ACCURATE SELF-ORGANIZING MAPS IN LEARNING METRICS

Arto Klami, Jaakko Peltonen, and Samuel Kaski *
Helsinki University of Technology
Neural Networks Research Centre
P.O Box 9800, FIN-02015 HUT, Finland
{Arto.Klami,Jaakko.Peltonen,Samuel.Kaski}@hut.fi
http://www.cis.hut.fi/projects/mi/

Improved methods are presented for learning metrics that measure only important distances. It is assumed that changes in primary data are relevant only to the extent that they cause changes in auxiliary data, available paired to the primary data. The metrics are here derived from estimators of the conditional density of the auxiliary data, and computational approximations of the distances are used. We apply learning metrics to improve the performance of Self-Organizing Maps (SOMs). A new performance indicator is presented that measures the accuracy of SOMs in preserving the distribution of the auxiliary data. The SOM in learning metrics is compared with the traditional SOM and the supervised SOM, and learning metrics are found to improve the accuracy of maps in this sense.

1 INTRODUCTION

Variable selection or feature extraction is a pressing problem for all exploratory data analysis. The results of exploration methods such as clustering and visualization methods are largely determined by the selected variables.

The goal of variable selection is to find a representation of data that reveals the interesting aspects underlying the data set. Data sets usually contain unimportant measurements, and the relevance of the features is highly dependent on the specific task on hand. To fight these problems, the variable selection is often done by a specialist of the application field.

The selected variables are often used for distance computation, i.e. to measure whether two samples are similar. Therefore, an alternative approach to the same problem is to choose a suitable metric for the data space.

Here we present one solution, based on the *learning metrics* principle [6, 7, 11], for the choice of the metric. The metric is scaled locally to measure distances only in important directions, where importance is obtained from auxiliary data.

2 THEORY OF LEARNING METRICS

Assume that there exist auxiliary data c paired with the primary data $\mathbf{x}$. Here $\mathbf{x}$ is a real-valued vector and c is categorical, i.e. it has finite number N_C of possible values. We

*This work was supported by the Academy of Finland, in part by the grant 52123.

are interested in studying the primary data, but the changes in $\mathbf{x}$ are assumed relevant only to the extent they cause changes in c.

An example situation would be to examine companies based on their financial statements. By choosing the knowledge of whether a company has gone bankrupt as the auxiliary variable, we can study what financial indicators affect the risk of bankruptcy in different situations.

The relevance information contained in the auxiliary data is taken into use by defining a metric that corresponds with the changes in the conditional distribution of c. We define the distance in learning metrics locally, between two close-by points $\mathbf{x}$ and $\mathbf{x} + d\mathbf{x}$ of the primary data space, by

$$d^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) \equiv D_{KL}(p(c|\mathbf{x}), p(c|\mathbf{x} + d\mathbf{x})) = d\mathbf{x}^T \mathbf{J}(\mathbf{x}) d\mathbf{x} \; . \tag{1}$$

Here D_{KL} is the Kullback-Leibler divergence and $\mathbf{J}(\mathbf{x})$ is the Fisher information matrix

$$\mathbf{J}(\mathbf{x}) = E_{p(c|\mathbf{x})} \left[\left(\nabla_{\mathbf{x}} \log p(c|\mathbf{x}) \right) \left(\nabla_{\mathbf{x}} \log p(c|\mathbf{x}) \right)^T \right] \tag{2}$$

parameterized by $\mathbf{x}$.

The global distances are measured by integrating (1) over the path that produces the smallest value. In this paper, we discuss an approach where the distances are calculated explicitly by approximations of the global distance. Alternatively, the learning metrics principle can be used by incorporating the metric into the cost function. This has been studied for example in [11] and is not considered here. The connections between the two approaches to the principle are discussed in [6].

3 SELF-ORGANIZING MAPS IN LEARNING METRICS

Learning metrics can in principle be used with any method involving distance calculations. Here we use it with the Self-Organizing Map (SOM) [8]. Self-Organizing Maps in learning metrics have earlier been studied in [7, 10].

A SOM is a regular lattice of units. Each unit i contains a model vector $\mathbf{m}_i$, which represents particular kinds of data in the data space. The model vectors are trained with an iterative algorithm to follow the data. At the same time, the SOM organizes so that the units close to each other in the lattice have similar model vectors.

The SOM training algorithm iterates two steps, winner search and adaptation. At each iteration t, an input sample $\mathbf{x}(t)$ is picked at random from the data, and a winner unit $w(t)$ is selected by finding the model vector closest to the sample by

$$w(t) = \arg\min_i d^2(\mathbf{x}(t), \mathbf{m}_i(t)) \; . \tag{3}$$

Here d^2 can be any distance function. Traditionally the distance has been either in the Euclidean or inner product metrics. In our case it is in the learning metric derived from the auxiliary data. For brevity, we call a SOM trained in learning metrics SOM-L and a SOM trained in Euclidean metrics SOM-E.

Once the winner has been selected, the model vectors are all adapted towards the input sample in the steepest descent direction. In Euclidean metrics, the direction is given by the gradient and in learning metrics by the natural gradient [1].

For a local distance approximation used in this paper, the natural gradient leads to the update rule

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t) h_{wi}(t)(\mathbf{x}(t) - \mathbf{m}_i(t)) \ , \tag{4}$$

which is the familiar SOM adaptation rule. Here $\alpha(t)$ is the learning rate and $h_{wi}(t)$ is the neighborhood function, a decreasing function of the distance between units i and $w(t)$ on the SOM lattice. In this paper, a Gaussian neighborhood was used.

In order to use SOM-L in practice, we need to make two approximations. Firstly, the local metric is computed from an estimate of the conditional auxiliary distribution $p(c|\mathbf{x})$. Secondly, the global distances between samples $\mathbf{x}$ and model vectors $\mathbf{m}$ are approximated.

4 ESTIMATING THE AUXILIARY DISTRIBUTION

In practice, we do not know the conditional probability density $p(c|\mathbf{x})$. In order to use SOM-L, we must estimate the density from the data. Here we discuss three parametric density estimation methods, all based on Gaussian kernels and used earlier in [10].

The conditional probabilities $p(c|\mathbf{x})$ can be obtained by estimating the joint density $p(c, \mathbf{x})$ and deriving the conditional probabilities from it, or by directly estimating the conditional distribution. As the metric is based on the matrix $\mathbf{J}(x)$ computed from the conditional distribution, estimating $p(c|\mathbf{x})$ directly should provide better results.

Here we use one estimator of the joint density, a version of Mixture Discriminant Analysis (MDA2) [3], and two estimators that directly estimate the conditional probabilities $p(c|\mathbf{x})$. The first is a kind of mixture of experts [5], where the experts are distributions $\psi_j = [\psi_{j1}...\psi_{jN_C}]$ constant with respect to $\mathbf{x}$, and the gating network is formed of Gaussian functions $y_j(\mathbf{x})$. The density estimate has the form

$$\hat{p}(c_i|\mathbf{x}) = \sum_{j=1}^{N_U} y_j(\mathbf{x}) \psi_{ji} \ . \tag{5}$$

Here N_U is the number of components, $y_j(\mathbf{x})$ are multivariate Gaussians with covariance matrix $\sigma^2 I$, normalized so that $\sum_{j=1}^{N_U} y_j(\mathbf{x}) = 1$, and ψ_{ji} tells the probability of class i given component j.

The other estimator is a simple product of experts [4], defined by

$$\hat{p}(c_i|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{j=1}^{N_U} \exp\left(y_j(\mathbf{x}) \log \psi_{ji}\right) \ , \tag{6}$$

where the y_j and ψ_{ji} are defined as before and the sum of probabilities is normalized to one by the function $Z(x) = \sum_{i=1}^{N_C} \prod_{j=1}^{N_U} \exp\left(y_j(\mathbf{x}) \log \psi_{ji}\right)$.

The MDA2 is fitted to data by maximizing the joint log-likelihood by the EM algorithm. For the other two models, the mean conditional log-likelihood of the auxiliary data is maximized by conjugate gradient algorithms. The variance σ^2 of the Gaussians is not learned by the training algorithms, but chosen to maximize the performance of SOM-L on a validation set. This is because the tests done so far have shown that the optimal variance for SOM-L training is often different from what is best for density estimation.

5 APPROXIMATING THE DISTANCE

The true global distance between a sample $\mathbf{x}$ and a model vector $\mathbf{m}$ is a minimum path integral of (1). As the matrix $\mathbf{J}(\mathbf{x})$ (2) depends on the $\mathbf{x}$ in a complex, non-linear way, accurate computation of the distance would be prohibitive.

A simple way of approximating the distance, introduced in [7], is to assume that $\mathbf{J}(\mathbf{x})$ is constant. In that case, the local distance measure at a point $\mathbf{x}$ can be extended to global distances to points $\mathbf{m}$, leading to the approximation

$$\hat{d}_1^2(\mathbf{x}, \mathbf{m}) = (\mathbf{m} - \mathbf{x})^T \mathbf{J}(\mathbf{x})(\mathbf{m} - \mathbf{x}) \; . \tag{7}$$

We call $\hat{d}_1$ the *1-point distance approximation* since it requires evaluating the local metric at one point $\mathbf{x}$. It is accurate when the model vectors are close to the sample $\mathbf{x}$. In SOM training, we need to find the closest model vector, and only the order of the model vectors is interesting, not the distances as such; therefore the inaccurate distances may be sufficient for training.

The smoothness of the auxiliary probability estimate affects the validity of the assumption of $\mathbf{J}(\mathbf{x})$ being constant. We noticed that SOM-L with the 1-point distance approximation requires far smoother estimates (larger values of dispersion parameter σ) than what would optimally describe the data in the sense of maximizing the likelihood of the probability estimates. If the estimate of $p(c|\mathbf{x})$ changes rapidly, the distance (7) is determined solely by the local properties of data which might not reflect the global distances well. Using smoother estimates helps, but the smaller details of the data are inevitably lost.

A more accurate but still computationally feasible approximation to the distance is obtained by assuming that the shortest path is a straight line between the sample $\mathbf{x}$ and the model vector $\mathbf{m}$. That is, the shortest path in learning metrics equals the shortest path in Euclidean metrics. The difference from the 1-point distance approximation is that the metric is allowed to change along the line.

The global distance can then be computed by dividing the connecting line evenly into T segments and evaluating the local metric at the start of each segment, resulting in

$$\hat{d}_T^2(\mathbf{x}, \mathbf{m}) = \frac{1}{T^2} \left(\sum_{t=1}^{T} \left((\mathbf{m} - \mathbf{x})^T \mathbf{J} \left(\mathbf{x} + \frac{t-1}{T}(\mathbf{m} - \mathbf{x}) \right) (\mathbf{m} - \mathbf{x}) \right)^{1/2} \right)^2 \; . \tag{8}$$

The *T-point distance approximation* $\hat{d}_T$ was first introduced in [10].

Unfortunately, the T-point distance approximation is computationally very demanding. The computational complexity of a single SOM-L training iteration with the T-point approximation is $\mathcal{O}(N_{DIM} N_C N_U N_{SOM} T)$ for N_{SOM} model vectors with dimensionality N_{DIM}, N_C classes, and N_U mixture components. By comparison, the complexity of the 1-point approximation is $\mathcal{O}(N_{DIM} N_C (N_U + N_{SOM}))$.

In many cases, the computational cost is too high. Since we are only interested in finding the closest model vector, it is unnecessary to compute the precise distances to the model vectors that are likely to be far from $\mathbf{x}$. Therefore we first make rough estimates of the distances and compute the T-point distance approximation only to the W model vectors that are closest according to the rough estimates. We call this procedure *winnowing* and use the 1-point distance approximation to make the rough estimates. This reduces the computational complexity to $\mathcal{O}(N_{DIM} N_C (N_U W T + N_{SOM}))$.

The values T and W can be tuned to compromise between the accuracy of the distance approximation and the computational cost. Here both values are set to ten, and all experiments with the T-point distance approximation are done with the winnowing procedure.

6 ALTERNATIVE METHODS

SOM-E is an established method for visualizing multidimensional data sets. We have previously [10] compared SOM-L with SOM-E as they can be used for the same task. In fact, SOM-E can be replaced with SOM-L if a suitable auxiliary variable is available.

It is intuitively clear that a method that is able to utilize more information performs better. Another method applicable to improving SOM training in the presence of extra information is the supervised Self-Organizing Map [8], originally proposed for improving the classification accuracy.

With supervised Self-Organizing Maps (here denoted SOM-S for brevity) we have vector-valued auxiliary data $\mathbf{y}$ paired with each data sample $\mathbf{x}$. The extra information is used in training by concatenating $\mathbf{x}$ with $\mathbf{y}$, that is, the SOM-S is fitted to data $[\mathbf{x}\ \mathbf{y}]$. For test data, the extra components corresponding to $\mathbf{y}$ are treated as missing values, meaning that only the components corresponding to $\mathbf{x}$ are used to find the winner units.

The extra information is such that the vector $\mathbf{y}$ is the same for samples from the same class, and different for samples from different classes; the SOM-S then enhances class separation on the map.

Here we need vector-valued auxiliary data while our original setting was that the auxiliary data is categorical. We encode the auxiliary variable c into a vector $\mathbf{y}$ by so-called *1-out-of-N* coding. For a sample $(\mathbf{x}, c)$ with $c = j$, the SOM-S input vector will be $[\mathbf{x}\ \mathbf{y}]$, where $\mathbf{y}$ is a vector of length N_C whose jth component is set to s and other components to zero.

The value s governs the importance of the auxiliary data for winner selection, and is here chosen to maximize the performance of SOM-S on a validation set. Small values of s mean that the auxiliary data has little effect; winners are mostly selected by primary features $\mathbf{x}$. If the value is very large, the map is based almost solely on the auxiliary data and the structure of the primary data is lost.

7 ACCURACY OF SELF-ORGANIZING MAPS

Previously [10] we have measured the performance of SOMs by computing the conditional likelihood of the auxiliary data at the winner units of test samples. A problem with this measure is that a density estimator is required also for evaluating the accuracy of the traditional methods (SOM-E and SOM-S) that do not use such estimators in training. Thus different results are obtained for the same SOM with different estimators.

Here we present a new way for measuring the accuracy of a SOM in the context of learning metrics. The auxiliary data describes what is interesting in the data, and it is important to preserve relevant aspects of the data in the SOM projection.

We postulate that a good projection should not mix concentrations of auxiliary values more than is necessary. We measure this by computing, in a sense, how 'pure' the auxiliary distributions of test samples are in each SOM unit. The measure is smoothed over neighboring units in order to also measure the homogeneity of auxiliary distributions.

Table 1: *The Data Sets*

Data set	Dimensions	Classes	Samples
Landsat Satellite Data *	36	6	6435
Letter Recognition Data *	16	26	20000
Phoneme Data from LVQ_PAK [9]	20	14	3656
TIMIT Data from [12]	12	41	14994

* from the UCI Machine Learning Repository [2]

To be precise, the accuracy of a SOM is computed as the logarithmic conditional likelihood of N_{TEST} test samples $\{\mathbf{x}_i, c_i\}_{i=1}^{N_{TEST}}$, based on smoothed proportions of auxiliary values of the test samples projected onto the map. We define

$$Accuracy = \sum_{i=1}^{N_{TEST}} \log \frac{\sum_{j=1}^{N_{SOM}} a_{jw_i} \frac{N_{jc_i}}{N_j}}{\sum_{j=1}^{N_{SOM}} a_{jw_i}} , \tag{9}$$

where N_{ji} denotes the number of test samples whose winner unit is j and whose auxiliary variable has value $c = i$, and N_j is the total number of samples with winner unit j. The winner unit of the ith sample is denoted by w_i and the corresponding auxiliary variable value is c_i.

The projected distributions N_{jc_i}/N_j are smoothed over neighboring SOM units by weights a_{jw_i} computed as products of Gaussian kernels and the number of samples N_j, i.e. we set

$$a_{jw_i} = \exp\left(-D^2(j, w_i)/2\lambda^2\right) N_j \ . \tag{10}$$

Here $D(j, w_i)$ is the distance between unit j and the winner unit w_i on the SOM lattice. The parameter λ governs the smoothness of the densities and therefore defines how local the smoothed conditional distributions are. We used the value $\lambda = 1$ which equals the radius of the neighborhood function at the end of SOM training. The weights a_{jw_i} are affected by the number of samples in order to emphasize the more reliable estimates.

Both indicators aim to measure how well the SOM represents changes in the auxiliary distributions. The indicator we used previously, the likelihood at the winner units, gives high values if the auxiliary distributions of test samples are similar to the estimated conditional distributions at the winner units. The indicator (9) presented here measures the homogeneity (with respect to the auxiliary data) of test samples projected close to each other on the SOM.

According to empirical tests (not shown in this paper), both indicators measure the accuracy quite similarly; parameter and method validation produces similar choices with both indicators.

8 EMPIRICAL TESTS

We tested the methods on four real world data sets (Table 1). On each data, the class labels were used as the auxiliary information c, and the data sets were preprocessed by removing classes with only a few samples.

Table 2: *The p-values of the paired t-tests. Each entry means that the method on that row is on average better than the method on that column. Significant differences ($p < 0.01$) are underlined for convenience.*

	Letter				Landsat		
	1-point	SOM-S	SOM-E		1-point	SOM-S	SOM-E
T-point	$\underline{6 \times 10^{-8}}$	$\underline{10^{-9}}$	$\underline{< 10^{-10}}$	T-point	$\underline{4 \times 10^{-4}}$	$\underline{10^{-5}}$	$\underline{3 \times 10^{-8}}$
1-point	-	$\underline{2 \times 10^{-8}}$	$\underline{< 10^{-10}}$	1-point	-	0.04	$\underline{8 \times 10^{-5}}$
SOM-S	-	-	$\underline{< 10^{-10}}$	SOM-S	-	-	$\underline{10^{-4}}$

	LVQ_PAK				TIMIT		
	T-point	SOM-E	1-point		SOM-S	1-point	SOM-E
SOM-S	0.98	$\underline{0.008}$	0.10	T-point	$\underline{2 \times 10^{-5}}$	$\underline{3 \times 10^{-6}}$	$\underline{3 \times 10^{-9}}$
T-point	-	0.014	0.03	SOM-S	-	0.02	$\underline{2 \times 10^{-5}}$
SOM-E	-	-	0.21	1-point	-	-	$\underline{0.004}$

The significance of the difference between SOM-L and the traditional methods was tested using a 10-fold cross-validation. The parameters σ and s were validated anew for each fold by finding the values that provided the most accurate maps on a validation set separated from each training set. The accuracy (9) for the best SOM was then computed on the corresponding test set.

For SOM-L, we made preliminary runs with the 1-point distance approximation to find the best density estimation method for each data set. We used 10, 30, and 100 kernels in the density estimation phase and selected the estimation method that produced the SOM-L with the best likelihood. Thanks to this (suboptimal) choice, we did not have to validate over the density estimation methods during the cross-validation. For the T-point distance approximation, we fixed the number of kernels to 30 to further reduce computation time.

9 RESULTS

We tested the significance of the differences between the methods by paired t-tests. The resulting p-values are collected in Table 2.

On three of four data sets, SOM-L attains improved results. With the T-point distance approximation, SOM-L is significantly better than both of the traditional methods. SOM-S is also significantly better than SOM-E. The latter is to be expected since SOM-S uses auxiliary information, while SOM-E does not. From another viewpoint, this shows that the new accuracy measure detects the difference between SOM-S and SOM-E as it should.

With the 1-point distance approximation, SOM-L is significantly better than SOM-E on the same three data sets. Compared with SOM-S, only one of the differences is significant in favor of SOM-L. The 1-point distance approximation is thus inadequate. Notice additionally that the T-point distance approximation is significantly better than the 1-point distance approximation.

On the fourth data set, the LVQ_PAK data, SOM-L is almost significantly better than SOM-E ($p = 0.014$) and comparable to SOM-S. SOM-S is again significantly better than SOM-E. SOM-L with the 1-point distance approximation performs on average worse

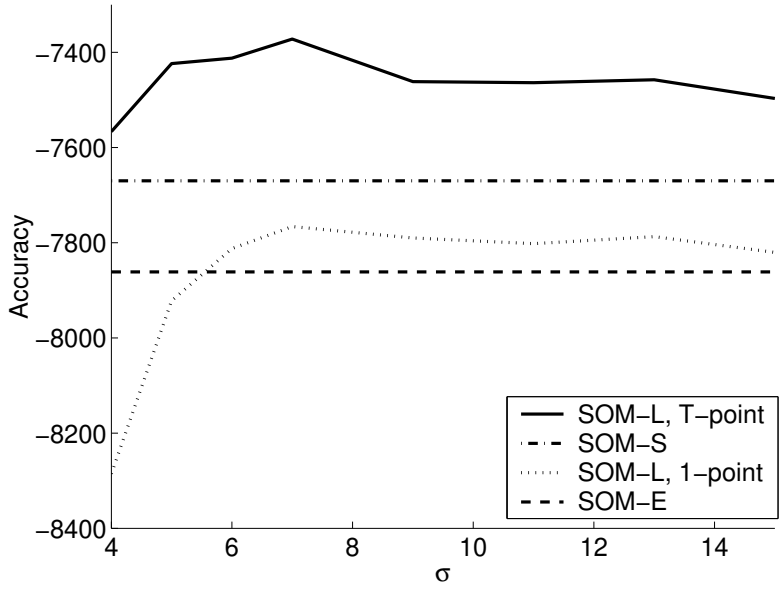

Figure 1: *Accuracy (9) of the methods on the TIMIT data, averaged over the 10-fold validation sets. The T-point distance approximation is clearly superior to the 1-point distance approximation and to the traditional methods. SOM-L results are shown as a function of the dispersion parameter σ of the conditional probability estimate. SOM-S is here depicted for the value s that produced the best average accuracy. Note that SOM-E equals SOM-S with $s = 0$, so the SOM-E line presents the lowest value of SOM-S.*

than either of the traditional SOMs, but the differences are not significant.

The mixture of experts (5) was the best density estimator on three data sets and the product of experts (6) was best on the LVQ_PAK data. It seems that the methods that directly estimate the conditional density outperform the joint density estimator MDA2 as density estimation methods for learning metrics.

The results are illustrated for one data set in Figure 1. With the T-point distance approximation, SOM-L is superior to the traditional methods on a wide range of smoothing parameter values. SOM-S is at least equal to SOM-E regardless of the length s of the subvectors for the auxiliary data.

10 DISCUSSION

We have shown that on most data sets, the Self-Organizing Map in learning metrics is able to preserve the relevant aspects of the data better than traditional methods, the Self-Organizing Map and the supervised Self-Organizing Map. We measured this by an indirect indicator that measures how well the auxiliary information is retained in the SOM projection.

Using more accurate distance approximations instead of the previously used local approximation is necessary for good results. Computing the T-point distance approximation is computationally intensive. A heuristic speedup was presented, and the experiments showed that the maps were accurate even with the speedup.

The accuracy measure presented in this paper seems reasonable as it is able to reveal

the difference between SOM-S and SOM-E on every data set. It is computable based only on the winner units and does not otherwise require knowledge of the metric.

There is still room for improvement in both the density estimation and the distance approximation. The current distance approximation relies heavily on density evaluations. This means that more complex estimators would increase the computation time. On the other hand, the time taken by fitting the density estimator is small compared with the SOM-L training time. Therefore the importance of quick estimator fitting is low.

REFERENCES

[1] Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10:251–276, 1998.

[2] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.

[3] Trevor Hastie, Robert Tibshirani, and Andreas Buja. Flexible discriminant and mixture models. In J. Kay and D. Titterington, editors, *Neural Networks and Statistics*. Oxford University Press, 1995.

[4] G. E. Hinton. Products of experts. In *Proceedings of ICANN99, the Ninth International Conference on Artificial Neural Networks*, pages 1–6. IEE, 1999.

[5] M. I. Jordan. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

[6] Samuel Kaski and Janne Sinkkonen. Principle of learning metrics for data analysis. *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology, special issue on Data Mining and Biomedical Applications of Neural Networks*, 2002. Accepted for publication.

[7] Samuel Kaski, Janne Sinkkonen, and Jaakko Peltonen. Bankruptcy analysis with self-organizing maps in learning metrics. *IEEE Transactions on Neural Networks*, 12:936–947, 2001.

[8] Teuvo Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1995. (Third, extended edition 2001).

[9] T. Kohonen, J. Kangas, J. Laaksonen, and K. Torkkola. LVQ_PAK: A program package for the correct application of Learning Vector Quantization algorithms. In *Proceedings of IJCNN'92, International Joint Conference on Neural Networks*, volume I, pages 725–730, 1992.

[10] Jaakko Peltonen, Arto Klami, and Samuel Kaski. Learning more accurate metrics for self-organizing maps. In *Artificial Neural Networks - ICANN 2002*, pages 999–1004, Springer-Verlag, Berlin, 2002.

[11] Janne Sinkkonen and Samuel Kaski. Clustering based on conditional distributions in an auxiliary space. *Neural Computation*, 14:217–239, 2002.

[12] CD-ROM prototype version of the DARPA TIMIT acoustic-phonetic speech database, 1998.