MIXED DATA CLUSTERING USING DYNAMIC GROWING HIERARCHICAL SELF-ORGANIZING MAP WITH IMPROVED LM LEARNING

Dr. D. Hari Prasad
Professor, Department of Computer Applications,
SNR Sons College, Coimbatore

Abstract

A primary drawback of the traditional SOM is that the size of the mapis fixed and the number of neurons in the map should be determined a priori. This might not be feasible for some applications and result in a significant limitation on the final mapping. Several dynamic SOM models have been proposed recently to reduce the limitations of the fixed network architecture of the traditional SOM. These models rely on an adaptive architecture where neurons and connections are inserted into or removed from the map during their learning process according to the particular requirements of the input data. A major variation of traditional SOM model is proposed in this paper.

1. INTRODUCTION

GHSOM can directly handle numeric, categorical and mixed data. EAOI can be used to investigate major values and it resolves the problems with discretizing numeric attributes. Although the structure of GHSOM is static and it does not have potential to manage the growth of the map. The quantization error becomes major criteria for preceding the training process. Therefore, Dynamic Growing Hierarchical Self-Organizing Map (DGHSOM) is integrated with MLP which can be trained using MLM learning to overcome the existing limitations.

It is essential for all the knowledge discovery applications to have definite control on the growth of the map. This can be accomplished by controlling the control parameter r_1 (Alahakoon et al 2000). The requirement for a measure for controlling the growth of the GHSOM is very important.

In case of using feature maps to recognize the clusters, it is helpful if there is a way to initially observe the most significant clusters and this will assist the data analyst to obtain some idea of the whole dataset, to get finer clusters. In addition, this will also support the data analyst in building decisions on regions of the data that are not of attention. And tunes the finer cluster only to the regions of interest. In order to accomplish this control, a process is developed to point out the amount of spread required by identifying a control parameter (Alahakoonetal., 2000). Definite control on the growth of the map is essential for all the knowledge discovery applications. The requirement of a measure for controlling the growth of the GHSOM is very important. Hence in this research work, an integrated framework of Dynamic Growing Hierarchical Self-Organizing Map and MLP with Modified LM algorithm is proposed for mixed data clustering.

In this paper explains about the proposed Dynamic Growing Hierarchical Self-Organizing Map which is further improved from GHSOM dynamically. And discusses about the MLP algorithm with Modified LM algorithm proposed for mixed data clustering. Then the integration of clustering and pattern extraction is also explained.

2. DYNAMIC GROWING HIERARCHICAL SELF-ORGANIZING MAP

One of the shortcomings of the SOM lies with its fixed architecture that has to be defined apriori. Dynamically growing variants of the SOM, on the other hand, tend to produce huge maps that are hard to handle. This lead to the development of the GHSOM, a new architecture which grows in a hierarchical way according to the data distribution, allowing a hierarchical decomposition and navigation in sub-parts of the data. And in a horizontal way, meaning that the size of each individual map adapts itself to the requirements of the input space.

3. DGHSOM-MLP WITH MODIFIED LM LEARNING TECHNIQUE

Modified LM approach is used for training MLP. Modified LM approach uses a technique which modifies the learning parameter that resulted in decrease of both learning iteration and oscillation (Amir et al, 2005). A modification technique called Modified LM by altering the learning parameter has been used to accelerate LM algorithm. In addition, the error oscillation has been considerably reduced.

The network's architecture is clearly shown in Figure 1 which has eight layers. The first layer with p nodes scales the data and it is the scaling interface between user and the system at the input side. The second and third layers comprise the DGHSOM layer. The output of the scaling layer is fed as the input to the DGHSOM layer. So, the second layer has p nodes. As discussed earlier for the DGHSOM net, there are complete connections between layers 2 and 3.

The output layer of the DGHSOM net possesses K number of nodes. So, there are K MLP networks, each of which receives inputs. As a result, the fourth layer has K_p nodes. These K_p nodes comprise the input layer of a set of K MLP networks. Without any loss of generalization, it is presumed that each of the K MLP networks contains only one hidden layer, even though it could be more than one and it can change for different MLP nets. The nodes in layer four is numbered as $N_i, i = 1, 2, \dots, K_p$. Nodes N_1toN_p will be the input nodes of the first MLP (M2); nodes $Np + 1$ to will be input nodes of the second MLP (M2); Similarly, nodes $N\ (K - 1)\ p + 1$ to N_{np} will be the input nodes of K^th MLP (MK). $p = 9k$as mentioned earlier. The j^th input node of MLP $M_i$gets the j^th normalized input (say, x_j) and passes it on to the first hidden layer of M_i

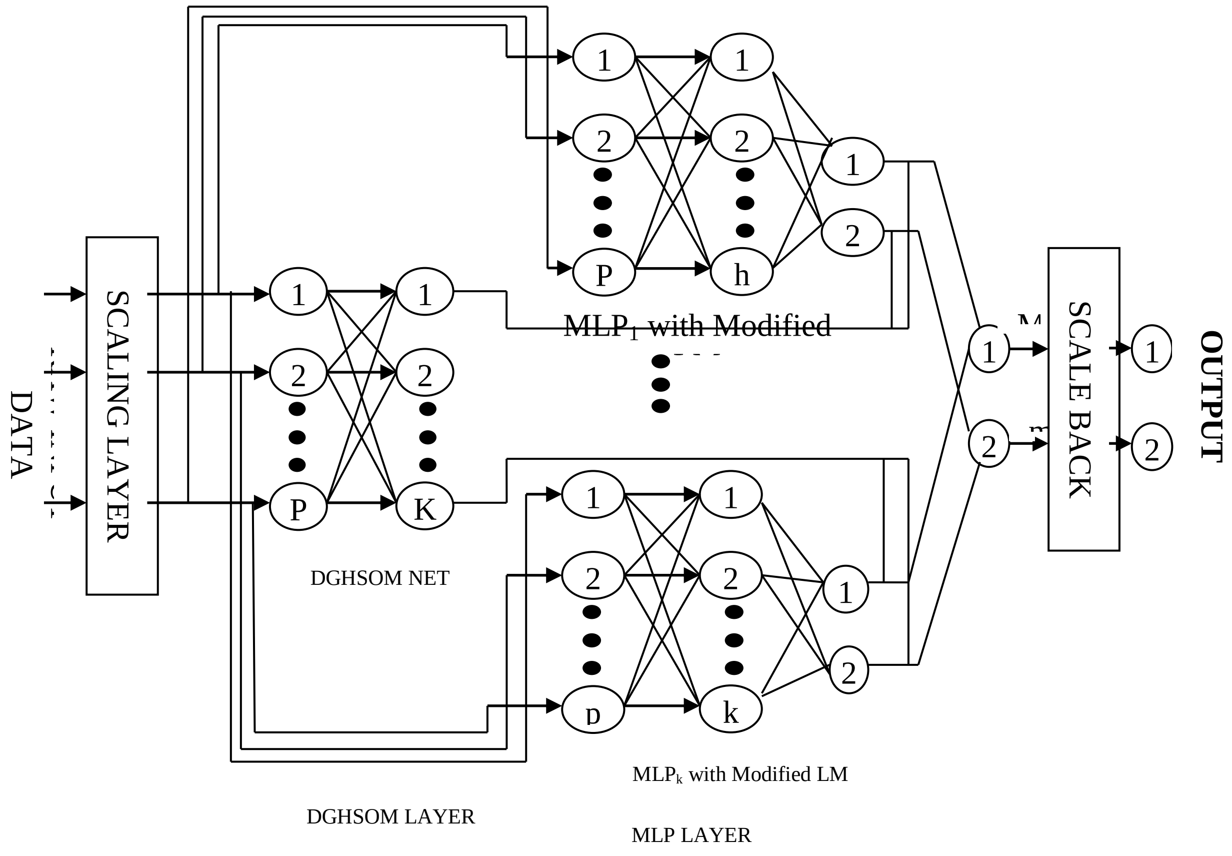

Figure 1 Architecture of DGHSOM-MLP with Modified LM for Mixed Data Clustering

The output of the node of the DGHSOM (say, O_i is linked to the output of every node of the last layer ofM_i. The product of the MLP output and the DGHSOM output then moves to layer 7. The product can be computed using an additional layer with two neurons for each MLP. Since only one of the DGHSOM output value will be one, and the remaining value will be zero, only one of the MLPs will pass its output unattenuated to layer 7. The left behind (k-1) MLPs will transmit zero to layer 7. Because it is presumed that only one hidden layer, the nodes in layer six are the output nodes of the MLP nets. Each MLP, Mi will have two output nodes. These nodes are represented by $\mathrm{O}_{\mathrm{ij}}^{6}$where the index corresponds to the MLP, Mi and j=1, 2, Layers 4–6 together constitute the MLP layer in figure 2.

The outputs of this MLP layer are then aggregated in layer seven which has just two nodes. These two nodes are represented as m and M. Now nodes O_{i1}^{6}, $\forall i = 1,2, \dots K$ are connected to node m and O_{i2}^{6}, $\forall i = 1,2, \dots K$ are connected to node M. All connection weights between layers 6 and 7 are fixed to unity and nodes m and M compute the weighted sum of all inputs as the output which is then passed to the scaling layer. It is observed that the network architecture guarantees that the aggregated output that is provided to the scaling layer is the output of the MLP equivalent to the winning node of the DGHSOM net.

The main reason for using DGHSOM is that the prototypes formed by DGHSOM preserve both topology and density. This density preservation attribute must be used. As the density matching property of DGHSOM, if a specific region of the input space contains often occurring stimuli, it will be denoted by a larger area in the feature map than a region of the input space where the stimuli occur rarely. As a result, if there is a dense area in the input space, more prototypes will be positioned there by DGHSOM. So, there will be more competitive MLPs for dense regions. Thus, finer aspects of the process can be modeled efficiently and this results in the overall enhancement of the performance.

3.1 Training Phase of the Hybrid DGHSOM-MLP

Input normalization (i.e., scaling) layer is used to normalizeX_{tr}. Then the GHSOM is trained with the normalizedX_{tr}. Once the GHSOM training is over$X_{tr}$is partitioned into K subsets,$X_{tr}^{(l)}, l = 1, 2, \dots, K$ as follows

$$X_{tr}^{(l)} = \{x_i \in R^p \,|\, ||x_i - v_l|| = \min_j ||x_i - v_j||\}$$

$X_{tr}^{(l)}$can also be said as the set of input vectors for which the lth prototype, of the GHSOM becomes the winner. Let $Y_{tr}^{(l)}$be the set of output vectors associated with vectors in $X_{tr}^{(l)}$. Now multilayer perceptron nets M_1, M_2, M_K, is trained with Modified LM algorithm.

3.2 Modified LM Algorithm

A Modified LM algorithm is used for training the neural network. Considering performance index is $F(w) = e^T e$using the Newton method the equation obtained is as

$$\mathrm{W_{K+1} = W_K - A_K^{-1}.g_K}$$

$$\mathrm{A_k = \nabla^2 F(w)|_{w=w_k}}$$

$$\mathrm{g_k = \nabla F(w)|_{w=w_k}}$$

$$\mathrm{[\nabla F(w)]_j = \frac{\partial F(w)}{\partial w_j} = 2\sum_{i=1}^{N} e_i(w).\frac{\partial e_i(w)}{\partial w_j}}$$

The gradient can be written as

$$\nabla \mathrm{F}(\mathrm{x}) = 2\mathrm{J}^{\mathrm{T}}\mathrm{e}(\mathrm{w})$$

where

$$\mathrm{J}(\mathrm{w}) = \begin{bmatrix} \frac{\partial \mathrm{e}_{11}}{\partial \mathrm{w}_1} & \frac{\partial \mathrm{e}_{11}}{\partial \mathrm{w}_2} & \cdots & \frac{\partial \mathrm{e}_{11}}{\partial \mathrm{w}_N} \\ \frac{\partial \mathrm{e}_{21}}{\partial \mathrm{w}_1} & \frac{\partial \mathrm{e}_{21}}{\partial \mathrm{w}_2} & \cdots & \frac{\partial \mathrm{e}_{21}}{\partial \mathrm{w}_N} \\ & \vdots & & \\ \frac{\partial \mathrm{e}_{KP}}{\partial \mathrm{w}_1} & \frac{\partial \mathrm{e}_{KP}}{\partial \mathrm{w}_2} & \cdots & \frac{\partial \mathrm{e}_{KP}}{\partial \mathrm{w}_N} \end{bmatrix}$$

$J(w)$is called the Jacobian matrix.

Then, the Hessian matrix is to be found. The k, j elements of the Hessian matrix yields as

$$[\nabla^2 F(w)]_{k,j} = \frac{\partial^2 F(w)}{\partial w_k \partial w_j} = 2\sum_{i=1}^{N} \left\{ \frac{\partial e_i(w)}{\partial w_k}\frac{\partial e_i(w)}{\partial w_j} + e_i(w).\frac{\partial^2 e_i(w)}{\partial w_k \partial w_j} \right\}$$

The Hessian matrix can then be represented as follows

$$\nabla^2 F(w) = 2J^T(W).J(W) + S(W)$$

If $S(w)$ is small assumed, the Hessian matrix can be approximated as

$$\nabla^2 F(w) \cong 2J^T(w)J(w)$$

Using A_k and $S(w)$, the Gauss-Newton method is obtained as follows

$$W_{k+1} = W_k - [2J^T(w_k) \cdot J(w_k)]^{-1} 2J^T(w_k)e(w_k)$$
$$\cong W_k - [J^T(w_k) \cdot J(w_k)]^{-1} J^T(w_k)e(w_k)$$

The advantage of Gauss-Newton is that it does not need computation of second derivatives. The problem in the Gauss-Newton method is the matrix $H = J^T J$ may not be invertible. This can be overcome by using the following modification.

Hessian matrix can be written as

$$\mathrm{G} = \mathrm{H} + \mu \mathrm{I}$$

Suppose that the Eigen values and eigenvectors of H are $\{\lambda_1, \lambda_2, \dots \dots, \lambda_n\}$ and $\{z_1, z_2, \dots \dots, z_n\}$.
ThenGz_{i} is represented as

$$\mathrm{Gz}_{\mathrm{i}} = [\mathrm{H} + \mu \mathrm{I}]\mathrm{z}_{\mathrm{i}} = \mathrm{Hz}_{\mathrm{i}} + \mu \mathrm{z}_{\mathrm{i}} = \lambda_{\mathrm{i}}\mathrm{z}_{\mathrm{i}} + \mu \mathrm{z}_{\mathrm{i}} = (\lambda_{\mathrm{i}} + \mu)\mathrm{z}_{\mathrm{i}}$$

As a result, the eigenvectors of G are the identical as the eigenvectors of H and the Eigen values of G are $(\lambda_i + \mu)$. The matrix G is positive definite by enhancing μ until $(\lambda_i + \mu) > 0$ for all i therefore the matrix will be invertible.

This leads to LM algorithm as

$$\mathrm{w}_{\mathrm{k}+1} = \mathrm{w}_{\mathrm{k}} - [\mathrm{J}^{\mathrm{T}}(\mathrm{w}_{\mathrm{k}})\mathrm{J}(\mathrm{w}_{\mathrm{k}}) + \mu \mathrm{I}]^{-1}\mathrm{J}^{\mathrm{T}}(\mathrm{w}_{\mathrm{k}})\mathrm{e}(\mathrm{w}_{\mathrm{k}})$$

$$\Delta \mathrm{w}_{\mathrm{k}} = [\mathrm{J}^{\mathrm{T}}(\mathrm{w}_{\mathrm{k}})\mathrm{J}(\mathrm{w}_{\mathrm{k}}) + \mu \mathrm{I}]^{-1}\mathrm{J}^{\mathrm{T}}(\mathrm{w}_{\mathrm{k}})\mathrm{e}(\mathrm{w}_{\mathrm{k}})$$

As known, learning parameter, μ is illustrator of steps of real output movement to preferred output. In the standard LM method, μ is a steady number. This work modifies LM method using μ as

$$\mu = 0.01e^{T}e$$

where$e$ is a $k \times 1$ matrix therefore $e^{T}e$is a 1×1 therefore $[J^{T}J + \mu I]$is invertible.

Consequently, if actual output is far than the preferred output or similarly, errors are huge so, it converges to preferred output with large steps. Similarly, when the measurement of error is small then, the actual output approaches to the preferred output with soft steps. Therefore, error oscillation reduces significantly.

The GHSOM-MLP with Modified LM Algorithm is outlined as follows

Step 1: Raw input data is given as input to the scaling layer of the hybrid DGHSOM _MLP networks (First Layer).

Step 2: The scaled data is given as input to the DGHSOM layer (Second and Third Layer).

Step 3: MLP layer has K nodes. MLP networks receive p inputs (Fourth Layer).

Step 4: MLP is trained using Modified Levenberg-Marquardt algorithm

Step 5:K_p nodes constitutes the input layer of a set of K MLP networks.

Step 6: Nodes in fourth layer is numbered as $N_{i.}\ i = 1,2, \ldots\ldots K_p$.

Step 7: Nodes N_1 *to* N_p is the input of the first MLP M_1.

Step 8: Each MLP, $M_i$has two output nodes *m* and M.

Step 9: The outputs of this MLP-layer are then aggregated in layer seven which has just two nodes one for the minimum and the other for the maximum object.

Modified LM algorithm for learning is used for learning of the DGHSOM-MLP which provides significant performance in clustering mixed data. The proposed Modified LM algorithm converges to desired output and moreover error oscillation also reduces greatly. Thus, an effective neural network approach is used for clustering mixed data using Modified LM Algorithm.

4. EXPLORATORY CLUSTERING AND PATTERN EXTRACTION

By blending the DGHSOM and the EAOI can result in a perfect and efficient tool for mixed data. So it is planned to have an interactive, visualized analysis framework for data clustering and pattern extraction. This new medley of tool will provide better analysis potential, as compared to stand alone techniques. It is construed DGHSOM as such is insufficient in extracting clusters' characteristics, but EAOI can bring out over generalization if the data are diversified and not clustered before generalization. The benefits of the blending are visualization and exploration, making it an excellent tool in exploratory data mining. The visualized SOM helps in making the issue less difficult problem of determining the suitable cluster number. Besides, the DGHSOM and the EAOI allow users to explore the data from various angles in addition to their conventional analyses. Especially the DGHSOM can handle categorical data straightaway, and EAOI can inquire for major values and provide an equivalent for obtaining numeric attributes. Once this is over the data are projected into a two dimensional DGHSOM, and then the visualized data clustering is done automatically or semi automatically on the trained DGHSOM, and at the end the EAOI is used to extract the characteristics from individual clusters. The following type of clusters can be analyzed viz cluster characteristics, Characteristic rules and discriminant rules as in Chung & Sheng (2006).

5. EXPERIMENTAL RESULTS

In the present work, the proposed three approaches for mixed data clustering are evaluated. The performances of the algorithms have been evaluated using the various parameters. The performance of the proposed mixed data clustering approaches are evaluated using the parameters like, Number of Resultant Clusters

The number of resultant clusters for synthetic dataset using Modified SOM, GHSOM and DGHSOM with various distance criteria are compared with GSOM proposed by Chung & Sheng (2006) is provided in the Table 1. It can be seen that the projected DGHSOM with EAOI technique outcomes in better grouping than GSOM of existing one and it identifies the clusters and outliers successfully.

Table 1 Number of Resultant Clusters with Different Distance Criteria for Iris Dataset

	Generalized SOM (GSOM)		Modified SOM		GHSOM		DGHSOM	
	Cluster	Outlier	Cluster	Outlier	Cluster	Outlier	Cluster	Outlier
d=0	81	-	79	-	68	-	61	5
$d \leq 1.414$	17	-	14	-	10	-	7	2
$d \leq 2.828$	8	-	7	-	6	-	5	2
$d \leq 3\&Adj$	9	3	6	2	5	4	4	5

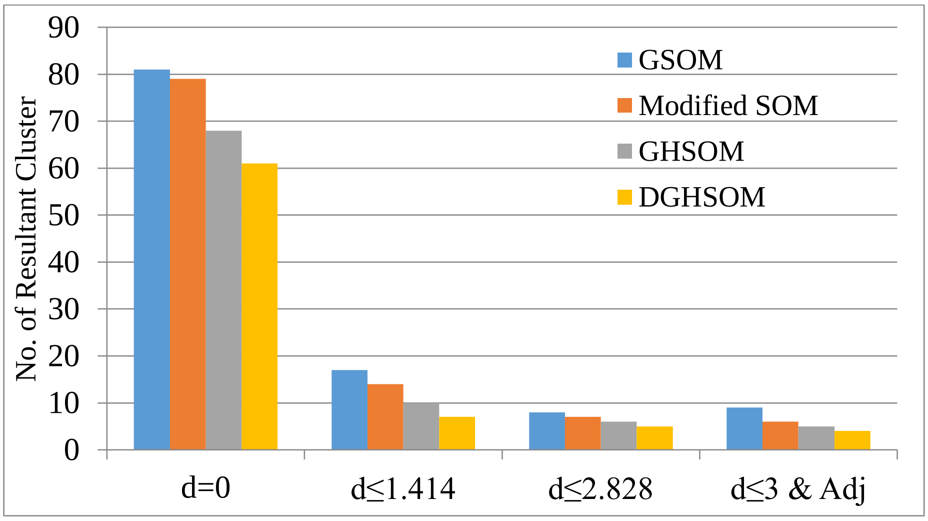

Figure 2 Number of Resultant Clusters with Different Distance Criteria for Iris Dataset

The number of resultant clusters by means of DGHSOM, GHSOM and Modified SOM and base method of Chung & Sheng (2006) with dissimilar distance criteria for cleve dataset is provided in Figure 2. Dynamic Growing Hierarchical Self-Organizing Map (DGHSOM) results in better grouping, as it evades unnecessary clusters and thus the numbers of clusters are significantly decreased.

6. CONCLUSION

This paper discuss about the proposed Dynamic Growing Hierarchical Self-Organizing Map which is modified dynamically from visualized GSOM for mixed data clustering. Multi Layer Perceptron and Modified LM algorithm is also discussed which are useful for extracting patterns from the clusters. Thus this paper provides the overall information for mixed data clustering and exploratory data mining along with proved results.

REFERENCES

1. Chung-chain Hsu 2004, 'Extending attribute–oriented induction algorithm for major values and numeric values', Expert system with applications, Vol. 27, No. 2, pp 187-202.
2. Chung-Chian Hsu& Sheng-Hsuan Wang 2004, 'Extending attribute-oriented induction algorithm for major values and numeric values'. Expert Systems with Applications, Vol. 27, No. 2, pp.187–202.
3. Chung-Chian Hsu & Sheng-Hsuan Wang 2006 'An Integrated Framework for Visualized and Exploratory Pattern Discovery in Mixed Data', IEEE Transactions on Knowledge and Data Engineering, Vol. 18, No. 2, pp. 161 - 173.
4. Chung-Chian Hsu & Yan-Ping Huang 2008, 'Incremental clustering of mixed data based on distance hierarchy', Expert Systems with Applications, Vol.35, pp.1177–1185.
5. Alahakoon, D,Halgamuge& S.K, Srinivasan, B 2000, 'Dynamic self-organizing maps with controlled growth for knowledge discovery', IEEE Transactions on Neural Networks, Vol. 11, No. 3, pp. 601–614.
6. Amir AbolfazlSuratgar, Mohammad BagherTavakoli& Abbas Hoseinabadi 2005, 'Modified Levenberg-Marquardt Method for Neural Networks Training', World Academy of Science, Engineering and Technology 6.
7. Attik, M,Bougrain, L &Alexandre, F 2005, 'Self-organizing Map Initialization', Artificial Neural Networks: Biological Inspirations: Lecture Notes in Computer Science', Volume 3696: Springer Berlin Heidelberg, pp 357-362.
8. Banfield, J.D. &Raftery, A.E 1993, 'Model-based Gaussian and non-Gaussian clustering Biometrics', Vol. 49, pp. 803-821.
9. Bernd Fischer &Buhmann, J. M 2003, 'Path-Based Clustering for Grouping of Smooth Curves and Texture Segmentation', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 4, pp. 513-518.
10. Norashikin Ahmad &DammindaAlahakoon, 2010, 'Generating Concept Trees from Dynamic Self-organizing Map', World Academy of Science, Engineering and Technology, Vol.41, pp.706-711.
11. Osmar RZaiane 1998, Principles of Knowledge Discovery in Databases-Chapter 8:DataClustering,<http://www.cs.ualberta.ca/~zaiane/courses/cmput690/slides/Chapter 8/index.html>.
12. Pavan, M. &Pelillo, M 2003, 'Dominant Sets and Hierarchical Clustering', Proceedings of IEEE International Conference Computer Vision, Vol. 1, pp. 362-369.
13. Pavan, M. &Pelillo, M 2005, 'Efficient Out-of-Sample Extension of Dominant-Set Clusters', Advances in Neural Information Processing Systems 17, L.K. Saul, Y. Weiss, and L. Bottou, eds, pp. 1057-1064.