Label Propagation for Semi-Supervised Learning in Self-Organizing Maps

Lutz Herrmann and Alfred Ultsch
Databionics Research Group
Dept. of Mathematics and Computer Science
Philipps-University Marburg
email: {lherrmann, ultsch}@mathematik.uni-marburg.de

Keywords: semi-supervised learning, label propagation, clustering

***Abstract*— Semi-supervised learning aims at discovering spatial structures in high-dimensional input spaces when insufficient background information about clusters is available. A particulary interesting approach is based on propagation of class labels through proximity graphs. The Emergent Self-Organizing Map (ESOM) itself can be seen as such a proximity graph that is suitable for label propagation. It turns out that Zhu's popular label propagation method can be regarded as a modification of the SOM's well known batch learning technique. In this paper, an approach for semi-supervised learning is presented. It is based on label propagation in trained Emergent Self-Organizing Maps. Furthermore, a simple yet powerful method for crucial parameter estimation is presented. The resulting clustering algorithm is tested on the fundamental clustering problem suite (FCPS).**

1 Introduction

In the field of machine learning, semi-supervised learning (SSL) represents a midpoint between supervised learning, in which all input samples are preclassified, and unsupervised learning, in which no class labels are given. SSL aims at incorporating a small amount of preclassified data into unsupervised learning methods in order to increase performance of data analysis. Therefore, SSL has recently become focus of interest, particularly because of application domains in which unclassified data are plentiful, such as bioinformatics and medical domains.

This work describes how SSL is accomplished by the so-called Emergent Self-Organizing Map. Basic concepts and algorithms are introduced in sections 2 and 3. In section 4 a label propagation method for the Emergent Self-Organizing Map is introduced. Section 5 describes how label propagation can be used to accomplish semi-supervised learning tasks with trained Emergent Self-Organizing Maps. Furthermore, a simple yet powerful method for parameter estimation is presented. An experiment is done in section 6 in order to evaluate the clustering performance of our proposed method. Finally, in sections 7 and 8 features and benefits of our method are discussed and summarized.

2 Basic Principles

Semi-supervised learning (SSL) means learning from both preclassified and yet unclassifed input samples. Thus, SSL methods are located between classical supervised learning techniques, in which all input samples are preclassified, and unsupervised learning techniques, where no class labels are given at all. For a comprehensive overview on SSL methods see [11].

Formally, SSL aims at construction of a classifier function from a finite set of partially classified input samples $X \subset \mathbb{R}^n$. A classifier function $c : X \to \{0, C_1, ..., C_k\}$ assigns class labels to input samples. An input sample $x \in X$ is said to be preclassified $c(x) \in \{C_1, ..., C_k\}$. If this is not the case x is said to be unclassified, i.e. $c(x) = 0$. SSL aims at generalizing c into a meaningful $c' : X \to \{C_1, ..., C_k\}$ such that subsets of *coherent* input samples with identical class labels, so-called clusters, do emerge.

Here, it has to be mentioned that SSL methods come in two flavors. Semi-supervised *classification* means that the resulting classifier function c' is limited to the class labels given in the set of preclassified input samples. In case of semi-supervised *clustering*, the resulting classifier function c' is free to add new or remove class labels, if it complies with the given spatial structure of X.

A promising approach for realization of semi-supervised learning is subsumed as *label propagation* (LP). Label propagation methods operate on proximity graphs in order to spread informations about class memberships to nearby nodes. This is based on the assumption that nearby entities should belong to the same class, whereas far away entities might belong to different classes.

Formally, a proximity graph (V, E) consists of nodes $V = \{1, ..., |X|\}$ representing the input samples of training set X and undirected edges $E \subset V \times V$ represent similarities between them. These similarities are more precisely determined by a weight matrix W whose entries $w_{ij} \in [0, 1]$ are non-zero iff x_i and x_j are neigbouring, i.e. $(i, j) \in E$. Common ways to determine W are k-nearest-neighbour [8] or gaussian kernel approaches [9]. For more details see sections 3 and 4.

Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM 2007)
Published by the Neuroinformatics Group, Bielefeld University, Germany, ISBN 978-3-00-022473-7
All contributions to WSOM 2007 are available online at: http://biecoll.ub.uni-bielefeld.de

For LP purposes, each node is assigned a so-called *label vector*. A label vector $l_i \in [0,1]^k$ contains the probabilistic memberships of input sample $x_i \in X$ to the available $k \in \mathbb{N}$ clusters. Nodes belonging to preclassified input samples $x_i \in X$ with $c(x_i) = C_q \in \{C_1, ..., C_k\}$ have fixed label vectors l_i, i.e. l_{ij} is 1 iff j equals q, otherwise it is set to zero.

In LP methods, the nodes propagate their label vectors to all adjacent nodes according to the proximity defined in W (see sections 3 and 4). Thus, nearby nodes are more likely to have similar label vectors than far away nodes. Label vectors of unclassified nodes are initialized at random and iteratively adapt to the fixed labels of preclassified nodes according to the proximity graph. Therefore, nodes of preclassified input samples are used as *seed points* for spreading known labels through the proximity graph of input samples. This is how LP methods accomplish semi-supervised learning tasks.

3 Related Work

In the following, some related works are presented in order to give an idea how semi-supervised clustering usually is accomplished.

The *semi-supervised fuzzy c-means* (ssFCM, [1]) is, to our knowledge, the first semi-supervised clustering algorithm using label vectors without a proximity graph. For each input sample $x_i \in X$, there is a label vector $l_i \in [0,1]^k$ that contains fuzzy memberships to each available cluster. The ssFCM method aims at minimizing a fuzzyfied version of the popular k-means error criterion, i.e. $\sum_{x_i \in X} \sum_{j=1}^{k} l_{ij} d^2(x_i, m_j)$ with m_j being the center of cluster j and d denoting some distance measure, e.g. the euclidean metric. Cluster centers and label vectors are iteratively updated according to the Expectation-Maximization principle. For preclassified input samples x_i with $c(x_i) = C_q$, the label vector is kept constant, i.e. l_i being 1 in position q and 0 elsewhere. For unclassified input samples the label vectors are iteratively updated.

A popular graph-based approach on semi-supervised learning was published by Zhu et al. [9] [10]. A fully connected graph is used for label propagation. Each node corresponds to an input sample of training set $X \subset \mathbb{R}^n$. The edge weights w_{ij} between input samples $x_i, x_j \in X$ are determined by a gaussian kernel function that depends on the distance $d(x_i, x_j)$ and a radius $\sigma \in \mathbb{R}^+$. From that a matrix $T \in \mathbb{R}^{|X| \times |X|}$ is derived that contains the transition probabilities between graph nodes, i.e. $T_{ij} = \frac{w_{ij}}{\sum_r w_{rj}}$. According to T, the class informations of preclassified samples are propagated through the set of input samples. For each input sample x_i, there is a label vector l_i that is fixed for preclassified input samples and iteratively altered for unclassified input samples, i.e. $L' = TL$ with $L = (l_i)_{x_i \in X} \in \mathbb{R}^{|X| \times k}$ being the matrix of label vectors. Obviously, the class boundaries are pushed through high density regions and settle in low density gaps. Proof of convergence has been given in [9], but results are highly dependent on parametrization of the gaussian kernel.

Another popular graph based semi-supervised learning method was published by Belkin and Niyogi [2]. The main idea is that a few preclassified input samples may not be enough to confidently classify the remaining unclassified input samples. Therefore, learning of underlying *manifolds* (in the sense of Riemannian manifolds) is done by the help of all available input samples. As a model for such a manifold serves a proximity graph whose nodes are the input samples from $X \subset \mathbb{R}^n$. Nodes i, j are connected with an edge iff they are adjacent, i.e. the underlying input samples x_i, x_j are sufficiently close. Edge weights $W = (w_{ij})_{x_i, x_j \in X}$ are derived from the distances between the corresponding input samples. The so-called Laplacian $L(W)$ is a symmetric, positive semidefinite matrix which can be thought of as an operator on functions defined on nodes of the proximity graph or on matrix W, respectively. The eigenfunctions of the Laplacian provide a natural basis for functions on the manifold and the desired classifier function $c : X \to \{C_1, ..., C_k\}$ can be expressed in such a basis.

4 Label propagation in Emergent Self-Organizing Maps

As seen in section 3, class information in the form of label vectors propagate through proximity graphs. In this section, it is shown how label propagation can be realized on top of an already trained SOM.

A Self-Organizing Map (SOM) consists of a finite set I of neurons. To simplify matters, each neuron $i \in I$ is a joint index for a codebook vector $m_i \in \mathbb{R}^n$ and a fixed position $p_i \in \mathbb{N}^2$ on a regular grid. The SOM's training algorithm iteratively modifies the codebook vectors such that they approximate distance and density structure of training set $X \subset \mathbb{R}^n$, and sufficiently retain their input space topology on the low-dimensional grid.

There are two types of Self-Organizing Maps that can be distinguished [6]: first, SOM in which each neuron represents a cluster of input samples. These maps can be thought of as a variant of the k-means clustering algorithm. In these SOM, the number of neurons somehow corresponds to the number of clusters assumed in the input samples. In contrast to that, SOM may be used as tools for visualization of structural features of the data space using U-Matrix and P-Matrix techniques [6]. A characteristic of this paradigm is the large number of neurons, usually several thousands. These SOM allow the emergence of intrinsic structural features of the data space on the map. Therefore, they are

called Emergent Self-Organizing Maps (ESOM). For details on SOM and ESOM see [3] [6] [7]. In the following, we will concentrate on ESOM.

In this paper, the method of Zhu [9] [10] is adapted as follows. For each neuron $i \in I$, there is a label vector $l_i \in [0,1]^k$. The label vectors are seen as nodes in a proximity graph whose edges' weights are derived from the pairwise distances of the ESOM's neurons.

The measure for inter neuron distances is the so-called u-distance [7]. The u-distance $udist : I \times I \rightarrow \mathbb{R}_0^+$ is defined as the minimal path length along grid-neighbouring neurons (see equation 1). The u-distance is sensitive to the neurons' distances and incorporates the grid structure as displayed by the U-Matrix.

$$udist(i,j) = \min_{(i=i_1,\ldots,i_p=j)} \sum_{k=1}^{p-1} d(m_{i_k}, m_{i_{k+1}}) \quad (1)$$

The edge weight between nodes i, j results as $w_{ij} = \exp(-\frac{udist^2(i,j)}{\sigma^2})$. The parameter σ acts as a radius that defines how far label information may spread in the graph. Radius σ is ideally chosen as the minimal inter-cluster distance. For estimation of σ in Emergent Self-Organizing Maps see section 5.

Propagation of class labels through the proximity graph is realized as follows: Nodes belonging to bestmatching neurons of preclassified input samples $x_i \in X$ with $c(x_i) = C_q \in \{C_1, ..., C_k\}$ have fixed label vectors l_i, i.e. l_{ij} is 1 iff j equals q, otherwise it is set to zero. Label vectors of unclassified neurons j are updated according to equation 2 until the algorithm converges, i.e. no more changes in $(l_i)_{i \in I}$ occur.

$$l'_j = \frac{\sum_{i \in I} w_{ij} \cdot l_i}{\sum_{i \in I} w_{ij}} \quad (2)$$

This update mechanism strongly resembles the batch-map learning rule [3] of the SOM whereas the edge weights w_{ij} act as fixed neighbourhood function values and the label vectors act as input samples. Obviously, Zhu's label propagation [9] and our proposed method are both analogies of the batch-map technique.

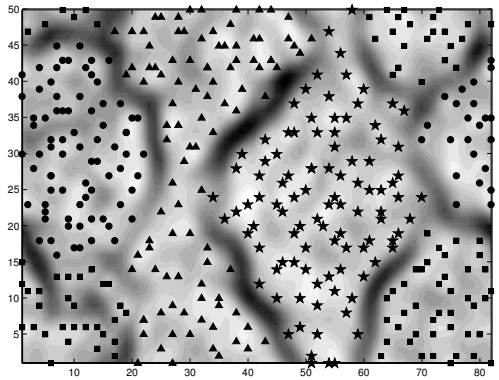
5 Semi-supervised Clustering in Emergent Self-Organizing Maps

As seen in section 4, our method produces an additional label vector $l_i \in [0,1]^k$ for each neuron $i \in I$. In this section, label vectors are used for realization of semi-supervised learning tasks in terms of clustering. Furthermore, a method on how to determine the crucial parameter σ is proposed for Emergent Self-Organizing Maps.

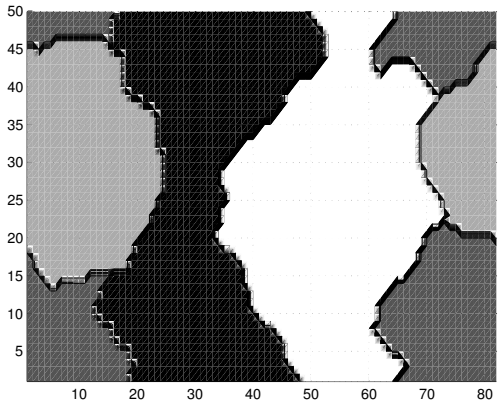
A simple to realize application of ESOM-derived label vectors is automatic clustering of unclassified input samples. Let $x \in X$ be an unclassified input sample that is assigned to neuron $bm(x) = \arg\min_{i \in I} d(x, m_i)$. In analogy to other classifiers dealing with label vectors (see section 3) x is assigned the class label with the highest value on its label vector, i.e. $c(x) = \arg\max_{C_j \in \{C_1,\ldots,C_k\}} l_{bm(x),j}$. For an example see figure 1(b).

A crucial problem [9], however, is defining a suitable propagation parameter $\sigma \in \mathbb{R}^+$. When $\sigma \rightarrow 0$ the label propagation method performs approximately like a k-nearest-neighbour classifier with $k = 1$. When $\sigma \rightarrow \infty$ the set of label vectors effectively shrinks to a single point, i.e. all unclassified input samples receive the same label vector. Obviously, the appropriate σ is in between.

For estimation of σ we propose a simple method that was loosely inspired by cost-benefit analysis and is based on distribution analysis of U-Matrix heights [5]. U-Matrix



(a) U-Matrix



(b) assignment of class labels

Figure 1: "Tetra" data set [4] from the FCPS: **(a)** toroidal U-Matrix with classified bestmatches, blurred cluster boundaries **(b)** proposed lpSM method: neurons classified according to label vectors, sharp decision boundaries obtained from a single preclassified input sample per class

Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM 2007)
Published by the Neuroinformatics Group, Bielefeld University, Germany, ISBN 978-3-00-022473-7
All contributions to WSOM 2007 are available online at: http://biecoll.ub.uni-bielefeld.de

heights denote local averages of inter neuron distances. U-Matrix heights will be low on inner cluster neurons and high on inter cluster neurons. Therefore, U-Matrix heights are useful for detection of cluster borders and, in our case, especially suitable for estimation of inter cluster distances, such as the desired σ.

The radius σ should be as small as possible in order not to spread the class information equally over the ESOM. On the other hand, the coverage of σ dependent kernels should be as big as possible in order to cover all neurons of the corresponding cluster. An easy to obtain indicator for coverage is the empirical cumulative distribution function $ecdf(\sigma) \in [0, 1]$ of U-Matrix heights. From the definition, $ecdf(\sigma)$ indicates the fraction of neurons that have smaller distances towards their neighbours than σ. Thus, $ecdf(\sigma)$ denotes the fraction of neurons that are covered by σ when spreading information towards immediately neighbouring neurons. We choose σ as the U-Matrix height value that minimizes the unrealized potential $urp(\sigma)$, i.e. $\sigma_{opt} = \arg\min_{\sigma \in [\sigma_{min}, \sigma_{max}]} urp(\sigma)$ with σ_{min} (σ_{max}) being the minimal (maximal) height of the U-Matrix. The unrealized potential of σ is defined as the euclidean distance between $(\sigma, ecdf(\sigma))$ and the hypothetically optimal point $(\sigma_{min}, 100\%)$ of minimal cost and maximal benefit as seen in equation 3 and figure 2. Obviously, inter cluster neurons lead to a saturation of the coverage curve and, therefore, allow a meaningful estimation of σ. The basic idea behind urp was introduced in [5] and has been applied in several domains. For illustration of convergence with different values of σ see figure 3.

$$urp(\sigma) = \sqrt{\left(\frac{\sigma - \sigma_{min}}{\sigma_{max} - \sigma_{min}}\right)^2 + (1 - ecdf(\sigma))^2} \quad (3)$$

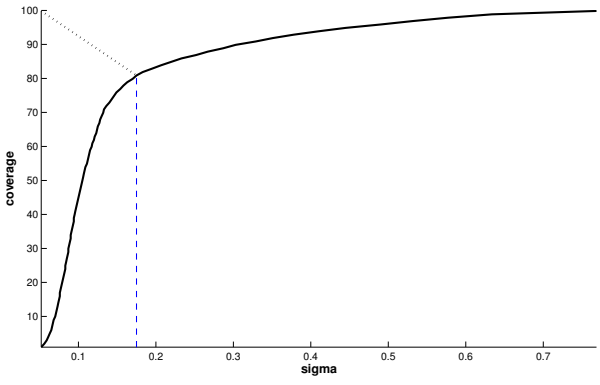
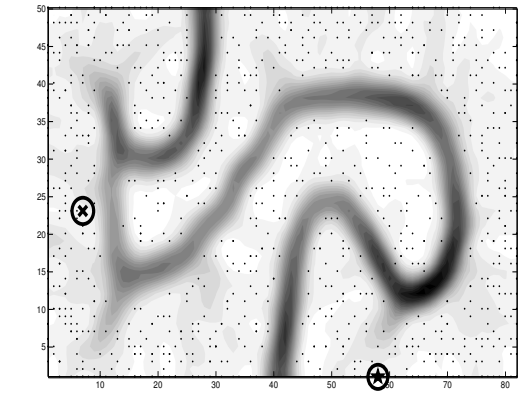
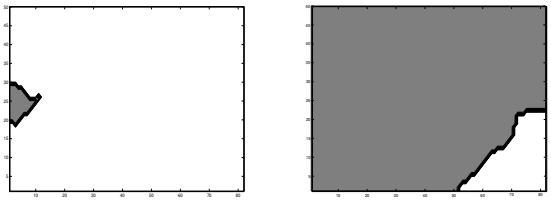


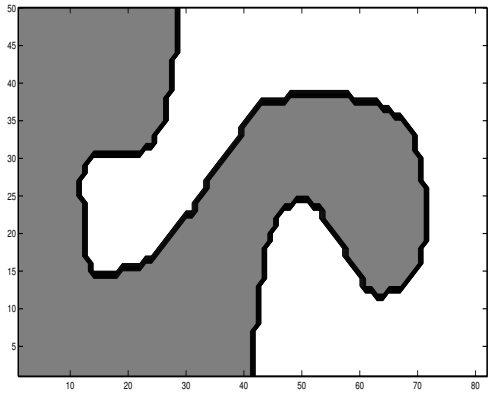
Figure 2: "Chainlink" data set [4] from the FCPS: U-Matrix heights' distribution for estimation of propagation radius σ by minimization of unrealized potential, percental coverage against radius σ (solid line), minimal unrealized potential (dotted line) and σ_{opt} (dashed line)



(a) U-Matrix with partially preclassified bestmatches



(b) decision boundaries with $\sigma \to 0$ and $\sigma \to \infty$



(c) decision boundaries with estimated σ_{opt}

Figure 3: "Chainlink" data [4] of FCPS **(a)** planar U-Matrix with two well separated clusters, many unclassified (black dots) and two preclassified (encircled) bestmatches **(b)** classified neurons: decision boundaries obtained from our label propagation method *lpSM* with too small or big σ **(c)** classified neurons: decision boundaries obtained from our label propagation method *lpSM* with σ correctly estimated from U-Matrix heights' distribution, classes correspond to visible clusters on the U-Matrix

6 Experimental Settings and Results

In order to evaluate the automatic clustering abilities of our so-called *label propagation for Self-organizing Maps*

Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM 2007)
Published by the Neuroinformatics Group, Bielefeld University, Germany, ISBN 978-3-00-022473-7
All contributions to WSOM 2007 are available online at: http://biecoll.ub.uni-bielefeld.de

(lpSM) method, its clustering performance was measured. The main idea is to use data sets on which the input samples' true classification is known in beforehand. Clustering accuracy can be evaluated as fraction of correctly classified input samples.

The lpSM method is tested against the manifold learning method of Belkin and Niyogi [2] which happens to be one of the most elaborated techniques from the field of semi-supervised learning. For each data set, both algorithms only got *a single* randomly chosen input sample per class with the true classification. The remaining samples are presented as unlabeled data.

The data comes from the fundamental clustering problem suite (FCPS). This is a collection of data sets for testing clustering algorithms. Each data set represents a certain problem that any clustering algorithm should be able to handle. For details see [4].

The Emergent Self-Organizing Maps were produced by the online learning algorithm [3] using cone shaped learning environments during 30 epochs. The size of the grid is 50×82 nodes. The learning rate is kept constant at 0.1.

Comparative results can be seen in table 1. The lpSM method clearly outperforms Belkin's manifold learning algorithm that supposedly suffers from its inability to recognize density defined cluster shapes, e.g. the "EngyTime" and "Wingnut" data.

7 Discussion

In this work, we described a novel approach (lpSM) to semi-supervised cluster analysis using class label propagation on trained Emergent Self-Organizing Maps. To our knowledge, this is the first approach that aims for the realization of semi-supervised learning paradigms on basis of unsupervised ESOM with label propagation techniques.

The lpSM method and Zhu's label propagation method obviously differ in two ways. First, our proposed distance measure *u-distance* is ESOM specific and, therefore, takes the visible grid structure into account. Finally, Zhu estimates the radius σ according to the longest edge length of a minimum spanning tree over all input samples. For available outliers, this leads to an overestimation of inter cluster distances and, therefore, poor classifications are the results. In contrast to that, the σ estimation procedure of lpSM relies on a distances' distribution based criterion that is supposedly robust against outliers.

data set	Belkin and Niyogi	lpSM
Atom	93	100
Chainlink	78	100
Hepta	100	100
Lsun	83	100
Target	53	87
Tetra	100	100
TwoDiamonds	75	95
Wingnut	70	90
EngyTime	57	96

Table 1: averaged percental clustering accuracy on FCPS data sets [4] over hundred runs, lpSM method outperforms laplacian manifold learning technique of Belkin and Niyogi [2]

Here, it has to be mentioned that any label propagation method following the scheme depicted in section 3 relies on the existance of a meaningful inter-cluster distance. If this is not the case, e.g. spatial clusters with different levels of scaling, the resulting classification may be misleading.

In contrast to most other clustering algorithms, the lpSM system has no inherent assumption about cluster shapes because it relies on structures learned by the Emergent Self-Organizing Map that is known for that [6]. The lpSM method picks up the distance *and* density structure of the ESOM's neurons in order to reflect it faithfully in the space of label vectors. This is realized using a modified batch-map update rule (see section 4). The lpSM is even superior to implicit manifold learning techniques, as demonstrated in section 6.

The proposed update-mechanism depends on a single parameter σ which acts as a radius for determination of how far class labels are propagated through the set of label vectors. For fixed σ the lpSM method converges which was shown in [9] for a comparable approach. Choosing of σ is crucial because results are highly dependent on a good estimation of an inter-cluster radius that effectively decreases propagation in low density space in between clusters. A practical estimation of σ can be derived from the distribution of U-Matrix heights.

8 Summary

In this paper, a novel method for *semi-supervised* cluster analysis is presented, which means that input samples of a given training set are partially classified. Topological information of a trained Emergent Self-Organizing Map (ESOM) is used in order to derive probabilistic class labels for the input samples. For that purpose, each neuron is enhanced with a vector of class probabilities. Class labels are propagated from bestmatching neurons of preclassified input samples to other neurons according to the underlying proximity structure of the ESOM. Such labels are useful for visualization and classification purposes. An estimation procedure for crucial parametrization was derived. It turned out that the distribution function of the U-Matrix visualization technique is a suitable indicator for the propagation radius.

Furthermore, it was shown that our proposed method *label propagation in Self-Organizing Maps* (lpSM) outperforms one of the most popular algorithms for semi-

supervised manifold learning on a set of fundamental clustering problems.

9 Outlook and Future Work

As outlined by Zhu and Ghahramani [9] [10], estimation of propagation radius σ is crucial for convergence of iterative label propagation into a meaningful fixpoint. Instead of choosing a global radius for every pair of entities, σ might be chosen sensitive to the local distance and density structure of neurons. Localization of σ promises to be more sensitive to variances in density.

As seen in section 4, label propagation is a special case of the batch-map's training method. Therefore, focus of our research will be incorporation of label propagation techniques into an online learning ESOM.

Another interesting aspect of unsupervised learning is its connection to semi-supervised learning. Unsupervised segmentation of a given ESOM into cluster representing subsets of neurons can easily be accomplished by the semi-supervised lpSM method if seed points are available for each class. Such neurons are usually derived from density estimations, e.g. the P-Matrix [6].

References

[1] A. Bensaid, L. Hall, J. Besdeck, L. Clarke, "Partially Supervised Clustering for Image Segmentation", *Pattern Recognition* Vol. 29 (5), Elsevier, pp. 859-871, 1996.

[2] M. Belkin, P. Niyogi, "Using Manifold Structure for Partially Labeled Classification", *Advances in Neural Information Processing Systems (NIPS)*, Vol. 15, MIT Press, 2003.

[3] T. Kohonen, "Self-Organizing Maps", Springer, 1995, 1997, 2001.

[4] A. Ultsch, "Fundamental Clustering Problem Suite", `http://www.mathematik.uni-marburg.de/~databionics`.

[5] A. Ultsch, "Proof of Pareto's 80/20 Law and Precise Limits for ABC-Analysis", *Technical Report No. 02/c*, Dept. of Mathematics and Computer Science, University of Marburg, Germany, 2002.

[6] A. Ultsch, "Maps for the Visualization of high-dimensional Data Spaces", *Proceedings Workshop on Self-Organizing Maps (WSOM 2003)*, Kyushu, Japan, pp. 225-230, 2003.

[7] A. Ultsch, L. Herrmann, "The architecture of emergent self-organizing maps to reduce projection errors", *Proceedings of the European Symposium on Artificial Neural Networks (ESANN 2005)*, Bruges, Belgium, pp. 1-6, 2005.

[8] F. Wang, C. Zhang, "Label Propagation through Linear Neighbourhoods", *Proc. 23rd International Conference on Machine Learning*, Pittburgh, PA, 2006.

[9] X. Zhu, Z. Ghahramani, "Learning from Labeled and Unlabeled Data with Label Propagation", Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.

[10] X. Zhu, Z. Ghahramani, J. Lafferty, "Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions", *Proceedings of the 20th International Conference on Machine Learning (ICML)*, 2003.

[11] X. Zhu, "Semi-Supervised Learning Literature Survey", Computer Sciences TR 1530, University of Wisconsin, Madison, 2006.