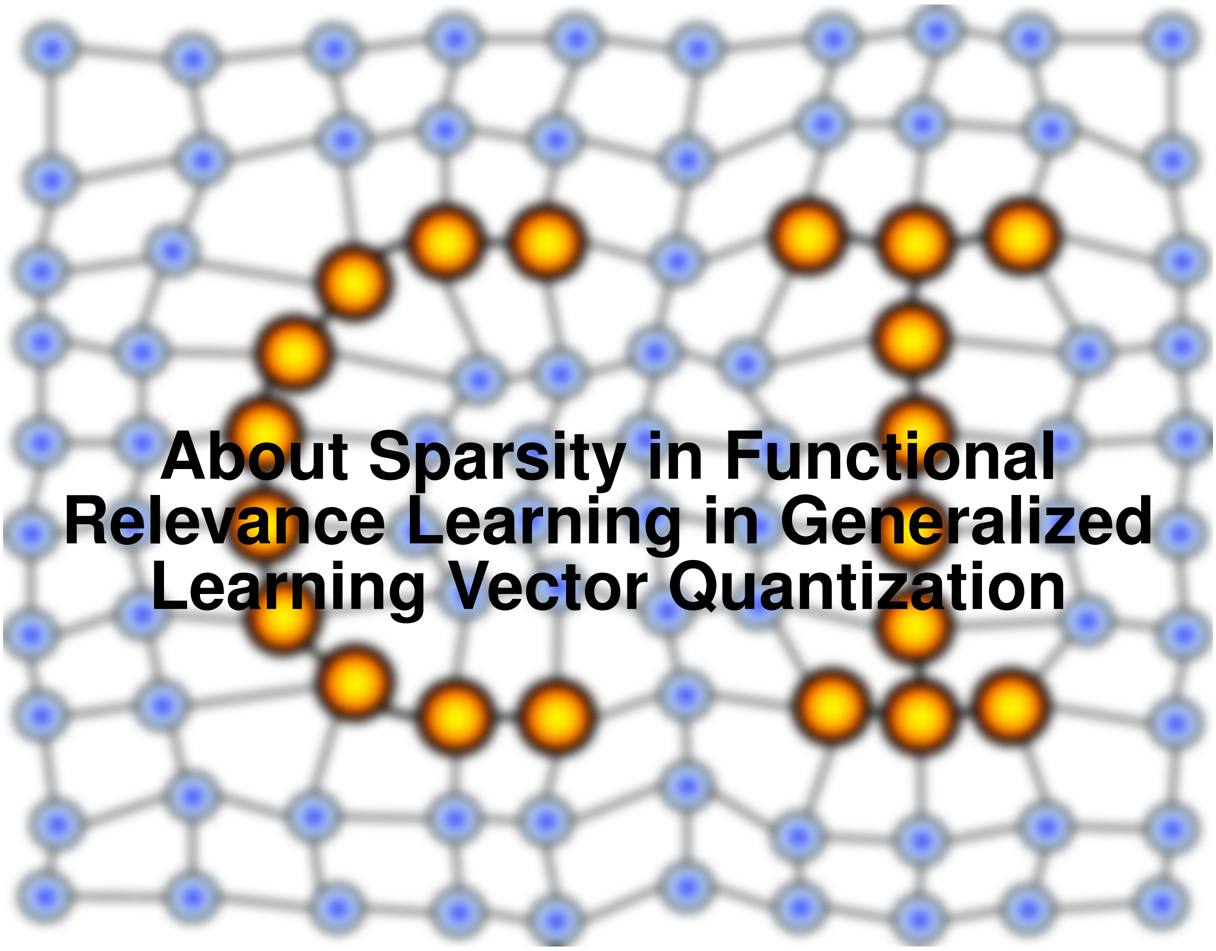
MACHINE LEARNING REPORTS



About Sparsity in Functional Relevance Learning in Generalized Learning Vector Quantization

Report 03/2011
Submitted: 04.05.2011
Published:16.05.2011

Marika Kästner[1], Thomas Villmann[1] and M. Biehl[2]
(1) University of Applied Sciences Mittweida,Faculty of Math., Nat.- & CS.,
Computational Intelligence Group, Technikumplatz 17, 09648 Mittweida, Germany
{kaestner,villmann@hs-mittweida.de}
(2) University Groningen Johann Bernoulli Institute for Math. & CS.,
Intelligent Systems Group, P.O. Box 407, 9700 AK Groningen, The Netherlands
{m.biehl@rug.nl}

Machine Learning Reports
http://www.techfak.uni-bielefeld.de/∼fschleif/mlr/mlr.html

Abstract

We propose a functional approach to relevance learning and matrix adaptation for learning vector quantization of high-dimensional functional data. We show how parametrization of the functional relevance profile or functional matrix learning can be established for a reasonable number of adaptive parameters. In particular we emphasize model sparsity in terms of structural sparsity and feature selection.

Keywords: functional vector quantization, relevance learning, matrix learning, information theory, feature selection

1 Introduction

During the last years prototype based models became one of the widely used paradigms for clustering and classification. Different strategies have been proposed in classification: Whereas support vector machines (SVMs) emphasize the class borders by the support vectors while maximizing the separation margin, the family of learning vector quantization (LVQ) algorithms is motivated by class representative prototypes and decision margin optimization to achieve high classification accuracy [2]. Based on the original but heuristically motivated standard LVQ introduced by KOHONEN [7] several more advanced methods were proposed. One key approach is the generalized LVQ (GLVQ) suggested by SATO&YAMADA [11] approximating the accuracy by a differentiable cost function to be minimized by stochastic gradient descent. This algorithm was extended to deal with metric adaptation to weight the data dimensions according to their relevance for classification [4]. Usually, this relevance learning is based on weighting the Euclidean distance, and, hence, the data dimensions are treated independently leading to large number of weighting coefficients, the so-called relevance profile, to be adapted in case of high-dimensional data. An extension of this approach is matrix learning where a parametric quadratic form of the distance is used [13].

If the data dimension is very large, as it is frequently the case for spectral data or time series, the relevance determination and the parameter adaptation may become infeasible or numerically instable. However, functional data have in common that the vectors can be seen as discrete realizations of functions. For this kind of data the index of the vector dimensions is a representative of the respective independent function variable, i.e. frequency, time or position etc. In this sense the data dimensions are certainly not uncorrelated or independent.

The aim of the new relevance and matrix learning methods proposed here is to exploit this property. We will interprete the relevance profile as well as a discrete representation of an one-dimensional relevance function. For the parameters of the quadratic form in matrix learning a two-dimensional function description is assumed. We suggest to approximate these functions as a superposition of only a few basis functions depending on a drastically decreased number of parameters compared to the huge number of independent weights or matrix elements in the original formulation of relevance learning. We call the resulting algorithms *Generalized Functional Relevance LVQ* (GFRLVQ) and *Generalized Functional Matrix LVQ* (GFMLVQ). Further, we propose the integration of a sparseness criterion for minimizing the number of basis functions based on an entropy criterion resulting in *Sparse GFRLVQ* (S-GFRLVQ) and *Sparse GFMLVQ* (S-GFMLVQ).

2 Relevance and Matrix Learning in GLVQ – GRLVQ

As mentioned before, GLVQ is an extension of standard LVQ based on energy function E approximating the accuracy. Given a set $V \subseteq \mathbb{R}^D$ of data vectors $\mathbf{v}$ with class labels $x_{\mathbf{v}} \in \mathcal{C} = \{1, 2, \ldots C\}$, the prototypes $\mathbf{w} \in W \subset \mathbb{R}^D$ with class labels y_j $(j = 1, \ldots, N)$ should be distributed in such a way that they represent the data classes as accurate as possible. In particular, the following cost function is minimized

$$E(W) = \frac{1}{2}\sum_{\mathbf{v}\in V} f(\mu(\mathbf{v})) \text{ with } \mu(\mathbf{v}) = \frac{d^{+}(\mathbf{v}) - d^{-}(\mathbf{v})}{d^{+}(\mathbf{v}) + d^{-}(\mathbf{v})} \tag{1}$$

where f is a monotonically increasing function usually chosen as sigmoidal or the identity function. The function $\mu(\mathbf{v})$ is the classifier function where $d^{+}(\mathbf{v}) = d(\mathbf{v}, \mathbf{w}^{+})$ denotes the distance between the data vector $\mathbf{v}$ and the closest prototype $\mathbf{w}^{+}$ with the same class label $y_{\mathbf{w}^{+}} = x_{v}$, and $d^{-}(\mathbf{v}) = d(\mathbf{v}, \mathbf{w}^{-})$ is the distance to the best matching prototype $\mathbf{w}^{-}$ with a class label $y_{\mathbf{w}^{-}}$ different from $x_{\mathbf{v}}$. The similarity measure $d(\mathbf{v}, \mathbf{w})$ is supposed differentiable with respect to the second argument but not necessarily to be a mathematical distance. More general similarity measures could be considered. Possible choices are the standard Euclidean distance or their weighted counterpart

$$d_{\lambda}(\mathbf{v}, \mathbf{w}) = \sum_{i=1}^{D} \lambda_{i} (v_{i} - w_{i})^{2} \tag{2}$$

with relevance weights $\lambda_{i} \geq 0$ and $\sum_{i} \lambda_{i} = 1$. The vector λ is called relevance profile.

Learning in GLVQ of $\mathbf{w}^{+}$ and $\mathbf{w}^{-}$ is done by stochastic gradient descent with respect to the cost function $E(W)$ according to

$$\frac{\partial_{S} E(W)}{\partial \mathbf{w}^{+}} = \xi^{+} \cdot \frac{\partial d^{+}}{\partial \mathbf{w}^{+}} \text{ and } \frac{\partial_{S} E(W)}{\partial \mathbf{w}^{-}} = \xi^{-} \cdot \frac{\partial d^{-}}{\partial \mathbf{w}^{-}}$$

with $\xi^{+} = f' \cdot \frac{2 \cdot d^{-}(\mathbf{v})}{(d^{+}(\mathbf{v}) + d^{-}(\mathbf{v}))^{2}}$ and $\xi^{-} = -f' \cdot \frac{2 \cdot d^{+}(\mathbf{v})}{(d^{+}(\mathbf{v}) + d^{-}(\mathbf{v}))^{2}}$. Relevance learning in this model can be performed by adaptation of the relevance, weights again by gradient descent:

$$\frac{\partial E_{S}(W)}{\partial \lambda_{j}} = \xi^{+} \cdot \frac{\partial d_{\lambda}^{+}}{\partial \lambda_{j}} + \xi^{-} \cdot \frac{\partial d_{\lambda}^{-}}{\partial \lambda_{j}} . \tag{3}$$

The respective algorithm is named Generalized Relevance LVQ – GRLVQ [4], which aims at the optimization of the decision margin and therefore is comparable to support vector machines (SVM) [3]. Yet, in this model the relevance weights as well as the vector components are treated independently as it seems natural in the Euclidean distance or its weighted variant.

Matrix learning generalizes the idea of relevance learning [14, 13]. Instead of the weighted Euclidean distance (2), a positive definite bilinear form is used:

$$d_{\mathbf{\Lambda}}(\mathbf{v}, \mathbf{w}) = (\mathbf{v} - \mathbf{w})^{T} \mathbf{\Lambda} (\mathbf{v} - \mathbf{w}) \tag{4}$$

with a quadratic, positive semi-definite matrix $\mathbf{\Lambda}$. Using the fact that each matrix $\mathbf{\Lambda}$ can be decomposed into

$$\mathbf{\Lambda} = \mathbf{\Omega}^{T} \mathbf{\Omega} , \tag{5}$$

where $\mathbf{\Omega} \in \mathbb{R}^{D \times m}$ and $m > 0$ an arbitrary positive integer [1], the distance (4) can be rewritten as

$$d_{\mathbf{\Lambda}}(\mathbf{v}, \mathbf{w}) = (\mathbf{\Omega}(\mathbf{v} - \mathbf{w}))^{2} \tag{6}$$

In analogy to relevance learning, we get

$$\frac{\partial_{S} E(W)}{\partial \Omega_{ij}} = \xi^{+} \cdot \frac{\partial d_{\mathbf{\Lambda}}^{+}}{\partial \Omega_{ij}} + \xi^{-} \cdot \frac{\partial d_{\mathbf{\Lambda}}^{-}}{\partial \Omega_{ij}} \tag{7}$$

for the matrix learning vector quantization algorithm (GMLVQ).

3 Functional Relevance and Matrix Learning for GLVQ

As we have seen, the data dimensions are handled independently according to their sequence in both, GRLVQ and GMLVQ. This leads to a huge number of relevance weights to be adjusted, if the data vector are really high-dimensional as it is the case in many applications. For example, processing of hyperspectral data frequently requires the consideration of hundreds or thousands of spectral bands; time series may consist of a huge number of time steps. This huge dimensionality may lead to instable behavior of relevance learning in GRLVQ. For GMLVQ the number of free parameters scales with the square of the number of input dimensions although a self-regularizing mechanism leads to the fact that the effective number of free parameters is linear as in GRLVQ [12].

Yet, if the data vector are discrete representations of functions, both relevance and matrix learning can make use of this functional property to reduce the number of parameters in relevance learning. More precisely, we assume in the following that data vectors $\mathbf{v} = (v_1, \ldots, v_D)^T$ are representations of functions $v(t)$ with given values $v_i = v(t_i)$.

3.1 Functional Relevance Learning

In *functional relevance learning* the relevance profile is interpreted as a function $\lambda(t)$ with $\lambda_j = \lambda(t_j)$, too. In the recently proposed *generalized functional relevance* LVQ (GFRLVQ) [5], the relevance function $\lambda(t)$ is supposed to be a superposition

$$\lambda(t) = \sum_{l=1}^{K} \beta_l \mathcal{K}_l(t) \tag{8}$$

of simple basis functions $\mathcal{K}_l$ depending on only a few parameters with the restriction $\sum_{l=1}^{K} \beta_l = 1$. Famous examples are standard Gaussians or Lorentzians:

$$\mathcal{K}_l(t) = \frac{1}{\sigma_l \sqrt{2\pi}} \exp\left(-\frac{(t-\Theta_l)^2}{2\sigma_l^2}\right) \tag{9}$$

and

$$\mathcal{K}_l(t) = \frac{1}{\eta_l \pi} \frac{\eta_l^2}{\eta_l^2 + (t-\Theta_l)^2}\ , \tag{10}$$

respectively. Now, relevance learning takes place by adaptation of the parameters β_l, Θ_l,σ_l and η_l, respectively. For this purpose, again a stochastic gradient scheme is applied. For an arbitrary parameter ϑ_l of the dissimilarity measure d we have

$$\frac{\partial_S E}{\partial \vartheta_l} = \xi^+ \cdot \frac{\partial d^+}{\partial \vartheta_l} + \xi^- \cdot \frac{\partial d^-}{\partial \vartheta_l}$$

Using the convention $t_j = j$ we get in the case of Gaussians for the weighting coefficient β_l, the center Θ_l and the width σ_l for

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \beta_l} = \frac{1}{\sigma_l\sqrt{2\pi}}\sum_{j=1}^{D}\exp\left(-\frac{(j-\Theta_l)^2}{2\sigma_l^2}\right)(v_j-w_j)^2 \tag{11}$$

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \Theta_l} = \frac{\beta_l}{\sigma_l^3\sqrt{2\pi}}\sum_{j=1}^{D}(j-\Theta_l)\exp\left(-\frac{(j-\Theta_l)^2}{2\sigma_l^2}\right)(v_j-w_j)^2 \tag{12}$$

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \sigma_l} = \frac{\beta_l}{\sigma_l^2\sqrt{2\pi}}\sum_{j=1}^{D}\left(\frac{(j-\Theta_l)^2}{\sigma_l^2}-1\right)\exp\left(-\frac{(j-\Theta_l)^2}{2\sigma_l^2}\right)(v_j-w_j)^2 \tag{13}$$

whereas for the Lorentzian we obtain

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \beta_l} = \frac{1}{\pi}\sum_{j=1}^{D}\frac{\eta_l}{\eta_l^2+(j-\Theta_l)^2}(v_j-w_j)^2 \tag{14}$$

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \Theta_l} = \frac{\beta_l}{\pi}\sum_{j=1}^{D}\frac{2\eta_l(j-\Theta_l)}{\left(\eta_l^2+(j-\Theta_l)^2\right)^2}(v_j-w_j)^2 \tag{15}$$

$$\frac{\partial d(\mathbf{v},\mathbf{w})}{\partial \eta_l} = \frac{\beta_l}{\pi}\sum_{j=1}^{D}\frac{(j-\Theta_l)^2-\eta_l^2}{\left(\eta_l^2+(j-\Theta_l)^2\right)^2}(v_j-w_j)^2 \tag{16}$$

Instabilities may occur if the center locations Θ_l, Θ_k become very similar for $l \neq k$. To avoid this phenomenon a weighted penalty term

$$P_R = \sum_{l=1}^{K}\sum_{\substack{m=1\\ m\neq l}}^{K}\exp\left(-\frac{(\Theta_m-\Theta_l)^2}{2\xi_l\xi_m}\right) \tag{17}$$

is added to the cost function (1) according to the used basis functions. The resulting new cost function is

$$E_{GFRLVQ} = E(W) + \varepsilon_R P_R \tag{18}$$

with a properly chosen penalty weight $\varepsilon_R > 0$. For Gaussian basis functions we set $\xi_k = \sigma_k$, and for the Lorentzians we take $\xi_k = \eta_k$. The penalty can be interpreted as a repulsion with an influence range determined by the local correlations $\xi_l\xi_m$. The resulting additional update term for Θ_l-learning is

$$\frac{\partial P_R}{\partial \Theta_l} = \frac{1}{2}\sum_{m=1}^{K}\frac{(\Theta_l-\Theta_m)}{\xi_l\xi_m}\exp\left(-\frac{(\Theta_m-\Theta_l)^2}{2\xi_l\xi_m}\right)$$

leading to a minimum spreading of the basis function centers Θ_l. Analogously, an additional term occurs for the adjustments of the ξ_l according to $\frac{\partial P_R}{\partial \xi_l}$, which has to be taken into account for the update of σ_k and η_k for Gaussians and Lorentzians, respectively.

3.2 Functional Matrix Learning

For *Functional Matrix Learning Vector Quantization* (GFMLVQ) we assume in complete analogy to the functional relevance learning approach that the matrix $\mathbf{\Omega}$ involved in (6) by the decomposition (5) is described in terms of a superposition

$$\mathbf{\Omega}\left(t_1,t_2\right)=\sum_{l=1}^{K}\beta_l\mathbf{K}_l\left(t_1,t_2\right) \tag{19}$$

of two-dimensional basis functions $\mathbf{K}_l\left(t_1,t_2\right)$, i.e. we have

$$\mathbf{\Lambda}\left(t_1,t_2\right)=\int\mathbf{\Omega}\left(t_1,t\right)\mathbf{\Omega}\left(t_2,t\right)dt$$

and, therefore,

$$\mathbf{\Lambda}\left(t_1,t_2\right)=\sum_{l=1}^{K}\sum_{m=1}^{K}\beta_l\beta_m\cdot\int\mathbf{K}_l\left(t_1,t\right)\cdot\mathbf{K}_m\left(t_2,t\right)dt\;. \tag{20}$$

The basis functions $\mathbf{K}_l\left(t_1,t_2\right)$ are now two-dimensional. For the Gaussian example we have

$$\mathbf{K}_l\left(t_1,t_2\right)=\frac{1}{\sigma_{1,l}\cdot\sigma_{2,l}\cdot 2\pi}\exp\left(-\left(\frac{\left(t_1-\Theta_{1,l}\right)^2}{2\sigma_{1,l}^2}+\frac{\left(t_2-\Theta_{2,l}\right)^2}{2\sigma_{2,l}^2}\right)\right) \tag{21}$$

whereas for the Lorentzian we get

$$\mathbf{K}_l\left(t_1,t_2\right)=\frac{1}{\eta_{1,l}\cdot\eta_{2,l}\cdot\pi^2}\left(\frac{\eta_{1,l}^2}{\eta_{1,l}^2+\left(t_1-\Theta_{1,l}\right)^2}\cdot\frac{\eta_{2,l}^2}{\eta_{2,l}^2+\left(t_2-\Theta_{2,l}\right)^2}\right) \tag{22}$$

and the derivatives have to be performed accordingly.

The penalty term (17) known from GFRLVQ avoiding there the total overlap of different basis functions $\mathbf{K}_l$ and $\mathbf{K}_k$ for $k\neq l$ has also to be adapted and reads now as

$$P_M=\sum_{l=1}^{K}\sum_{m=1}^{K}\exp\left(-\left(\frac{\left(\Theta_{1,m}-\Theta_{1,l}\right)^2}{2\xi_{1,m}\xi_{1,l}}+\frac{\left(\Theta_{2,m}-\Theta_{2,l}\right)^2}{2\xi_{2,m}\xi_{2,l}}\right)\right) \tag{23}$$

again with the settings $\xi_{i,k}=\sigma_{i,k}$ and $\xi_{i,k}=\eta_{i,k}$ for Gaussians and Lorentzians, respectively. Thus the full cost function

$$E_{GFMLVQ}=E\left(W\right)+\varepsilon_M P_M \tag{24}$$

is finally obtained for GFMLVQ with the penalty weight $\varepsilon_M>0$.

4 Sparse GFRLVQ and GFMLVQ

We have to distinguish at least two different kinds of sparsity. The first one is *structural sparsity* emphasizing the sparsity of the generative model of the relevance profile with respect to the selection of basis functions. The second one we call *feature sparsity* reflecting the sparsity in terms of data dimensions, which are taken into account for classification.

4.1 Structural Sparsity

In the GFRLVQ model the number K of basis functions to be used can be chosen freely so far. Obviously, if K is too small, an appropriate relevance weighting is impossible. Otherwise, a value of K too large complicates the problem more than necessary. Hence, a good adjustment is demanded. This problem can be seen as a structural sparseness requirement in functional relevance learning model.

A suitable methodology to judge sparsity is information theory. In particular, the Shannon entropy H of the weighting coefficients $\beta = (\beta_1, \ldots, \beta_K)$ can be applied to quantify structural sparsity. Maximum sparseness, i.e. minimum entropy, is obtained, iff $\beta_l = 1$ for exactly one certain l whereas the other β_m are equal to zero. However, maximum sparseness may be accompanied by a decrease of accuracy in classification and/or increased cost function value E_{GFRLVQ}.

To achieve an optimal balancing, we propose the following strategy: The cost function E_{GFRLVQ} is extended to

$$E_{S-GFRLVQ} = E_{GFRLVQ} + \gamma(\tau) \cdot H(\beta) \tag{25}$$

with τ counting the adaptation steps. Let τ_0 be the final time step of the usual GFRLVQ-learning. Then $\gamma(\tau) = 0$ for $\tau < \tau_0$ holds. Thereafter, $\gamma(\tau)$ is slowly increased in an adiabatic manner [6], such that all parameters can immediately follow the drift of the system. An additional term for β_l-adaptation occurs for non-vanishing $\gamma(\tau)$-values according to this new cost function (25):

$$\frac{\partial E_{S-GFRLVQ}}{\partial \beta_l} = \frac{\partial E_{GFRLVQ}}{\partial \beta_l} + \gamma(\tau) \frac{\partial H}{\partial \beta_l} \tag{26}$$

with $\frac{\partial H}{\partial \beta_l} = -(\log(\beta_l) + 1)$. This term triggers the β-vector to become sparse. The adaptation process is stopped if the E_{GFRLVQ}-value or the classification error shows a significant increase compared to the time τ_0.

Obviously, this optimization scheme can also be applied to GFMLVQ yielding *Sparse* GFMLVQ (S-GFMLVQ) with

$$E_{S-GFMLVQ} = E_{GFMLVQ} + \gamma(\tau) \cdot H(\beta) \tag{27}$$

as cost function.

4.2 Feature Sparsity

A different sparsity requirement concerns the contribution of data dimensions to the classification decision. In GFRLVQ this feature selection can be controlled by an entropy term

$$H_F(\lambda) = -\int \lambda(t) \ln(\lambda(t))\, dt \tag{28}$$

enforcing the sparsity in the relevance profile $\lambda(t)$. According to the functional profile model (8) with the basis functions $\mathcal{K}_l(t)$ we have

$$H_F(\lambda) = -\int \left(\sum_{l=1}^{K} \beta_l \mathcal{K}_l(t)\right) \ln\left(\sum_{l=1}^{K} \beta_l \mathcal{K}_l(t)\right) dt \tag{29}$$

Considering the derivatives $\frac{\partial H_F(\lambda)}{\partial \beta_j}$, $\frac{\partial H_F(\lambda)}{\partial \Theta_j}$ and $\frac{\partial H_F(\lambda)}{\partial \sigma_j}$ we get

$$\frac{\partial H_F(\lambda)}{\partial \beta_j} = -\int \mathcal{K}_j(t)\left[1+\ln\left(\sum_{l=1}^{K}\beta_l \mathcal{K}_l(t)\right)\right]dt\ , \tag{30}$$

$$\frac{\partial H_F(\lambda)}{\partial \Theta_j} = -\int \beta_j \frac{\partial \mathcal{K}_j(t)}{\partial \Theta_j}\left[1+\ln\left(\sum_{l=1}^{K}\beta_l \mathcal{K}_l(t)\right)\right]dt\ , \tag{31}$$

and

$$\frac{\partial H_F(\lambda)}{\partial \sigma_j} = -\int \beta_j \frac{\partial \mathcal{K}_j(t)}{\partial \sigma_j}\left[1+\ln\left(\sum_{l=1}^{K}\beta_l \mathcal{K}_l(t)\right)\right]dt\ , \tag{32}$$

respectively.

Feature sparsity in the matrix version GMRLVQ can be enforced by the maximization of the entropy of the diagonal elements of $\mathbf{\Lambda}$ in the matrix distance (4): Vanishing diagonal elements of $\mathbf{\Lambda}$ imply by use of the decomposition $\mathbf{\Lambda} = \mathbf{\Omega}^T\mathbf{\Omega}$ (5) that the respective columns of $\mathbf{\Omega}$ in the rewritten distance (6) can be neglected. Transferring this idea to functional matrix relevance GFMLVQ (20) we write the entropy term in complete analogy as

$$H_F(\mathbf{\Lambda}) = -\int \mathbf{\Lambda}(\tau,\tau)\ln\left(\mathbf{\Lambda}(\tau,\tau)\right)d\tau\ . \tag{33}$$

with

$$\mathbf{\Lambda}(\tau,\tau) = \sum_{l=1}^{K}\sum_{m=1}^{K}\beta_l\beta_m \cdot \int \mathbf{K}_l(\tau,t)\,\mathbf{K}_m(\tau,t)\,dt \tag{34}$$

and $\mathbf{K}_l(\tau,t)$ are the underlying two-dimensional basis functions of the functional model (19). Triggering the feature sparseness is realized again by application of the derivatives $\frac{\partial H_F(\mathbf{\Lambda})}{\partial \beta_j}$, $\frac{\partial H_F(\mathbf{\Lambda})}{\partial \sigma_{1,j}}$, $\frac{\partial H_F(\mathbf{\Lambda})}{\partial \sigma_{2,j}}$, $\frac{\partial H_F(\mathbf{\Lambda})}{\partial \Theta_{1,j}}$, and $\frac{\partial H_F(\mathbf{\Lambda})}{\partial \Theta_{2,j}}$ in learning. For these we get

$$\begin{aligned}\frac{\partial H_F(\mathbf{\Lambda})}{\partial \beta_j} &= -\int\left(2\sum_{l=1}^{K}\beta_l \cdot \int \mathbf{K}_l(\tau,t)\,\mathbf{K}_j(\tau,t)\,dt\right)\\ &\quad \cdot\left[1+\ln\left(\sum_{l=1}^{K}\sum_{m=1}^{K}\beta_l\beta_m \cdot \int \mathbf{K}_l(\tau,t)\,\mathbf{K}_m(\tau,t)\,dt\right)\right]d\tau\end{aligned} \tag{35}$$

and

$$\begin{aligned}\frac{\partial H_F(\mathbf{\Lambda})}{\partial \xi_j} &= -\int\left(2\sum_{l=1}^{K}\beta_l\beta_j \cdot \int \mathbf{K}_l(\tau,t)\,\frac{\partial \mathbf{K}_j(\tau,t)}{\partial \xi_j}dt\right)\\ &\quad \cdot\left[1+\ln\left(\sum_{l=1}^{K}\sum_{m=1}^{K}\beta_l\beta_m \cdot \int \mathbf{K}_l(\tau,t)\,\mathbf{K}_m(\tau,t)\,dt\right)\right]d\tau\end{aligned} \tag{36}$$

where ξ_j stands for any of these variables $\sigma_{1,j}$, $\sigma_{2,j}$, $\Theta_{1,j}$, $\Theta_{2,j}$.

5 Conclusion

In this paper we propose the *functional* relevance and matrix learning for generalized learning vector quantization. Functional learning supposes that the data vectors are representations of functions such that the relevance profile or the parameter matrix can be written as a superposition of one- or two-dimensional basis functions, respectively. These basis functions depend on only a few parameters to be adapted during learning compared to the huge number of free parameters to be adjusted in usual relevance or matrix learning. To obtain an optimal number of basis functions for the superposition a sparsity constraint is suggested. There, sparsity is judged in terms of the entropy of the respective sparsity model: structural sparsity prunes the superposition of the basis functions wheras feature sparsity leads to the use of a reduced number of input dimensions.

The approach is here exemplified for the weighted Euclidean distance and a bilinear form also based on the Euclidean norm, for simplicity. Obviously, the Euclidean distance is not based on a functional norm. Yet, the transfer to real functional norms and distances like Sobolev norms [17], the Lee-norm [8, 9], kernel based LVQ-approaches [16] or divergence based similarity measures [15],[10], which carry the functional aspect inherently, is straightforward and topic of future investigations.

Obviously, the functional matrix approach can also be applied for matrices Λ of limited rank, i.e. rectangular matrices. This leads to a functional version limited rank matrix LVQ (LiRaMLVQ), which is proposed in [1].

References

[1] K. Bunte, B. Hammer, A. Wismüller, and M. Biehl. Adaptive local dissimilarity measures for discriminative dimension reduction of labeled data. *Neurocomputing*, 73:1074–1092, 2010.

[2] B. Hammer, M. Strickert, and T. Villmann. Relevance LVQ versus SVM. In L. Rutkowski, J. Siekmann, R. Tadeusiewicz, and L. Zadeh, editors, *Artificial Intelligence and Soft Computing (ICAISC 2004)*, Lecture Notes in Artificial Intelligence 3070, pages 592–597. Springer Verlag, Berlin-Heidelberg, 2004.

[3] B. Hammer, M. Strickert, and T. Villmann. On the generalization ability of GRLVQ networks. *Neural Processing Letters*, 21(2):109–120, 2005.

[4] B. Hammer and T. Villmann. Generalized relevance learning vector quantization. *Neural Networks*, 15(8-9):1059–1068, 2002.

[5] M. Kästner, B. Hammer, and T. Villmann. Generalized functional relevance learning vector quantization. In M. Verleysen, editor, *Proc. of European Symposium on Artificial Neural Networks (ESANN'2011)*, pages 93–98, Evere, Belgium, 2011. d-side publications.

[6] T. Kato. On the adiabatic theorem of quantum mechanics. *Journal of the Physical Society of Japan*, 5(6):435–439, 1950.

[7] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 1995. (Second Extended Edition 1997).

[8] J. Lee and M. Verleysen. Generalization of the l_p norm for time series and its application to self-organizing maps. In M. Cottrell, editor, *Proc. of Workshop on Self-Organizing Maps (WSOM) 2005*, pages 733–740, Paris, Sorbonne, 2005.

[9] J. Lee and M. Verleysen. *Nonlinear Dimensionality Reduction*. Information Sciences and Statistics. Springer Science+Business Media, New York, 2007.

[10] E. Mwebaze, P. Schneider, F.-M. Schleif, J. Aduwo, J. Quinn, S. Haase, T. Villmann, and M. Biehl. Divergence based classification in learning vector quantization. *Neurocomputing*, 74(9):1429–1435, 2011.

[11] A. Sato and K. Yamada. Generalized learning vector quantization. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference*, pages 423–9. MIT Press, Cambridge, MA, USA, 1996.

[12] P. Schneider, K. Bunte, H. Stiekema, B. Hammer, T. Villmann, and M. Biehl. Regularization in matrix relevanz learning. *IEEE Transactions on Neural Networks*, 21(5):831–840, 2010.

[13] P. Schneider, B. Hammer, and M. Biehl. Adaptive relevance matrices in learning vector quantization. *Neural Computation*, 21:3532–3561, 2009.

[14] P. Schneider, B. Hammer, and M. Biehl. Distance learning in discriminative vector quantization. *Neural Computation*, 21:2942–2969, 2009.

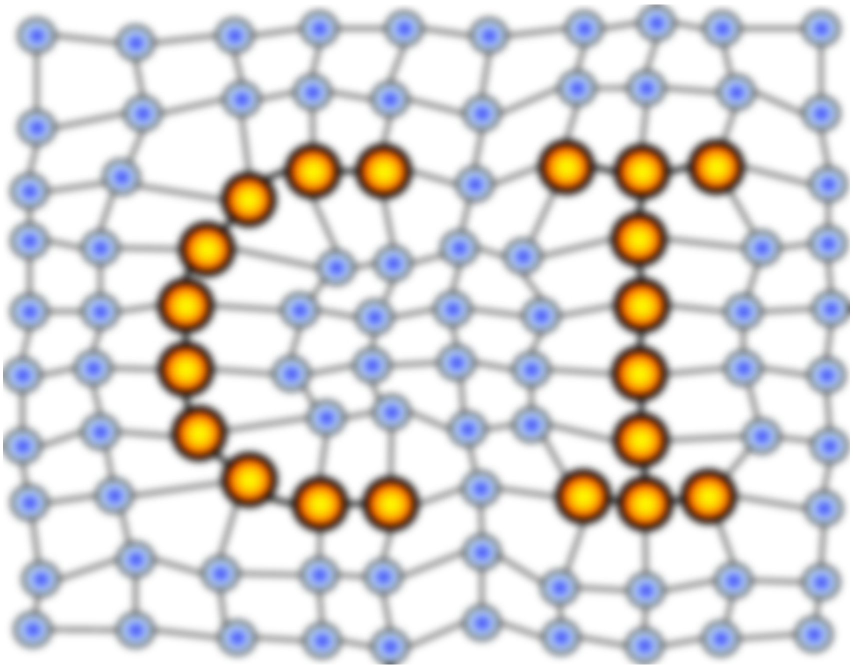
[15] T. Villmann and S. Haase. Divergence based vector quantization. *Neural Computation*, 23(5):1343–1392, 2011.

[16] T. Villmann and B. Hammer. Theoretical aspects of kernel GLVQ with differentiable kernel. *IfI Technical Report Series*, (IfI-09-12):133–141, 2009.

[17] T. Villmann and F.-M. Schleif. Functional vector quantization by neural maps. In J. Chanussot, editor, *Proceedings of First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS 2009)*, pages 1–4. IEEE Press, 2009. ISBN 978-1-4244-4948-4.

Machine Learning Reports

Report 03/2011



Impressum

Machine Learning Reports ISSN: 1865-3960

▽ Publisher/Editors

Prof. Dr. rer. nat. Thomas Villmann
University of Applied Sciences Mittweida
Technikumplatz 17, 09648 Mittweida, Germany
• http://www.mni.hs-mittweida.de/

Dr. rer. nat. Frank-Michael Schleif
University of Bielefeld
Universitätsstrasse 21-23, 33615 Bielefeld, Germany
• http://www.cit-ec.de/tcs/about

▽ Acknowledgments

We would like to thank the reviewers for their time and patience.

Machine Learning Reports
http://www.techfak.uni-bielefeld.de/∼fschleif/mlr/mlr.html