Available online at www.sciencedirect.com

ScienceDirect

Signal Processing 87 (2007) 2260–2267

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Fast communication

Unsupervised data reduction

Oleg Okun[a,*], Helen Priisalu[b]

[a]University of Oulu, Finland
[b]Teradata, Espoo, Finland

Received 20 October 2006; received in revised form 15 January 2007; accepted 8 February 2007
Available online 2 March 2007

Abstract

We propose a data reduction method based on fuzzy clustering and nonnegative matrix factorisation. In contrast to different variants of data set editing typically used for data reduction, our method is completely unsupervised, i.e., it does not need class labels to eliminate examples from a data set. Thus, it is useful in exploratory data analysis when class labels of examples are unknown or unavailable in order to gain insight into structure of different groups of patterns. Also unlike many types of unsupervised clustering relating a single example (cluster centroid) to each cluster, our method associates a set of the most representative examples with each cluster. Hence, it makes cluster structure more transparent to a data analyst.

Keywords: Exploratory data analysis; Data reduction; Nonnegative matrix factorisation

1. Introduction

According to [1] (p. 146), exploratory data analysis (EDA) is "an approach to data analysis that emphasizes the use of informal graphical procedures not based on prior assumptions about the structure of the data or on formal models for the data." The data structure can be modelled as Data = Smooth + Rough [1], where the "Smooth" is what characterises the essence of the data and the "Rough" is minute details that are often unimportant. The objective of EDA is to separate the "Smooth" from the "Rough". Thus, EDA is a mostly visual approach for data analysis intended to maximise insight into a data set and to uncover underlying structure of the data. A typical scenario where EDA is useful is when an analyst faces a data set without label information attached to examples. The analyst therefore needs to learn about data structure in order to decide upon the way of further data analysis.

Data reduction is a part of EDA which reduces the amount of data by removing redundant or irrelevant examples from a data set.[1] Hence, data reduction speeds up EDA because a data set becomes smaller. Typical instances of data reduction include nearest neighbour editing and its variants [2–4]. These techniques need class membership of examples to determine which examples to preserve in a data set. In contrast, unsupervised data reduction got much less attention in the scientific literature. Nevertheless, it is necessary for data

*Corresponding author.
E-mail address: oleg@ee.oulu.fi (O. Okun).

[1] We distinguish data and dimensionality reduction. The latter reduces the number of features.

0165-1684/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2007.02.006

exploration when class membership of examples is unknown or unavailable. One can argue that cluster analysis, which is synonymous with unsupervised pattern recognition [1], could be a solution in this case. However, cluster analysis algorithms, such as fuzzy C-means (FCM) [5], provide a single example associated with each cluster to characterise it. Besides, the main goal of cluster analysis is to partition the data into groups but not to reduce data set size.

In this paper, we argue that it would be highly desirable to have a set of examples best representing each cluster,[2] because certainly a set of examples provide more information about cluster and its structure than a single representative. Our approach to data reduction is based on a combination of FCM and two nonnegative matrix factorisations. All these methods are unsupervised in the sense that they do not require class labels of examples in a data set to be known in advance. This can greatly facilitate data analysis since it is often difficult or even impossible to manually label large data sets. The novelty of our approach is that it associates a set of examples with each cluster rather than a single representative, hence better understanding of cluster structure is achieved when examples in a data set do not have labels.

2. Previous work and potential improvements

In this paper, we assume that examples are collected into a $d \times n$ matrix, where d and n are dimensionality and cardinality of a data set.

Nonnegative matrix factorisation (NMF) of a $d \times n$ matrix V can be done as $V \approx WH$ (W is of size $d \times r$ and H is of size $r \times n$), subject to constraints that the elements of all matrices are nonnegative. NMF is an iterative algorithm [6], hence its convergence is strongly affected by initialisation as shown in [7–9]. In the absence of any guidance, the initialisation is typically random. Therefore, different runs of NMF converge to different local optima. It was shown in [8] that this fact affects classification accuracy.

A combination of FCM and NMF has been recently proposed [7,9]. The idea of this combination is to initialise matrices W and H with the FCM results.[3] For FCM, the number of clusters C should be fixed in advance. Given this number, the value of r in NMF is set to C, and W is initialised with d-dimensional cluster centroids obtained after the FCM convergence [7,9]. Another matrix, H, is initialised with membership values assigned to each of the n examples [7,9]. These memberships are fuzzy rather than crisp, which leads to the fact that each example belongs to each of the r clusters to some degree. As a result, both W and H are deterministically initialised, which implies that the outcome is no longer dependent on random initialisation.

Both FCM and NMF above work with $d \times n$ matrices as their inputs. If $d > n$ or $d \gg n$, as can occur, e.g., in face recognition, both FCM and NMF can be slow. Instead, we employ $n \times n$ matrices, resulting in faster convergence of both algorithms. This is our first improvement of the previous methods such as [7,9]. The second improvement comes from the fact that the final result of either method [7,9] cannot say anything useful about cluster structure since each cluster is represented by a single pattern, namely the cluster centroid. Besides, centroids are usually mixtures of examples rather than the actual examples. Our approach described in detail below assigns to each cluster a group of actual examples, thus, greatly facilitating the data exploration task for analysts. In contrast to approaches in [7,9] relying on the standard NMF as proposed by Lee and Seung in [6], we employ two extensions of the standard algorithm proposed by Ding and his colleagues in [10,11], which, to our best knowledge, have not yet been applied in combination with FCM.

3. Our approach

First, FCM groups examples into a pre-specified number of clusters, followed by two NMFs. The first factorisation uses the result of clustering and “unfolds” clusters so that their structure becomes more transparent. The second factorisation that takes the output provided by the first factorisation imposes orthogonality constraints on the resulting matrix whose entries can be treated as the posterior probabilities for examples to belong to clusters. As a result, certain examples have all but one zero (or

[2] Though the cluster membership value can be used as the indicator of how strongly a given example belongs to a cluster, it is often difficult to automatically determine a threshold separating best representing examples for each cluster from other cluster members.

[3] As demonstrated in [8,9], other methods can be used for initialisation as well.