Spatial Fuzzy Clustering Using Varying Coefficients

Huaqiang Yuan, Yaxun Wang, Jie Zhang, Wei Tan, Chao Qu, and Wenbin He

Department of Computer Science, DongGuan University of Technology
DongGuan, GuangDong 523808, China
hyuan66@163.com

Abstract. To consider spatial information in spatial clustering, the Neighborhood Expectation-Maximization (NEM) algorithm incorporates a spatial penalty term in the objective function. Such an addition leads to multiple iterations in the E-step. Besides, the clustering result depends mainly on the choice of the spatial coefficient, which is used to weigh the penalty term but is hard to determine a priori. Furthermore, it may not be appropriate to assign a fixed coefficient to every site, regardless of whether it is in the class interior or on the class border. In estimating class posterior probabilities, sites in the class interior should receive stronger influence from their neighbors than those on the border. To that end, this paper presents a variant of NEM using varying coefficients, which are determined by the correlation of explanatory attributes inside the neighborhood. Our experimental results on real data sets show that it only needs one iteration in the E-step and consequently converges faster than NEM. The final clustering quality is also better than NEM.

1 Introduction

Compared to conventional data, the attributes under consideration for spatial data include not only non-spatial normal attributes, but also spatial attributes that describe the object's spatial information such as location and shape. The assumption of independent and identical distribution is no longer valid for spatial data. In practice, almost every site is related to its neighbors. To that end, Ambroise et al. proposed the Neighborhood Expectation-Maximization (NEM) algorithm [1], which incorporates a spatial penalty term in the objective function to encourage neighboring sites with similar class posterior probabilities. In contrast to the standard EM algorithm [2] that maximizes likelihood alone, such an addition involves multiple iterations in the E-step. Besides, the clustering results rely heavily on the spatial coefficient, which specifies the degree of spatial smoothness in the clustering solution but is hard to determine a priori in practice. Furthermore, it may not be appropriate to assign a fixed coefficient to every site, regardless of whether it is in the class interior or on the class border.

Based on the observation above, this paper presents a Neighborhood EM algorithm using Varying coefficients (NEMV). Rather than set empirically, the

R. Alhajj et al. (Eds.): ADMA 2007, LNAI 4632, pp. 183–190, 2007.

coefficient is determined by the correlation of true explanatory attributes inside the neighborhood. Our experimental results on real data sets show that it only needs one iteration in the E-step and consequently converges faster than NEM. The final clustering quality is also consistently better than NEM.

The rest of the paper is organized as follows. Section 2 reviews the problem background and related work. In Section 3, we first outline NEM and then present our NEMV algorithm. Experimental evaluation is reported in Section 4. Finally Section 5 concludes this paper with a summary and discussion of future work.

2 Background and Related Work

In this section, we first introduce the background by formulating the problem. Then we briefly review related work.

2.1 Problem Formulation

The goal of spatial clustering is to partition data into groups or clusters so that pairwise dissimilarity, in both attribute space and spatial space, between those assigned to the same cluster tend to be smaller than those in different clusters. Let S denote the set of locations, e.g., the set of triple (index, latitude, longitude). Spatial clustering can be formulated as an unsupervised classification problem. We are given a spatial framework of n sites,$S = \{s_i\}_{i=1}^n$ with a neighbor relation $N \subseteq S \times S$. Sites s_i and s_j are neighbors iff $(s_i, s_j) \in N, i \neq j$. Let $N(s_i) \equiv \{s_j : (s_i, s_j) \in N\}$ denote the neighborhood of s_i. We assume N is given by a contiguity matrix W whose $W(i,j) = 1$ iff $(s_i, s_j) \in N$ and $W(i,j) = 0$ otherwise. Associated with each s_i, there is a d-dimensional feature vector of normal attributes $\mathbf{x}_i \equiv \mathbf{x}(s_i) \in \Re^d$. We need to find a many-to-one mapping $f : \{\mathbf{x}_i\}_{i=1}^n \rightarrow \{1, ..., K\}$. If each object $\mathbf{x}_i$ has a true class label $y_i \in \{1, ..., K\}$, naturally the ultimate goal is to maximize similarity between obtained clustering and true classification. However, since the class information is unavailable during learning, the objective in practice is to optimize some criterion function such as likelihood. Besides, spatial clustering imposes the following constraint of spatial autocorrelation. y_i is not only affected by $\mathbf{x}_i$, but also by $(\mathbf{x}_j, y_j)$ of its neighbors $N(s_i)$. Hence it is more appropriate to model the distribution of y_i with $P(y_i \mid \mathbf{x}_i, \{(\mathbf{x}_j, y_j) : s_j \in N(s_i)\})$ instead of $P(y_i|\mathbf{x}_i)$.

2.2 Related Work

Many methods have been proposed to incorporate spatial information in the clustering process. The simplest one is adding spatial information, e.g., spatial coordinates, directly into datasets [3]. Others achieve this goal by modifying existing algorithms, e.g., allowing an object assigned to a class if and only if this class already contains its neighbor [4]. Another class, where our algorithm falls, selects a model that encompasses spatial information [1]. This can be achieved by modifying a criterion function that includes spatial constraints [5], which mainly comes from image analysis where Markov random field and EM style algorithms were intensively used [6,7].

Clustering using mixture models with EM can be regarded as a soft K-means algorithm in that the output is posterior probability rather than hard classification. It does not account for spatial information and usually cannot give satisfactory performance on spatial data. NEM extends EM by adding a spatial penalty term in the criterion, but this makes it need more iterations in each E-step. If further information about structure is available, the structural EM algorithm may be used to learn Bayesian networks for clustering [8]. In our case, we assume that soft constraints can be derived with locations of sites. Another relevant problem is semi-supervised clustering, where some pairs of instances are known belonging to same or different clusters [9]. In their case, the goal is to fit the mixture model to the data while minimizing the violation of hard constraints.

3 The NEMV Algorithm

In this section, we first outline the basics of NEM. Then we present the NEMV algorithm.

3.1 NEM for Spatial Clustering

We assume the data $X = \{\mathbf{x}\}_{i=1}^{n}$ come from a mixture model of K components $f(\mathbf{x}|\Phi) = \sum_{k=1}^{K} \pi_k f_k(\mathbf{x}|\theta_k)$, where π_k is k-th component's prior probability, missing data (cluster label) $y \in \{1, ..., K\}$ indicate which component $\mathbf{x}$ comes from, i.e., $p(\mathbf{x}|y = k) = f_k(\mathbf{x}|\theta_k)$, and Φ denotes the set of all parameters. Because it is hard to directly maximize the sample likelihood $L(\Phi) = \sum_{i=1}^{n} \ln \left[f(\mathbf{x}_i|\Phi) \right]$, EM tries to iteratively maximize L in the context of missing data y. Let $\overline{P}$ denote a set of fuzzy classifications representing the grade of membership of $\mathbf{x}_i$ to class (component) k, i.e., $\{\overline{P}_{ik} \equiv \overline{P}(y_i = k)\}$. As highlighted in [10], the new objective function U of NEM that incorporates a spatial penalty term can be written as

$$U(\overline{P}, \Phi) = F(\overline{P}, \Phi) + \beta G(\overline{P})$$

where

$$\begin{aligned} F(\overline{P}, \Phi) &= E_{\overline{P}}[\ln(\, P(\{\mathbf{x}, y\}|\Phi)\,)] + H(\overline{P}) \\ &= \sum_{i=1}^{n} \sum_{k=1}^{K} \overline{P}_{ik} \ln(\pi_k f_k(\mathbf{x}|\theta_k)) - \sum_{i=1}^{n} \sum_{k=1}^{K} \overline{P}_{ik} \ln \overline{P}_{ik} \\ G(\overline{P}) &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij} \sum_{k=1}^{K} \overline{P}_{ik} \overline{P}_{jk} \end{aligned}$$

Compared to standard EM, in addition to maximizing $L(\Phi)$ which is achieved by maximizing $F(\overline{P}, \Phi)$, NEM also tries to increase $G(\overline{P})$, the spatial penalty that encourages neighboring sites with similar class posterior probabilities. The spatial penalty is weighted by $\beta > 0$, a fixed coefficient that determines the degree of smoothness in the solution clustering. U can be maximized by alternately estimating its two parameters $\overline{P}$ and Φ. Starting from an initial $\overline{P}^0$, NEM iterates the following two steps:

1. M-step: With $\overline{P}^t$ fixed, set $\Phi^t = \mathrm{argmax}_{\Phi} U(\overline{P}^t, \Phi)$, which is exactly the same as the M-step in EM, for G dose not depend on Φ.
2. E-step: With Φ^t fixed, set $\overline{P}^{t+1} = \mathrm{argmax}_{\overline{P}} U(\overline{P}, \Phi^t)$ by applying Eq. (1) repeatedly until convergence.

$$\overline{P}^*_{ik} = \frac{\pi_k f_k(\mathbf{x}_i|\theta_k)\exp\left(\beta \sum_{j=1}^n W_{ij}\overline{P}^*_{jk}\right)}{\sum_{l=1}^K \pi_l f_l(\mathbf{x}_i|\theta_l)\exp\left(\beta \sum_{j=1}^n W_{ij}\overline{P}^*_{jl}\right)} \qquad (1)$$

3.2 NEM with Varying Coefficients

EM is not appropriate for spatial clustering since it does not account for spatial information. In contrast, although NEM incorporates spatial information, it requires multiple iterations in E-step and the spatial coefficient is hard to determine a priori. To overcome these difficulties, we propose NEMV, which is based on the observation that it may not be appropriate to assign a constant coefficient to every site. For those in the class interior, the whole neighborhood is from the same class and hence the site should receive more influence from its neighbors, especially when their posterior estimates are accurate. For those on the class border, because their neighbors are from different classes, its own class membership should be determined mostly by its own explanatory attributes.

Along this line, NEMV employs a site-sensitive spatial coefficient, the local Moran's I measure, which is determined by the correlation of explanatory attributes inside the neighborhood [11]. Let z_{ip} denote the normalized p-th attribute of site s_i, i.e., $z_{ip} = x_{ip} - \overline{x}_p$, where $\overline{x}_p$ is the global mean of the p-th attribute. Let σ_p denote the global standard deviation of the p-th attribute. Then, for the p-th attribute at site s_i, the local I measure is defined as $I_{ip} = \frac{z_{ip}}{\sigma_p^2}\sum_j W_{ij} z_{jp}$, where W is a row-normalized (sum to 1) version of the original binary W. A high I (e.g., $I > 1$) implies a high local spatial autocorrelation at site s_i, which is likely to occur in the class interior. In NEMV, β_i is obtained by first averaging I_{ip} over all attributes and then normalizing to $[0, 1]$, i.e., $I_i = \mathrm{mean}_p(I_{ip})$, $\beta_i = \frac{I_i - \min_i\{I_i\}}{\max_i\{I_i\} - \min_i\{I_i\}}$. Then the new penalty and criterion become

$$G = \frac{1}{2}\sum_{i=1}^n \beta_i \sum_{j=1}^n W_{ij} \sum_{k=1}^K \overline{P}_{ik}\overline{P}_{jk}$$

$$U(\overline{P}, \Phi) = F(\overline{P}, \Phi) + G(\overline{P})$$

Besides, we find that one iteration is usually enough for the E-step in NEMV. Therefore, NEMV proceeds as follows.

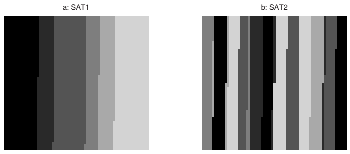

Fig. 1. Satimage data with site's location synthesized: (a) SAT1 (contiguity ratio 0.96) and (b) SAT2 (contiguity ratio 0.89)

1. E-step: Set $\overline{P}^t = \text{argmax}_{\overline{P}} U(\overline{P}, \varPhi^{t-1})$ by applying Eq. (1) once, where β has been replaced by β_i in Eq. (1).
2. M-step: Set $\varPhi^t = \text{argmax}_{\varPhi} U(\overline{P}^t, \varPhi)$, which is exactly the same as the M-step in EM.

4 Experimental Evaluation

In this section, we first introduce the clustering validation measures used in our experiments. Then we report comparative results on two real datasets.

4.1 Performance Criteria

If every site has a true class label, they can be used to evaluate the final clustering quality via external validation measures. Let $C, Y \in \{1, ..., K\}$ denote the true class label and the cluster label, respectively. Then clustering quality can be measured with conditional entropy $H(C|Y)$ defined in Eq. (2), which equals zero if their distributions are the same. We also use a more intuitive measure, error rate $E(C|Y)$, which computes the misclassified fraction of data in each cluster after assuming the true class label should be the major class in the cluster. The above two measures are only for the discrete target value. When the target variable C is continuous, we calculate the standard deviation defined in Eq. (3), where std$(\cdot)$ denotes the standard deviation operator and $(C|Y = k)$ denotes the C's values in cluster $Y = k$.

$$H(C|Y) = \sum_{k=1}^{K} P(Y = k) \times H(C|Y = k) \tag{2}$$

$$S(C|Y) = \sum_{k=1}^{K} P(Y = k) \times \text{std}(C|Y = k) \tag{3}$$

4.2 Experimental Results

Satimage Dataset. We first evaluate NEMV on a real landcover dataset, Satimage, which is available at the UCI repository. It consists of the four multi-spectral values of pixels in a satellite image together with the class label from a

Table 1. Clustering performance on the Satimage dataset: $^+$SAT1 and *SAT2

			SAT1		SAT2	
	supervised	EM	NEM	NEMV	NEM	NEMV
entropy	0.5121	0.6320	0.5391	0.5094	0.5635	0.5340
error	0.1508	0.2315	0.2039	0.1816	0.2142	0.2004
$-U(10^4)$	5.1884^+ 5.2274^*	5.1406^+ 5.1717^*	5.1029	5.0926	5.1416	5.1102
$-L(10^4)$	5.8128	5.7711	5.8207	5.7842	5.8141	5.7804

Table 2. Clustering performance on the Satimage dataset by NEMV with varying number of iterations of E-step

	SAT1				SAT2		
#E-step	1	10	20	30	1	5	10
entropy	0.5094	0.5092	0.5088	0.5086	0.5340	0.5332	0.5330
error	0.1816	0.1813	0.1810	0.1809	0.2004	0.2001	0.2000
$-U(10^4)$	5.0926	5.0916	5.0913	5.0912	5.1102	5.1101	5.1099
$-L(10^4)$	5.7842	5.7836	5.7834	5.7833	5.7804	5.7802	5.7801

six soil type set. Because the dataset is given in random order and there is no spatial location, we synthesize their spatial coordinates and allocate them in a 64×69 grid. 4-neighborhood is used in construction of W and contiguity ratio is computed as the fraction of edges shared by the pixels from the same class. To emphasize spatial autocorrelation, we generate two images SAT1 and SAT2 in Fig. 1(a) and (b) with high contiguity ratios 0.96 and 0.89, respectively.

We test NEM and set $\beta = 1$ empirically to maximize U. With random initialization, Table 1 gives the average results of 10 runs recorded at maximum L for EM, and maximum U for NEM and NEMV. The U values for EM are computed using the definition in NEM. For clarity, we report $-L$ and $-U$ so that all criteria in the tables are to be minimized. Note that due to different β used in NEM and NEMV, it is meaningless to compare U for them. For comparison, we also list the results under supervised mode where each component's parameters are estimated with all data from a single true class. We can see that the entropy and error generally decrease as $-U$, rather than $-L$, decreases. NEMV gives better results than NEM. As expected, both of them perform better on SAT1 than on SAT2, for the former's contiguity ratio is higher and hence fits our assumption more.

To see if one iteration of E-step is really enough in NEMV, we perform a series of experiments by varying the number of iterations of E-step. The average results of 10 runs are shown in Table 2. Note that 30/10 is the number of iterations of E-step we used in the standard NEM. Although the computational cost has been increased by an order of magnitude, we can see that the improvement is not significant, especially in error rate and U.

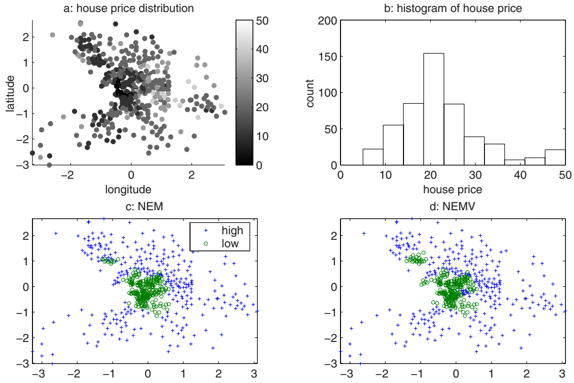

Fig. 2. (a) shows house price distribution in 506 towns in Boston area. The corresponding histogram is plotted in (b). Two sample clustering results are shown in (c) and (d) for NEM and NEMV, respectively.

Table 3. Clustering performance on the house dataset

	EM	NEM	NEMV
std	8.3377	8.3486	8.3088
$-U(10^4)$	1.2580	1.2675	1.2557
$-L(10^4)$	1.3942	1.4014	1.3966

House Dataset. We also evaluate NEMV on a real house price dataset with 12 explanatory variables, such as nitric oxides concentration and crime rate. The clustering performance is evaluated with the target variable, median values of owner-occupied homes, which is expected to has a small spread in each cluster. Fig. 2(a) shows the true house values of 506 towns in Boston area. Their histogram is plotted in Fig. 2(b). Using a Gaussian mixture of two components, we set $\beta = 1$ for NEM and about 20 iterations are needed for convergence in its E-step. Table 3 gives the average results of 10 runs. One can see that NEM performance is slightly worse than EM in terms of either standard deviation or U. But NEMV still gives the best results. Two sample clustering results are shown in Fig. 2(c) and (d) for NEM and NEMV, respectively. We can see that NEM yields a clustering with even stronger spatial continuity than that of NEMV, which may not be appropriate for such a mixed dataset with many borders sites on the class boundary.

5 Conclusion

Compared to EM, the incorporation of a weighted spatial penalty term into the criterion function makes NEM need multiple iterations in each E-step. Besides, it is difficult to determine the spatial coefficient, on which the clustering results depend heavily. This paper presented a variant of NEM algorithm using Variable coefficients (NEMV). The site-sensitive coefficients are determined by the local Moran's I measure using correlation of explanatory attributes inside the neighborhood. Empirical results on real data sets indicated that it not only led to better results than NEM, but also converged faster with only one iteration needed in the E-step. For future work, we plan to investigate online or stochastic versions of EM to reduce dependence on initialization. Other optimization techniques, such as genetic algorithms [12], are worth trying to improve convergence rate and final clustering quality. Finally, theoretical analysis and justification are also needed for NEMV.

References

1. Ambroise, C., Govaert, G.: Convergence of an EM-type algorithm for spatial clustering. Pattern Recognition Letters 19(10), 919–927 (1998)
2. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of Royal Statistical Society B39, 1–38 (1977)
3. Guo, D., Peuquet, D., Gahegan, M.: Opening the black box: Interactive hierarchical clustering for multivariate spatial patterns. In: Proceedings of the 10th ACM International Symposium on Advances in Geographic Information Systems, pp. 131–136 (2002)
4. Legendre, P.: Constrained clustering. In: Legendre, P., Legendre, L. (eds.) Developments in Numerical Ecology, NATO ASI Series G 14, pp. 289–307 (1987)
5. Rasson, J.P., Granville, V.: Multivariate discriminant analysis and maximum penalized likelihood density estimation. Journal of the Royal Statistical Society B57, 501–517 (1995)
6. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence 6, 721–741 (1984)
7. Solberg, A.H., Taxt, T., Jain, A.K.: A markov random field model for classification of multisource satellite imagery. IEEE Transactions on Geoscience and Remote Sensing 34(1), 100–113 (1996)
8. Pena, J.M., Lozano, J.A., Larranaga, P.: An improved Bayesian structural EM algorithm for learning Bayesian networks for clustering. Pattern Recognition Letters 21(8), 779–786 (2000)
9. Basu, S., Bilenko, M., Mooney, R.J.: A probabilistic framework for semi-supervised clustering. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 59–68 (2004)
10. Hathaway, R.J.: Another interpretation of the EM algorithm for mixture distributions. Statistics and Probability Letters 4, 53–56 (1986)
11. Shekhar, S., Chawla, S.: Spatial Databases: A Tour. Prentice-Hall, Englewood Cliffs (2002)
12. Pernkopf, F., Bouchaffra, D.: Genetic-based EM algorithm for learning gaussian mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8), 1344–1348 (2005)