Beyond Standard Metrics – On the Selection and Combination of Distance Metrics for an Improved Classification of Hyperspectral Data

Uwe Knauer, Andreas Backhaus, and Udo Seiffert

Fraunhofer IFF
Joseph-v.-Fraunhofer-Str. 1, 39106 Magdeburg, Germany
{uwe.knauer,andreas.backhaus,udo.seiffert}@iff.fraunhofer.de
http://www.iff.fraunhofer.de

Abstract. Training and application of prototype based learning approaches such as Learning Vector Quantization, Radial Basis Function networks, and Supervised Neural Gas require the use of distance metrics to measure the similarities between feature vectors as well as class prototypes. While the Euclidean distance is used in many cases, the highly correlated features within the hyperspectral representation and the high dimensionality itself favor the use of more sophisticated distance metrics. In this paper we first investigate the role of different metrics for successful classification of hyperspectral data sets from real-world classification tasks. Second, it is shown that considerable performance gains can be achieved by a classification system that combines a number of prototype based models trained on differently parametrized divergence measures. Data sets are tested using a number of different combination strategies.

Keywords: Divergence, Metrics, Hyperspectral, SNG, GLVQ, RBF.

1 Introduction

The optical characterization of organic and inorganic materials with hyperspectral imaging is becoming a widespread application within plant breeding, smart farming, material sorting, or quality control in food production. The generic behavior of the material to reflect, absorb, or transmit light is used to characterize its identity and even molecular composition. A hyperspectral camera records a narrowly sampled spectrum of reflected or transmitted light in a certain wavelength range and produces a high-dimensional pattern of highly correlated spectral channels per image pixel. Often, the direct relationship between this pattern and the target value, for example a material category is unknown. In the simple case exact spectral bands are known that correlate with the presence of certain chemical compounds. If such direct knowledge is unavailable, machine learning is used to learn a classification or regression task from available labeled reference data.

Prototype based models like the Learning Vector Quantization [12], Supervised Neural Gas [11], or Radial Basis Function Networks [21] provide a set of

T. Villmann et al. (eds.), *Advances in Self-Organizing Maps and Learning Vector Quantization,* Advances in Intelligent Systems and Computing 295,
DOI: 10.1007/978-3-319-07695-9_16,

tools to learn a classification task from high dimensional data. These methods utilize a certain similarity measure to compare an input pattern to a number of stored prototypes in order to predict the pattern's category. Commonly the Euclidean distance is used to calculate the similarity of the input and prototype pattern. Each feature is compared separately irrespective of its position in the high dimensional feature vector. In contrast, spectral pattern are data samples that describe a function or distribution of energy across a well ordered wavelength range. Therefore this type of data is also called 'functional data'.

An approach to calculate the similarity of statistical distributions are divergences which offer an alternative way to characterize dissimilarity between spectral patterns. Additionally, more general divergences like the γ-divergence include parameters that can be potentially tuned to adapt the dissimilarity measure to the learning task at hand. Divergence dissimilarity measures have been successfully integrated into prototype based machine learning models but performance gains have been minimal so far on models using just a single dissimilarity measure [22,26,15,27].

This paper shows that considerable performance gains can be achieved by a classification system that combines a number of prototype based models trained on differently parametrised divergence measures. A number of hyperspectral data sets from real-world classification tasks are tested using a number of different combination strategies.

2 Related Work

The idea to include task-adaptive non-standard metrics and dissimilarity function into a pattern recognition system has been widely researched. In [24] the Mahalanobis distance replaces the standard Euclidean distance. The Mahalanobis matrix is not calculated as the co-variance matrix but a distance metric learning method is used to calculate a transformation which assures small distance between nearest neighboring points from the same class and separation of points belonging to different classes by large margin. Likewise in [23], the label information of the data is used to calculate a task-specific distance function based on the Kullback-Leibler divergence. The distance is based on the conditional distribution of label information in dependence to the input data which is estimated on a validation set. In [1] the behaviour of the Minkowski distance to measure proximity especially in high-dimensional feature spaces was investigated. The methods highlighted have in common that they treat the process to find an adaptive metric separately from the actually learning of the classification model. In contrast, parameterized metrics and dissimilarity functions can be directly integrated into the learning process of models like GLVQ, SNG, or RBF. One parameterization is the use of relevance weights or matrices in the Euclidean distance [12,25,20]. Another possibility is the use of the generalized metric, in the case of the Euclidean norm the Minkowski norm as well as the use of divergences [26,15], for example the γ-divergence and its special case the Cauchy-Schwarz divergence [22]. Parameters are either systematically explored

or learned directly with other model parameters through minimizing the models object/energy function. So far the utilization of a distance function tuned to a single parameter setting has not shown significant performance improvements. The approach explored in this paper is the combination of a number of models, tuned to different γ parameters in order to create a classifier system whose global performance is significantly better then the performance of each model tuned to a single parameter.

In the field of multiple classifier fusion, several approaches have been proposed to create classifier ensembles with superior classification performance as well as to combine sets of existing classifiers to overcome the limitations of individual classifiers [13,4,17,2,6,14,16].

As we study the impact of parameters such as model size and distance metric, a large number of classifiers is trained for evaluation purposes by systematically varying these factors. However, this approach creates an ensemble of classifiers which may provide diverse as well as correlated decisions on the training and testing data.

While correlation and diversity between classifiers can be simply measured, it remains an open question which level of diversity and correlation provides the best results in classifier fusion [7]. In common approaches such as Bagging and Boosting, diversity is fostered by random sampling or by iteratively generating complementary classifiers for falsely classified feature vectors. However, these approaches also require a high level of correlation of the individual classifiers because final decisions are obtained by majority voting.

The existing approaches for classifier fusion can be roughly divided into trained and non-trained combiners [8]. Also early and late fusion can be easily discriminated. The topology of fusion methods is another important aspect to categorize the different approaches.

While the application of non-standard distance metrics is motivated by previous work on classification of functional data, its impact on the generation of classifier ensembles for the same problem is unknown. As trained combiners have shown superior performance in a previous study with non-functional data [18], we focus on ensemble learning with decision tree based learners. The advantages of using tree based learners are sketched in the next section.

3 Methods

3.1 Training and Evaluation of RBF, GLVQ, and SNG Classifiers

Classification models were implemented as published in [12,11,3]. For the GLVQ and SNG no non-linearity in the energy function was used. The distance function between a data vector $\mathbf{v}$ and a prototype vector $\mathbf{w}$ (respectively the hidden neurons in the RBF) was either the squared Euclidean distance defined as

$$d(\mathbf{v}, \mathbf{w}) = \sum_i (v_i - w_i)^2 , \tag{1}$$

or the γ-divergence defined as

$$d(\mathbf{v}, \mathbf{w}, \gamma) = \log \left(\frac{\left(\sum_i v_i^{\gamma+1}\right)^{\frac{1}{\gamma(\gamma+1)}} \left(\sum_i w_i^{\gamma+1}\right)^{\frac{1}{\gamma+1}}}{\left(\sum_i v_i w_i^{\gamma}\right)^{\frac{1}{\gamma}}} \right). \tag{2}$$

The γ-divergence with $\gamma = 2$ is widely known as the Cauchy-Schwarz distance. The model training in all three classifier systems (RBF, SNG, GLVQ) is essentially an energy minimization problem. In the standard learning scheme, stochastic gradient descent with step-sizes manually set for different parameters are used. In order to avoid a manually chosen step-size, we used the non-linear conjugate gradient approach with automatic step size from the optimization toolbox 'minFunc' available for Matlab. For this we provided the energy function as well as the first derivatives according to all model parameters. The parameter γ was set varying from 1 to 10 in steps of one. Additionally, the generalized Kullback-Leiber divergence [10] was used to investigate the behavior for convergence of γ to zero. Prototype vectors and network weights were initialized randomly. The RBF used a 1-of-N coding scheme at its output to represent discrete class information. In the RBF, SNG, and GLVQ the prototypes were pre-trained using a Neural Gas with the Euclidean distance or γ-divergence as similarity function with an identical setup compared to the later classification model. In the GLVQ and SNG model, separate pre-learning runs for prototypes from identical classes were performed. The dataset was divided into training and test data according to a 5-fold cross validation scheme with stratified random sampling. After training, the predicted labels for the test data with the respective model were collected as well as scalar model outputs. In case of the RBF, the scalar output was the output of the linear output layer. For the GLVQ and SNG we used the distances to the closest prototype of the same class as well as the smallest distance to a prototype of any other class as scalar output. We set 20, 30, or 40 as total number of prototypes/hidden neurons in all three models. In the GLVQ and SNG an identical number of prototypes per class was used. In addition to the Euclidean distance we also used weighted Euclidean distance as an alternate distance metric where the weights are automatically adapted in the training phase.

3.2 Fusion of RBF, LVQ, and SNG Results

The real-valued scalar outputs of the different classifiers make a feature vector which is used as the input for learning a combining rule. In this study, we focus on the application of decision tree based learners. Algorithms such as C4.5 or its variant J4.8 use local optimization of a threshold value and selection of a single input feature to maximize the separation into given target classes. As the input features are the output values of classifiers itself this is similar to the selection of operating points as known from receiver operating characteristic (ROC) and precision recall (PR) analysis. Hence, any decision of the resulting

trees can be easily interpreted as a sequence of operating point selections for the different input classifiers. To overcome known limitation of decision tree learning, ensembles of trees are used instead. Hence, for the combination of the different classifiers these methods are used:

1. AdaBoost with decision trees [9],
2. Random Forests [5], and
3. CRAGORS (cascaded reduction and growing of result sets) [19].

The used implementations of AdaBoost, decision trees, and Random Forests are part of the Spider toolbox and WEKA. For AdaBoost pruned decision trees are used for better generalization performance. For Random Forest classifiers unpruned trees are used. Boosting was set to 10 iterations and all Random Forest classifiers consist of 10 trees as well. CRAGORS is included to address the tradeoff between ensemble size and ensemble accuracy. This combining algorithm is expected to provide less accurate results, but to select small subset of relevant input classifiers which already provide a significant improvement in classification accuracy. The dataset for testing combination performance has been generated from the outputs of 5-fold cross-validation of the individual classifiers. For every spectrum the outputs of all the different classifiers have been collected. 10-fold cross-validation was used to obtain average accuracy values for the 3 combining methods.

4 Datasets

The hyperspectral datasets have been selected from several industrial applications where hyperspectral imaging can be used for the detection of a desired target material or defective objects for a subsequent material sorting. We deliberately chose classification tasks that showed mediocre classification accuracy on single prototype based models. Five binary classification problems were chosen for this publication:

1. Detection of aluminium within waste material,
2. Classification of mature vs. immature coffee beans,
3. Detection of putrid hazelnuts among healthy hazelnuts,
4. Detection of fungi infested hazelnuts among healthy hazelnuts, and
5. Anomality detection on the surface of fluffed pulp.

We limited our study to two-class problems for two major reasons. First, the detection of a single important class is a typical scenario in industrial applications. Hence, the above datasets have been collected separately. Especially the hazelnut datasets belong to different studies. Second, the current implementation of CRAGORS which we wanted to test on hyperspectral datasets is so far limited to two-class problems.

For the hyperspectral image acquisition, material samples of one class were positioned with a standard optical PTFE (polytetrafluoroethylene) calibration pad on a translation table. Hyperspectral images were recorded using a HySpex

Table 1. Average accuracy of base classifiers (5-fold cross-validation)

	RBF			SNG			GLVQ		
Datasets	L_2	γ	complete	L_2	γ	complete	L_2	γ	complete
D1	0.8150	**0.8265**	0.8265	0.7168	**0.7325**	0.7402	0.6853	**0.7110**	0.7110
D2	**0.8560**	0.8430	0.8560	0.6937	**0.7027**	0.7295	**0.6820**	0.6755	0.6820
D3	0.6680	**0.6813**	0.6813	0.5710	**0.6005**	0.6005	0.5370	**0.5493**	0.5530
D4	**0.9618**	0.9496	0.9618	0.7724	**0.7894**	0.7894	0.7626	0.7626	0.7626
D5	0.7452	**0.7635**	0.7635	0.6597	**0.6963**	0.6963	0.6312	**0.6575**	0.6575

SWIR-320m-e line camera (Norsk Elektro Optikk A/S). Spectra are from the short-wave infra-red range (SWIR) of 970 nm to 2,500 nm at 6 nm resolution yielding a 256 dimensional spectral vector per pixel. The camera line has a spatial resolution of 320 px and can be recorded with a maximum frame rate of 100 fps. Radiometric calibration was performed using the vendors software package and the PTFE reflectance measure. Material was segmented from background via Neural Gas clustering. From each material class, 2,000 labeled spectral samples for each class were chosen randomly and combined to the datasets representing the two-class problems listed above. Spectral vectors were normalized to unit length.

5 Results and Discussion

As a baseline we measured the accuracy gain of using γ-divergence instead of the Euclidean distance for the datasets D1 to D5. Tab. 1 lists the accuracies of GLVQ, RBF, and SNG classifiers with respect to the used metric. The comparison shows a minor improvement in the accuracy when using the γ-divergence only for a few datasets. This is in accordance to previously reported results on using alternative distance measures [22,26,15,27]. For SNG classifiers the γ-divergence yields better results on all datasets. However, RBF classifiers outperform SNG and GLVQ on all datasets. The column *complete* lists the best results obtained from a slightly extended set of base classifiers including the Kullback-Leibler divergence and Cauchy-Schwarz divergence measures.

As we set 20, 30, or 40 as total number of prototypes/hidden neurons in all three models and also trained models for different values of γ, only the average accuracy of the best performing classifier is shown. Tab. 2 lists the parameter settings for these classifiers. We found, that the best results are obtained by different settings of the number of prototypes/hidden neurons as well as different similarity measures. Especially, γ differs significantly for the datasets for which application of γ-diversity is beneficial.

Tab. 3 shows that considerable performance gains can be achieved by the proposed classification system that combines a number of prototype based models trained on differently parametrized divergence measures. We combined the results of different subsets of the GLVQ, RBF, and SNG classifiers. By considering only variants of Euclidean based classifiers a significant increase in accuracy

Table 2. Parameter settings of the best classifiers, including Kullback Leibler divergence (KLD)

RBF	RBF		
Datasets	L_2	γ	complete
D1	40 neurons	20 neurons, $\gamma = 5$	same as γ
D2	30 neurons	40 neurons, $\gamma = 1$	same as L_2
D3	20 neurons	40 neurons, $\gamma = 5$	same as γ
D4	40 neurons	30 neurons, $\gamma = 2$	same as L_2
D5	40 neurons	40 neurons, $\gamma = 2$	same as γ
	SNG		
Datasets	L_2	γ	complete
D1	30 neurons	30 neurons, $\gamma = 1$	40 neurons, KLD
D2	20 neurons	40 neurons, $\gamma = 6$,$\gamma = 2$	same as γ
D3	40 neurons	40 neurons, $\gamma = 1$	same as γ
D4	30 neurons	30 neurons, $\gamma = 10$	same as γ
D5	40 neurons	40 neurons, $\gamma = 10$	same as γ
	GLVQ		
Datasets	L_2	γ	complete
D1	40 neurons	40 neurons, $\gamma = 6$	same as γ
D2	20 neurons	40 neurons, $\gamma = 1$	same as L_2
D3	20 neurons, weighted Euclidean	20 neurons, $\gamma = 2$	same as γ
D4	40 neurons	40 neurons, $\gamma = 8$	same as γ and L_2
D5	40 neurons	30 neurons, $\gamma = 2$	same as γ

is found. However, γ-divergence based classifier ensembles perform better on all datasets. Especially, for datasets D3 and D5 a large difference between Euclidean and γ based ensembles exists for all groups of combined classifiers (RBF, GLVQ, SNG). Additional improvements are possible if Euclidean and γ-diversity based classifiers are merged. Adding the Kullback-Leibler divergence based classifiers does not further improve the results significantly. As before, only the results of the best combination algorithm is shown in Tab. 3. For all tested datasets Random Forest and Boosted Decision Trees are competitive and there is no clear winner among these two methods. Additionally, the algorithm CRAGORS was used to find a subset of classifiers which provide a trade-off between the number of considered input classifiers and the gain in classification performance. The column *complete set* lists the results of a combination without discriminating beetween the pools of L_2-based and γ-based classifiers.

The comparison of the accuracies of different combining methods is shown in Tab. 4. The presentation is limited to the results of combining RBF network classifiers trained with γ-divergence based distance measures. The number in brackets reports the number of the used input features to indicate the trade-off between accuracy gain and the number of required RBF classifiers. The difference between AdaBoost and Random Forests is not significant. Also, the chosen limitation to 10 decision trees leaves room for additional improvements of the accuracy.

Table 3. Average accuracy of combined classifiers, 10-fold cross-validation

RBF	Ensemble pool			
Datasets	L_2	γ	$L_2 + \gamma$	complete set
D1	0.9737	**0.9965**	0.9982	0.9972
D2	0.9775	**0.995**	0.994	0.9960
D3	0.8010	**0.8985**	0.9155	0.9133
D4	0.9959	**0.9984**	0.9984	0.9992
D5	0.9205	**0.9790**	0.9825	0.9822

SNG	Ensemble pool			
Datasets	L_2	γ	$L_2 + \gamma$	complete set
D1	0.8465	**0.9698**	0.9660	0.9723
D2	0.8550	**0.9517**	0.9540	0.9550
D3	0.6190	**0.7463**	0.7412	0.7410
D4	0.9276	**0.9951**	0.9976	0.9976
D5	0.7798	**0.9377**	0.9383	0.9390

LVQ	Ensemble pool			
Datasets	L_2	γ	$L_2 + \gamma$	complete set
D1	0.7773	**0.9412**	0.9417	0.9463
D2	0.82	**0.8605**	0.8865	0.9012
D3	0.5493	**0.5877**	0.5867	0.5962
D4	0.9024	**0.9911**	0.9919	0.9878
D5	0.7505	**0.8830**	0.8953	0.8920

It should be noted that the used comparison method also contributes to classifier diversity. 5-fold cross-validation was used to train base classifiers and to collect realistic classifier outputs for unseen samples. Hence, all the classifiers have been trained on different subsets representing 80 percent of all samples. However, the Euclidean and γ-diversity based classifiers have been obtained under the same conditions. Therefore, the observed difference in performance is clearly related to the used metric. To study the different contributions to classifier diversity and performance in more detail, hold-out testing with independently sampled data should be used.

In contrast to other combining methods such as boosting the proposed approach is built on top of a set of independently tuned classifiers. The main advantage of a separation between tuning of the base classifiers and their combination into an ensemble is that it can be easily adapted to existing classification frameworks. Additionally, using a supervised classification algorithm for the combination instead of a simple combining rule such as majority voting is beneficial.

The results indicate that optimizing γ in the training of a single classifier may not yield the significant gain in accuracy as reported for the ensembles. Because we variied γ systematically over a small but meaningful range we expect such an approach to achieve a result competetive to our baseline condition.

Table 4. Average accuracy of different combined classifiers (RBF networks with γ-distances only, baseline includes all distance measures), 10-fold cross-validation, numbers in brackets denote required RBF networks

RBF	Fusion method			
Datasets	Baseline	CRAGORS	AdaBoost	Random Forest
D1	0.8265 (1)	0.9355 (12)	**0.9965 (48)**	0.9953 (55)
D2	0.856 (1)	0.9435 (13)	**0.9950 (53)**	0.9925 (59)
D3	0.6813 (1)	0.7445 (20)	**0.8985 (59)**	0.8855 (59)
D4	0.9618 (1)	0.9846 (7)	**0.9984 (25)**	**0.9984 (59)**
D5	0.7635 (1)	0.8840 (18)	**0.9790 (55)**	0.9683 (57)

6 Summary

The results show that choosing another metric or modifying model size may slightly improve classification accuracy. However, the tuning of parameters is required. The question remains whether other classification algorithms, other parameter setting than the tested ones, different model sizes, or a different topology of neural networks may yield better results or not. The major contribution of this paper with respect to γ-metrics and multiple classifier fusion is that it was possible to demonstrate for all tested datasets, that systematically varying the γ value of the distance metric is an extraordinarily effective way to create a diverse ensemble of classifiers. Especially, the trade-off between diversity and correlation seems to be near optimal for classification of hyperspectral data. Hence, the major contribution from an engineering perspective is to provide an easy-to-use framework for the analysis of hyperspectral data. By the fusion of classifier results, great improvements in classification accuracy have been made for several real-world applications. Moreover, for the first time the improvements reached a level which meets application specific lower boundaries on the precision and the detection rate. However, a lot of future work has to be done to get a deep theoretic understanding of the role of the γ-metric with respect to ensemble diversity. We also limited our study to the analysis of hyperspectral data as a representative of functional data. For practical applications, the selection of classifier subsets or a parallel computation of classifier results may be required to meet application specific time constraints.

References

1. Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 420–434. Springer, Heidelberg (2000)
2. Al-Ani, A., Deriche, M.: A new technique for combining multiple classifiers using the dempster-shafer theory of evidence. Journal of Artificial Intelligence Research 17, 333–361 (2002)
3. Backhaus, A., Bollenbeck, F., Seiffert, U.: Robust classification of the nutrition state in crop plants by hyperspectral imaging and artificial neural networks. In:

Proc. 3rd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Lisboa, Portugal (2011)
4. Bishop, C.M., Svensén, M.: Hierarchical Mixtures of Experts. In: 19th Conference on Uncertainty in Artificial Intelligence (2003)
5. Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)
6. Chen, X., Li, Y., Harrison, R., Zhang, Y.-Q.: Type-2 fuzzy logic-based classifier fusion for support vector machines. Applied Soft Computing 8(3), 1222–1231 (2008)
7. Didaci, L., Fumera, G., Roli, F.: Diversity in Classifier Ensembles: Fertile Concept or Dead End? In: Zhou, Z.-H., Roli, F., Kittler, J. (eds.) MCS 2013. LNCS, vol. 7872, pp. 37–48. Springer, Heidelberg (2013)
8. Duin, R.P.W.: The combining classifier: to train or not to train. In: ICPR (2002)
9. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: European Conference on Computational Learning Theory, pp. 23–37 (1995)
10. Geweniger, T., Kästner, M., Villmann, T.: Optimization of parametrized divergences in fuzzy c-means. In: ESANN (2011)
11. Hammer, B., Strickert, M., Villmann, T.: Supervised Neural Gas with general similarity measure. Neural Processing Letters 21, 21–44 (2005)
12. Hammer, B., Villmann, T.: Generalized relevance learning vector quantization. Neural Networks 15, 1059–1068 (2002)
13. Jordan, M.I., Jacobs, R.A.: Hierarchical Mixtures of experts and the EM-algorithm. Neural Computation 6(2), 181–214 (1994)
14. Kang, S., Park, S.: A fusion neural network classifier for image classification. Pattern Recogn. Lett. 30(9), 789–793 (2009)
15. Kästner, M., Backhaus, A., Geweniger, T., Haase, S., Seiffert, U., Villmann, T.: Relevance learning in unsupervised vector quantization based on divergences. In: Laaksonen, J., Honkela, T. (eds.) WSOM 2011. LNCS, vol. 6731, pp. 90–100. Springer, Heidelberg (2011)
16. Khreich, W., Granger, E., Miri, A., Sabourin, R.: Iterative Boolean combination of classifiers in the ROC space: An application to anomaly detection with HMMs. Pattern Recognition 43(8), 2732–2752 (2010)
17. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(3), 226–239 (1998)
18. Knauer, U., Seiffert, U.: A Comparison of Late Fusion Methods for Object Detection. In: IEEE International Conference on Image Processing, pp. 1–8 (2013)
19. Knauer, U., Seiffert, U.: Cascaded Reduction and Growing of Result Sets for Combining Object Detectors. In: Zhou, Z.-H., Roli, F., Kittler, J. (eds.) MCS 2013. LNCS, vol. 7872, pp. 121–133. Springer, Heidelberg (2013)
20. Mendenhall, M.J., Merényi, E.: Relevance-based feature extraction for hyperspectral images. IEEE Transactions on Neural Networks 19(4), 658–672 (2008)
21. Moody, J., Darken, C.J.: Fast learning in networks of locally tuned processing units. Neural Computation 1, 281–294 (1989)
22. Mwebaze, E., Schneider, P., Schleif, F.-M., Haase, S., Villmann, T., Biehl, M.: Divergence based Learning Vector Quantization. In: Verleysen, M. (ed.) 18th European Symposium on Artificial Neural Networks (ESANN 2010), pp. 247–252. d-side publishing (2010)
23. Peltonen, J., Klami, A., Kaski, S.: Learning more accurate metrics for self-organizing maps. In: Dorronsoro, J.R. (ed.) ICANN 2002. LNCS, vol. 2415, pp. 999–1004. Springer, Heidelberg (2002)

24. Płoński, P., Zaremba, K.: Improving performance of self-organising maps with distance metric learning method. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part I. LNCS, vol. 7267, pp. 169–177. Springer, Heidelberg (2012)
25. Schneider, P., Schleif, F.-M., Villmann, T., Biehl, M.: Generalized matrix learning vector quantizer for the analysis of spectral data. In: ESANN, pp. 451–456 (2008)
26. Villmann, T., Haase, S.: Divergence based vector quantization of spectral data. In: 2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), pp. 1–4 (2010)
27. Villmann, T., Haase, S.: Divergence-based vector quantization. Neural Comput. 23(5), 1343–1392 (2011)