ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

Visualizing asymmetric proximities with MDS models

Alberto Muñoz and Manuel Martin-Merino
University Carlos III of Madrid, C/Madrid 126, 28903 Getafe, Spain

Email: albmun@est-econ.uc3m.es
University Pontificia of Salamanca, C/Compañía 5, 37002 Salamanca, Spain
Email: mmerino@ieee.org

Abstract. Multidimensional scaling algorithms (MDS) are useful to visualize object relationships using a matrix of dissimilarities (or proximities). A common assumption is symmetry: $\delta_{ij} = \delta_{ji}$. Many of these algorithms are computationally demanding. An interesting application is the representation of topics in text data bases. In this paper we extend an MDS algorithm suitable to deal with textual data to the asymmetric case, and propose some improvements to reduce the computational complexity. The final algorithm can be considered an improved alternative to the Asymmetric Self Organizing Maps (ASOM) presented in the literature. Experimental work shows the performance of the proposed algorithm on text mining and gene expression databases.

1 Introduction

Consider a set of n objects represented as vectors in $\mathbb{R}^m$. Let $D = (\delta_{ij})$ be the dissimilarity matrix made up of object dissimilarities. Asymmetry arises when $\delta_{ij} \neq \delta_{ji}$. There are a large variety of problems in which object relations should be modeled by asymmetric dissimilarities [16]. For instance when modeling word relations many people will relate "neural" to "networks" more strongly than conversely. Similarly gene expression data should be modeled by asymmetric similarities in the sense that broad genes that are shared by several diseases include more specific genes but the reverse relation is weaker. Besides, those applications are computationally demanding. Therefore any algorithm proposed should be able to work with asymmetric dissimilarities while being scalable with the number of patterns.

SOM [6] and MDS [10] algorithms have been widely used to generate visual representations of object relations. In [8] it has been presented a simple iterative MDS algorithm based on classic mechanics that achieves a balance between clusters separation and distances preservation. This behavior gives rise

Financial support from DGICYT grant BEC2000-0167 (Spain) is gratefully appreciated.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

to maps in which overlap between groups of similar objects is reduced. However this mapping algorithm relies on the use of symmetric similarities. This fact degrades the map quality when object relations are asymmetric. Besides, the computational complexity is excessively high to be applicable to large datasets. In this paper we first extend the MDS algorithm mentioned above to deal with asymmetric dissimilarities. Next the computational efficiency is increased by representing the objects by a low number of prototypes. Note that this new algorithm stands up as a fast alternative to the asymmetric SOM algorithm presented in [11].
This paper is organized as follows. Section 2 review the symmetric spring model and extends it to asymmetric dissimilarities. Section 3 proposes some straightforward ideas to increase computational efficiency in a SOM-like fashion. In section 4 we compare with well known alternative techniques in two real problems and finally, section 5 gets conclusions and points out some directions for future work.

2 Asymmetry

Let $D = (\delta_{ij})$ be the dissimilarity matrix between the objects. Asymmetry arises when $(\delta_{ij} \neq \delta_{ji})$. In this case the dissimilarity matrix can be decomposed into a symmetric and skew symmetric component $(D = S + A)$ [16] where $s_{ij} = (\delta_{ij} + \delta_{ji})/2$ and $a_{ij} = (\delta_{ij} - \delta_{ji})/2$. When asymmetry is very strong, that is $(\delta_{ij} \gg \delta_{ji})$ the symmetric component of the dissimilarity matrix gives lower values than expected [12]. Therefore, distances between the corresponding objects in the map become too large. To prevent this fact, object proximities (s_{ij}) should be compensated proportionally to the value of the skew symmetric component of the similarity matrix (a_{ij}).
Let now compute (a_{ij}) for the widely used fuzzy logic asymmetric similarity measure [9],

$$a_{ij} = \frac{|x_i \wedge x_j|}{|x_i|} - \frac{|x_j \wedge x_i|}{|x_j|} \propto |x_j| - |x_i| \tag{1}$$

This expression shows that asymmetry is a property associated to individual objects and may be modeled by the following coefficient of asymmetry, $l_i = \frac{| x_i |}{\max_k | x_k |}$. This coefficient will be used in section 2.2 to define an asymmetric dissimilarity that allows to reduce the object distances in the map corresponding to asymmetric relations.
Notice that L_1 norm histogram is very skew and obeys a Zipf's law in our practical problems. Therefore the skew-symmetric component of the similarity matrix (1) will take large values for certain objects becoming an important problem that should be corrected.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

2.1 Symmetric spring model: background

This MDS model has been presented in [8]. Each object is represented by a mass point and they are connected to each other by springs of elasticity proportional to the similarity between the objects . The mass point coordinates are updated until convergence where the final point distances represent the dissimilarity between the objects. Let m_i be the mass of each point, $\Delta x_{ij} = x_j - x_i$, x_j the vector position of word j in the two dimensional Euclidean space, k_i the frictional resistance coefficient and e_{ij} the elasticity coefficient.
Then the forces applied over particle i are: $f_{mi} = -m_i \ddot{x}_i, f_{ki} = -k_i \dot{x}_i, f_{eij} = e_{ij} \Delta x_{ij}$ .f_{mi} is the inertia force, f_{ki} is the frictional resistance force and f_{eij} is the elasticity force that is proportional to the similarity between the words. In equilibrium the sum of all forces over particle i has to be 0.
This equation gives after some approximations detailed in [8] a simple updating solution for each component k,

$$x_i^k(t+1) = x_i^k(t) + \frac{\Delta t}{k_i} \sum_j e_{ij} \Delta x_{ij}^k \tag{2}$$

where $e_{ij} = \frac{J_{ij} - T}{\max\limits_{ij}(J_{ij}) - T}$. J_{ij} is any symmetric similarity such as the Jaccard similarity. T is an experimental parameter that controls that forces between particles of the same cluster are attractive and forces between particles of different clusters are repulsive. It may be fixed by analysis of the J_{ij} histogram. Δt is the step length and $k_i = 1$ for all particles.
It can be easily shown that the second term of equation (2) optimizes an index similar to a correlation measure. This suggests that the improvements proposed in this paper may be easily extended to a broad class of MDS algorithms.

2.2 Asymmetric spring model by incorporating asymmetric distances

A natural way to incorporate asymmetry and that is related to the work developed by the MDS community [14], is to define asymmetric distances between the mass points. In this case distances can be expressed as a symmetric component (for instance Euclidean distance) plus a skew symmetric one. For the sake of clarity let examine one example in which asymmetric distances arise and that we encounter everyday. It is not the same to walk the hill up than down although we move the same distance. So we may define an effective distance that include the effect of the gravity force. This distance will be asymmetric and has to be larger when going up than when going down.
We then define the asymmetric vector of difference coordinates between two mass points as

$$\overline{\Delta x a_{ij}} = ||\overline{\Delta x a_{ij}}|| \overline{u_{ij}} = \left(||\overline{\Delta x_{ij}}|| + \frac{l_j - l_i}{2 max_k(l_k)} \right) \overline{u_{ij}} \tag{3}$$

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

where $||\overline{\Delta xa_{ij}}||$ is the asymmetric distance defined over the visual map and $\overline{u_{ij}}$ is a unitary vector in the direction of the line joining i and j. l_i is the asymmetry coefficient of object i defined in section 2. The symmetric component of $d^a_{ij} = ||\overline{\Delta xa_{ij}}||$ reduces to the Euclidean case. On the other hand the skew-symmetric component is proportional to $l_j - l_i$ and so to the asymmetric component of the fuzzy logic similarity defined by (1). This term allows that $d^a_{ji} > d^a_{ij}$ if j has larger asymmetry coefficient than i. By substituting $\overline{\Delta xa_{ij}}$ into the expression of the elastic force (see section 2) we get

$$f_{eij} = e^{(s)}_{ij} \left(||\overline{\Delta x_{ij}}|| + \frac{l_j - l_i}{2max_k(l_k)} \right) \overline{u_{ij}} \tag{4}$$

where $e^{(s)}_{ij}$ is the symmetric component of the similarity matrix. This expression allows to explain better how the new dissimilarity reduces the Euclidean distances in the map associated to asymmetric relations. Equation (4) shows that forces due to terms with large asymmetry coefficient get stronger due to asymmetry. Therefore, Euclidean distances between objects that verify $(l_j - l_i) \uparrow\uparrow$ (corresponding to asymmetric relations) get smaller in the map.
Notice that this feature reduces also the percentage of dissimilarities that are close to 1. Therefore it avoids partially that specific words concentrate strongly around the center map due to the indiferenciation effect (see [3] for more detail about this problem).

3 An efficient alternative to asymmetric SOM and MDS algorithms

Computational complexity of the MDS algorithm presented in section 2.2 is quadratic with the number of patterns, $\mathcal{O}(N^2)$. In this section we propose two methods that achieve linear complexity with the number of patterns. The ideas proposed reduce the number of prototypes used by the MDS algorithm in such a way that asymmetry is preserved. This fact will improve the maps generated by the asymmetric MDS algorithms.
Note that both methods are suitable to work with any dataset that verifies Zipf's law for L_1 norm histogram.
The first method is related to [7] but asymmetry is better preserved. We propose to generate a low number of prototypes by a k-means algorithm that take into account term frequency distribution. Only the 50% of larger L_1 norm terms are submitted to the quantization algorithm. Rare terms are neglected because they hardly provide information about word map structure and are very noisy [15]. This term set is divided into r regions according to the L_1 norm histogram . The number of regions and the cut-points between regions are chosen experimentally (see [12] for details).
Besides, as we have mentioned in section 2 asymmetry is related to term's L_1 norm. Therefore, to avoid that asymmetry is partially lost, each region is independently submitted to the quantization algorithm with a number of

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

prototypes proportional to the number of patterns per region. Finally each vector is represented by the nearest prototype image provided by the mapping algorithm in a SOM like fashion.
The second method selects a small proportion of the larger L_1 norm terms (usually $p < 10\%$). Points that have not been used by the MDS algorithm are interpolated using any standard method proposed in the literature [10].
Computational complexity for the first technique is roughly $\mathcal{O}(K^2) + \mathcal{O}(KdN)$ where K is the number of prototypes computed, d is the vector space dimension and N is the number of data. The first term corresponds to the complexity of the asymmetric MDS algorithms proposed in section 2.2. The second term is due to the k-means clustering algorithm. However it reduces to $\mathcal{O}(N)$ if we make use of the sparsity of our document vector space and apply any of the accelerating techniques proposed in [13]. So for $K \ll N$ the whole algorithm complexity is roughly $\mathcal{O}(N)$. Notice also that complexity of the alternative SOM algorithm is significantly larger, $\mathcal{O}(KN^2)$ [6].
Computational complexity for the second technique is roughly $\mathcal{O}(K^2)$ with K the number of patterns selected. For $K \ll N$ this yields great computational savings. Experimental results show that values as low as 5% of N give excellent outcomes.

4 Experimental results

In this section we apply our algorithms to the construction of word maps that visualize word relations. Next we carry out some preliminary experiments to check the applicability of our asymmetric model to DNA microarray data analysis [4].
Assessing the performance of algorithms that produce word maps is not an easy task. We will use a thesaurus to check if neighboring words in the map are related in the thesaurus. Notice that there is no a priori classification of words into topics for large document collections. Therefore we have built a database made up of 1000 documents that group in 7 topics according to the available thesaurus.
To check the quality of the mapping algorithm we will first evaluate if neighbor's order in the original document space is preserved in the word map. To this aim we will use the Spearman correlation coefficient [2]. Next we check if words belonging to the same cluster according to the map are assigned to the same group by the thesaurus. For this purpose, we first run the MDS algorithms and cluster the word map into 7 groups using PAM algorithm [5]. Word clusters are evaluated using the following measures: The F measure [1] shows if words from the same class according to the thesaurus are clustered together. The entropy measure [1] gives the uncertainty for the classification of words from the same cluster. Small values suggest little overlapping in the map between words belonging to different classes. Finally Mutual Information [1] is a nonlinear correlation measure between the word classification induced by the thesaurus and the word classification given by the clustering algorithm. This measure

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 51-58

gives more weight to specific words and therefore informs about changes in the position of specific terms.
Documents are represented using the vector space model [1]. First a low number of prototypes are generated using any of the methods proposed in section 3. Next the prototype vectors are mapped and clustered into 7 groups using a PAM clustering algorithm . Each object is classified to the cluster of the nearest neighbor prototype in a SOM-like fashion. Object coordinates for the asymmetric algorithms are updated by equation (2) where $\overline{\Delta x_{ij}}$ incorporates the asymmetric dissimilarity given by equation (3). The parameter T for the spring model is taken for all experiments as the 0.75 quantile of the similarity matrix.

	Text mining data				MDNA
	Sp	F	Ent.	M. Inf	Sp.
(1) Symmetric SOM	0.55	0.52	0.53	0.19	0.52
(2) Symmetric spring (over all data)	0.27	0.50	0.51	0.20	
(3) Asym. SOM	0.58	0.54	0.50	0.20	0.55
(4) Asym. Spr. (10% random sample)	0.40	0.50	0.44	0.15	0.39
(5) Asym. Spr. (k-means 10% cent. over 50% data)	0.70	0.56	0.48	0.20	
(6) Asym. Spr. (k-means 5% cent. over 50% data)	0.73	0.54	0.46	0.20	0.60
(7) Asym. Spr. (10% larger L_1 norm)	0.61	0.60	0.45	0.22	0.67

Table 1: Comparison of asymmetric spring models (5) (6) (7) with some alternatives proposed in the literature for text mining data and Microarray DNA data (MDNA).

Table 1 shows that the first asymmetric mapping algorithm (5) with a k-means preprocessing step (see section 3) outperforms both symmetric (1) and asymmetric (3) SOM [11] algorithms. Our algorithm allows to capture better the word clustering structure improving F measure up to 7.7% and reducing clustering overlapping (Entropy) up to 9.4%. Spearman coefficient supports that distances are also better preserved. Our algorithm (5) outperforms the symmetric counterpart (2) as well. In this case F measure increases up to 12%, entropy is reduced up to 5.9% and distances are preserved much better. Row (6) shows that results hardly degrade when the number of prototypes is reduced. Row (4) shows that selecting a low number of prototypes by random sampling gives poor results. Last technique (7) in which 10% of larger L_1 norm words have been selected yields excellent results.
Finally last column of table 1 gives the Spearman coefficient for DNA Microarray data. We point out that results are promising and similar to those obtained for text mining data. We have not computed supervised measures due to the lack of an expert that provides gene relations.
Notice that as has been mentioned in section 3 our algorithms (6) (7) (8) are

computationally efficient and so may be applied to large datasets.

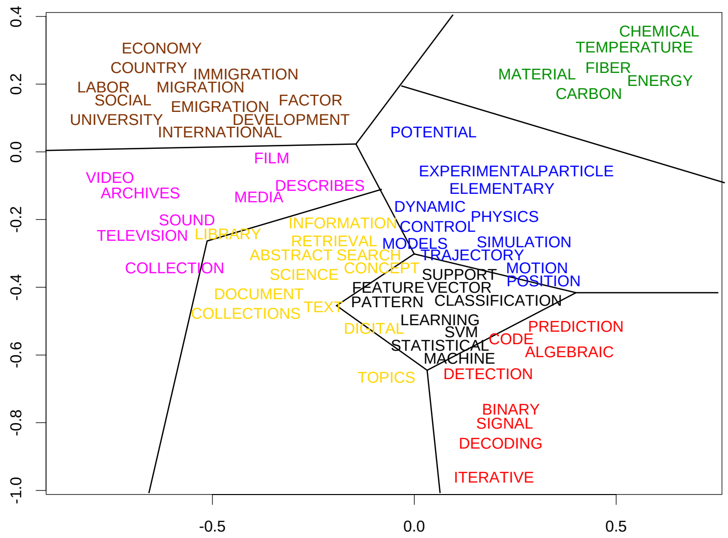

Figure 1: Word map generated by the asymmetric MDS model proposed.

Finally we show in figure 1 the visual map generated by our asymmetric MDS algorithm. Each class according to the classification induced by the thesaurus has been shown with different gray intensity. The boundaries between classes have been drawn in bold line. For the sake of clarity only a small sample of words is shown. The word map suggests successfully which words belong to the same topic and the semantic relations between different topics. Note that overlapping between term clusters in the map is small.

5 Conclusions

In this work we have extended a class of MDS algorithm to deal with asymmetric dissimilarities. Next we have proposed some straightforward techniques to make the resulting algorithm suitable to work with large datasets. Results show that the proposed algorithms outperform both SOM and other MDS based techniques according to several objective measures. Besides, our algorithm scale linearly with the number of data. Future research will focus on the study of asymmetric efficient models that work directly from the dissimilarity matrix.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison Wesley, Wokingham, UK, 1999.

[2] J. C. Bezdek and N. R. Pal . An Index of Topological Preservation for Feature Extraction, Pattern Recognition, 28(3):381-391, 1995.

[3] A. Buja, B. Logan, F. Reeds and R. Shepp. Inequalities and positive default functions arising from a problem in multidimensional scaling, Annals of Statistics, 22, 406-438, 1994.

[4] T. Hastie, T. Tibshirani and J. Friedman. The Elements of Statistical Learning, Springer Verlag, Heidelberg, 2001 (http://www-stat.stanford.edu/ tibs/ElemStatLearn/index.html).

[5] L. Kaufman and P. J. Rousseeuw. Finding Groups in Data. An Introduction to Cluster Analysis. John Wiley & Sons. New York. 1990.

[6] T. Kohonen, S. Kaski, K. Lagus, J. Salojarvi, J. Honkela, V. Paatero and A. Saarela. Organization of a Massive Document Collection. IEEE Transactions on Neural Networks, 11(3):574-585, 2000.

[7] A. König, Interactive Visualization and Analysis of Hierarchical Neural Projections for Data Mining, IEEE Transactions on Neural Networks, 11(3), 615-624, 2000.

[8] A. Kopcsa and E. Schievel. Science and Technology Mapping: A New Iteration Model for Representing Multidimensional Relationships. Journal of the American Society for Information Science, 49(1): 7-17, 1998.

[9] B. Kosko. Neural Networks and Fuzzy Systems: A Dynamical Approach to Machine Intelligence. Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[10] L. Lebart, A. Morineau and J. F. Warwick. Multivariate Descriptive Statistical Analysis, John Wiley, New York, 1984.

[11] M. Martin-Merino and A. Muñoz. Self Organizing Map and Sammon Mapping for Asymmetric Proximities, ICANN, LNCS 2130, 429-435, Springer Verlag, 2001.

[12] A. Muñoz. Compound key word generation from document databases using a hierarchical clustering ART model. Journal of Intelligent Data Analysis, 1(1), 1997.

[13] D. Pelleg and A. Moore. Accelerating exact k-means algorithms with geometric reasoning. In SIGKDD, 277-281, San Diego, USA, ACM, 1999.

[14] T. Saito, Analysis of Asymmetry Proximity Matrix by a Model of Distance and Additive Terms, Behaviormetrika, 29, 45-60, 1991.

[15] Y. Yang and J. O. Pedersen, A Comparative Study on Feature Selection in Text Categorization, Proc. of the 14th International Conference on Machine Learning, 412-420, Nashville, Tennessee, USA, July, 1997.

[16] B. Zielman and W.J. Heiser. Models for asymmetric proximities. British Journal of Mathematical and Statistical Psychology, 49:127-146, 1996.