means (FCM) or k-means, we would need to choose a suitable distance measure (i.e. a way to measure the relationship between the objects (points in this example)). A simple measure that is most often used is the Euclidean distance. Given the rather complicated geometry of this data set, it is not surprising that FCM equipped with the Euclidean distance is unable to discover a structure of data, see Fig. 2.

The boundaries of the clusters are shown in Fig. 2 (note that those are determined by finding all points where the membership values of the two neighboring clusters are equal). Let us express the relationship between these objects using proximity rather than distance, where proximity describes the degree of closeness of the objects. In the example, these objects are points on a two dimensional grid; therefore, we can quantify the degree of closeness using a Gaussian membership function (see Table 1 for the formula). The width parameter σ^2 offers some flexibility of the construct. When using a proximity-based relational clustering algorithm to cluster the data, its "real" structure has been revealed as can be seen in Fig. 3.

When $\sigma^2=0.7$, we obtain two spherical clusters and a third cluster that is formed by the rest of the data. The flexibility offered by the width parameter is visualized in Fig. 3(c)–(e): when the values of σ^2 increase, the two spherical clusters become larger and "over-emphasized." The other values of sigma show the flexilibity offered by adjusting sigma. In Fig. 3(c)–(e), the two spherical clusters become larger and over-emphasized as sigma increases. As a result, they

* Corresponding author. Tel.: +1 780 461 8702; fax: +1 780 492 1811.
E-mail addresses: dgraves@ualberta.ca, dangraves77@gmail.com (D. Graves), j.noppen@uea.ac.uk (J. Noppen), wpedrycz@ualberta.ca (W. Pedrycz).

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.12.019

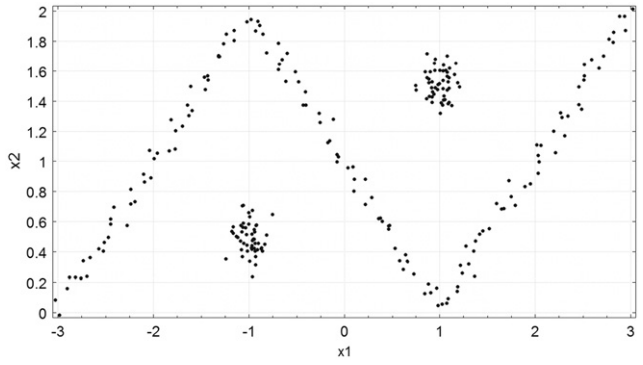

Fig. 1. An illustrative example.

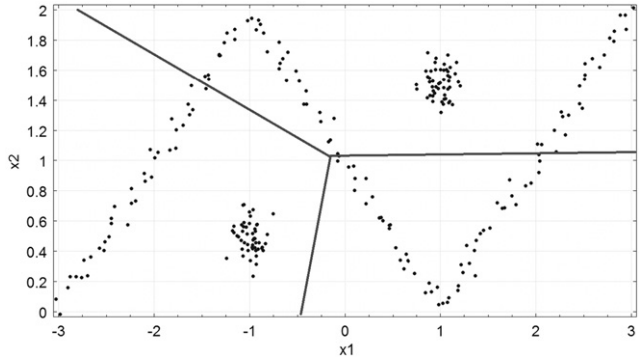

Fig. 2. Boundaries of FCM-constructed clusters.

Table 1
Common proximity functions.

Name	*Proximity function*	*Parameters*
Gaussian	$p(\mathbf{x},\mathbf{y}) = \exp(-\|\mathbf{x}-\mathbf{y}\|^2/\sigma^2)$	$\sigma^2 > 0$
Cosine	$p(\mathbf{x},\mathbf{y}) = \frac{1}{2}\left[\frac{\mathbf{x}^T\mathbf{y}}{\sqrt{(\mathbf{x}^T\mathbf{x})(\mathbf{y}^T\mathbf{y})}}\right] - \frac{1}{2}$	
Polynomial	$p(\mathbf{x},\mathbf{y}) = \frac{1}{2}\left[\frac{(\mathbf{x}^T\mathbf{y}+\theta)^p}{\sqrt{(\mathbf{x}^T\mathbf{x}+\theta)^p(\mathbf{y}^T\mathbf{y}+\theta)^p}}\right] - \frac{1}{2}$	$\theta \geq 0, p \in$ naturals;
ANOVA	$p(\mathbf{x},\mathbf{y}) = \frac{1}{n}\sum_{k=1}^{n} \exp\left(-\frac{1}{\sigma^2}\|\mathbf{x}^k-\mathbf{y}^k\|^{2p}\right)$	$\sigma \in \Re, p > 0, n \in$ naturals;

include part of the zig-zag pattern. Therefore, choosing proximity knowledge instead of the Euclidean distance (which is quite inflexible) has definite advantages including:

- semantics arising from graded similarity, e.g. "close" and "far," and
- flexibility.

This flexibility arises from the relationship of proximity functions to kernel functions since a wide range of kernel functions [27] can be employed to quantify proximity. This relationship is well known, cf. [28], where the authors show that clustering based on the adjacency matrix is equivalent to kernel-based clustering. The adjacency matrix is found in spectral clustering literature [9,29] where it is used to find a graph cut. It should be noted that the definition of adjacency is very similar to but not identical to our definition of proximity.

The problem of selecting a form of relational knowledge becomes even more complicated as the available information may originate from a number of different sources [2]:

- obtained from a human expert, and
- computed based on measures of distance, similarity, or proximity.

Since we have to adhere to the properties of distance, similarity or proximity, requesting accurate relational knowledge from a human expert can become a problem. This is because the properties of, for example, distance must obey the triangular inequality, which cannot always be easily guaranteed. Recently, proximity functions have become popular in deriving relational information for use in clustering [4–6]. Roughly speaking, proximity captures the degree of resemblance between objects and can be described in a form of a binary fuzzy relation quantifying the notion of "closeness" or resemblance (as similarity described in [2]). Distance, on the other hand, is a measure of the separation of objects. The formal definition of proximity relational information is that it must satisfy the properties of identity ($p(\mathbf{x},\mathbf{x})=1$) and symmetry $p(\mathbf{x},\mathbf{y})=p(\mathbf{y},\mathbf{x})$. Proximity does not require transitivity. In contrast, distance must satisfy the triangular inequality; therefore, proximity can be more easily obtained from a human expert. In addition, the properties of proximity are interesting from a research perspective as they offer a number of advantages over the more common forms of relational knowledge due to the semantics of "closeness." However, there are a number of open problems including how to select the best form of domain knowledge for a given problem and how to determine the most suitable relational measure of that knowledge, e.g. proximity function, distance function, or similarity function. In particular limited work has been done on quantifying the benefits of the different forms of relational knowledge and their impact on relational clustering frameworks.

To illustrate some of the current issues arising from relational knowledge, consider the problems of clustering a collection of trees such as those discussed in [21,22]. An example of a set of trees is depicted in Fig. 4 where the nodes are labeled by lower case letters. A serious concern is the lack of a numerical representation of this problem in a feature space. These trees are entities, which are much easier to capture by relational knowledge for the purposes of clustering.

One way to describe these trees relationally is to use the edit distance, cf. [23–26]. Note that there are polynomial time algorithms that have been devised [6] to determine this. Using this relational knowledge, one can cluster the trees shown in Fig. 4 using relational FCM or a similar relational clustering algorithm. We may also choose to represent the relational information using proximity, which is produced by a proximity function. One of the benefits of proximity knowledge is that there is semantics associated with the relational information, i.e. "close" (proximity values approaching 1) and "far" (proximity values approaching 0). With many different forms of relational knowledge, a question arises on how does one choose the most suitable form of knowledge for a given problem. One of the goals of this study is to characterize the merits of using either proximity or distance to aid in this decision making.

To summarize, the main objectives of our study are to

- introduce and discuss the advantages and disadvantages of distance and proximity relational knowledge, and
- compare two clustering frameworks that make use of distance and proximity relational knowledge to better understand their differences and benefits.

The novelty introduced in this study stems from:

1. an in-depth presentation of the new proximity fuzzy clustering framework only briefly introduced in [7],