Large Margin Linear Discriminative Visualization by Matrix Relevance Learning

Michael Biehl*, Kerstin Bunte †, Frank-Michael Schleif †,
Petra Schneider ‡ and Thomas Villmann §
* University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science
P.O. Box 407, NL–9700 AK Groningen, The Netherlands, Email: m.biehl@rug.nl
† University of Bielefeld, Faculty of Technology, CITEC
D–33594 Bielefeld, Germany
‡ School of Clinical and Experimental Medicine, Centre for Endocrinology and Metabolism CEDAM
University of Birmingham, Birmingham B15 2TT, United Kingdom
§ University of Applied Sciences Mittweida, Fakultät für Mathematik, Physik und Informatik
Technikumplatz 17, D–09648 Mittweida, Germany

***Abstract*—We suggest and investigate the use of Generalized Matrix Relevance Learning (GMLVQ) in the context of discriminative visualization. This prototype-based, supervised learning scheme parameterizes an adaptive distance measure in terms of a matrix of relevance factors. By means of a few benchmark problems, we demonstrate that the training process yields low rank matrices which can be used efficiently for the discriminative visualization of labeled data. Comparison with well known standard methods illustrate the flexibility and discriminative power of the novel approach. The mathematical analysis of GMLVQ shows that the corresponding stationarity condition can be formulated as an eigenvalue problem with one or several strongly dominating eigenvectors. We also study the inclusion of a penalty term which enforces non-singularity of the relevance matrix and can be used to control the role of higher order eigenvalues, efficiently.**

I. Introduction

Given the ever increasing amount of large, high-dimensional data sets acquired in a variety scientific disciplines and application domains, efficient methods for dimension reduction and visualization play an essential role in modern data processing and analysis.

A multitude of methods for the low-dimensional representation of complex data sets has been proposed in recent years, see for instance [1] for an overview and categorization of the many different approaches. The diversity of methods reflects the large number of goals and criteria one can have in mind with respect to dimension reduction. Indeed, in particular for visualization, one of the key problems of the field seems to be the formulation of clear-cut objectives.

A somewhat special case is the discriminative visualization of labeled data as it occurs in classification problems or other supervised machine learning frameworks. The classification accuracy which can be achieved in the low-dimensional space provides at least one obvious guideline for the evaluation and comparison of visualizations.

In this contribution we restrict ourselves to methods which perform an explicit mapping from the original feature space to lower dimension [1], [2]. Moreover, we will consider only linear methods. While limited in power and flexibility, linear methods continue to be attractive due to their interpretability, low computational costs, and accessibility for mathematical analysis.

Probably the most prominent methods that employ linear projections are the well known Prinicipal Component Analysis for the unsupervised analysis of data sets and Linear Discriminant Analysis in the context of classification problems.

We present a novel, linear approach to the low-dim. representation and visualization of labeled data which is based on a particularly powerful and flexible framework for classification. In the recently introduced Generalized Matrix Relevance LVQ [3], a set of prototypes is identified as typical representatives of the classes. At the same time an adaptive distance measure is determined. The latter is parameterized by a matrix which corresponds to a linear transformation of feature space. The optimization of, both, prototypes and relevance matrix is guided by a margin based cost function. The method displays an intrinsic tendency to yield a low rank relevance matrix and, hence, its eigenvectors can be employed for discriminative low-dimensional representation and visualization.

The fact that the approach combines prototype based classification with linear low-dimensional representations makes it a particularly promising technique for interactive tasks. Prototypes serve as typical representatives of the classes and facilitate good interpretability of the classifier. This is clearly benefitial in discussions with domain experts. In the context of visualization, it offers the possibility to *zoom in* on regions of feature space which are most representative for the classes. User feedback or data driven, adaptive distance measures can also be readily employed in the context of interactive applications. They can be used, for instance, in the similarity based retrieval of images from a large data base, see [4] for an example in the medical domain. The combination of prototype based classification, adaptive similarity measures, and discriminative visualization clearly bears the promise to facilitate a number of novel techniques for the interactive analysis of complex data.

We illustrate the GMLVQ approach to discriminative visualization in terms of a few example data sets, comparing with the classical approaches of PCA and LDA. Moreover, we present a theoretical analysis which explains the tendency to low-rank representations in GMLVQ.

II. Linear discriminative visualization

We first introduce Generalized Matrix Relevance Learning as a tool for the discriminative low-dimensional representation of labeled data. In addition we review very briefly two classical statistical methods: Principal Component Analysis and Linear Discriminant Analysis.

A. Generalized Matrix Relevance Learning

Similarity based methods play a most important role in, both, unsupervised and supervised machine learning analysis of complex data sets, see [5] for an overview and further references. In the context of classification problems, Learning Vector Quantization (LVQ), originally suggested by Kohonen [6]–[8], constitutes a particularly intuitive and successful family of algorithms. In LVQ, classes are represented by prototypes which are determined from example data and are defined in the original feature space. Together with a suitable dissimilarity or distance measure they parameterize the classifier, frequently according to a *Nearest Prototype* scheme. LVQ schemes are easy to implement and very flexible. Numerous variations of the original scheme have been suggested, aiming at clearer mathematical foundation, improved performance, or better stability and convergence behavior, see e.g. [9]–[13]. Further references, also reflecting the impressive variety of application domains in which LVQ has been employed successfully, are available at [14].

A key issue in LVQ and other similarity based techniques is the choice of an appropriate distance measure. Most frequently, standard Euclidean or other Minkowski metrics are used without further justification and reflect implicit assumptions about, for instance, the presence of approximately isotropic clusters

Pre-defined distance measures are, frequently, sensitive to rescaling of single features or more general linear transformations of the data. In particular if data is heterogeneous in the sense that features of different nature are combined, usefulness of Euclidean distance based classification is far from obvious.

An elegant framework has been developed which can circumvent this difficulty to a large extent: In so-called Relevance Learning schemes, only the functional form of the dissimilarity is fixed, while a set of parameters is determined in the training process. To our knowledge, this idea was first proposed in [15] in the context of LVQ. Similar ideas have been formulated for other distance based classifiers, see e.g. [16] for an example in the context of Nearest Neighbor classifiers [17].

A generalized quadratic distance is parameterized by a matrix of relevances in Matrix Relevance Learning which is summarized in the following.

1) The adaptive distance measure:
Matrix Relevance LVQ employs a distance measure given by the quadratic form

$$d(\vec{y},\vec{z}) = (\vec{y}-\vec{z})^\top \Lambda\, (\vec{y}-\vec{z}) \quad \text{for} \quad \vec{y},\vec{z}\in\mathbb{R}^N. \tag{1}$$

It is required to fulfill the basic conditions $d(\vec{y},\vec{y}) = 0$ and $d(\vec{y},\vec{z}) = d(\vec{z},\vec{y}) \geq 0$ for all $\vec{y},\vec{z}$ with $\vec{y} \neq \vec{z}$. These are conveniently satisfied by assuming the parameterization $\Lambda = \Omega\Omega^\top$, i.e.

$$d(\vec{y},\vec{z}) = (\vec{y}-\vec{z})^\top \Omega\Omega^\top\, (\vec{y}-\vec{z}) \;=\; \left[\Omega^\top\,(\vec{y}-\vec{z})\right]^2 \tag{2}$$

Hence, $\Omega^\top$ defines a linear mapping of data and prototypes to a space in which standard Euclidean distance is applied.

In the frame of this contribution we only consider a *global metric* which is parameterized by a single matrix Λ for all prototypes. Extensions to locally defined distance measures, i.e. local-linear projections, are discussed in [3], [18].

Note that for a meaningful classification and for the LVQ training it is sufficient to assume that Λ is positive semi-definite; the transformation need not be invertible and could even be represented by a rectangular matrix $\Omega^\top$ [18]. Here we consider unrestricted $N\times N$-matrices Ω without imposing symmetry or other constraints on its structure. In this case, the elements of Ω can be varied independently. For instance, the derivative of the distance measure with respect to an arbitrary element of Ω is

$$\frac{\partial d(\vec{y},\vec{z})}{\partial \Omega_{km}} = 2(y_k - z_k)\left[\Omega^\top(\vec{y}-\vec{z})\right]_m \tag{3}$$

or in matrix notation:

$$\frac{\partial d(\vec{y},\vec{z})}{\partial \Omega} \;=\; =\; 2\,(\vec{y}-\vec{z})\,(\vec{y}-\vec{z})^\top\,\Omega. \tag{4}$$

This derivative is the basis of the matrix adaptation scheme considered in the following.

2) Cost function based training:
A particularly attractive and successful variant of LVQ, termed *Generalized* LVQ (GLVQ) was introduced by Sato and Yamado [10], [11]. Given a set of training examples

$$\{\vec{\xi}^{\,\nu},\sigma^\nu\}_{\nu=1}^P \quad \text{where } \vec{\xi}^{\,\nu}\in\mathbb{R}^N \;\text{ and } \sigma^\nu\in\{1,2,\ldots,n_c\} \tag{5}$$

for an n_c-class problem in N dimensions, training is based on the cost function

$$E \;=\; \frac{1}{P}\sum_{\nu=1}^{P} e(\vec{\xi}^{\,\nu}) \quad \text{with } e(\vec{\xi}) \;=\; \frac{d(\vec{w}_J,\vec{\xi}) - d(\vec{w}_K,\vec{\xi})}{d(\vec{w}_J,\vec{\xi}) + d(\vec{w}_K,\vec{\xi})} \tag{6}$$

Here, the index J identifies the closest prototype which carries the correct label $s_J = \sigma$, the so-called *correct winner* with $d(\vec{w}_J,\vec{\xi}) = \min_k\{d(\vec{w}_k,\vec{\xi})|s_k = \sigma\}$. Correspondingly, the *wrong winner* $\vec{w}_K$ is the prototype with the smallest distance $d(\vec{w}_i,\vec{x})$ among all $\vec{w}_i$ representing a different class $s_i \neq \sigma$. Frequently, an additional sigmoidal function $\Phi(e)$ is applied [10]. While its inclusion would be straightforward, we restrict the discussion to the simplifying case $\Phi(x) = x$, in the following.

Note that $e(\vec{\xi})$ in Eq. (6) is negative if the feature vector is classified correctly. Moreover, $-e(\vec{\xi})$ quantifies the margin of the classification and minimizing E can be interpreted as *large margin* based training of the LVQ system [10], [19]. Matrix relevance learning is incorporated into the framework of GLVQ by inserting the distance measure (2) with adaptive parameters Ω into the cost function (6), see [3].

The popular *stochastic* gradient descent [20], [21] approximates the gradient ∇E by the contribution $\nabla e(\vec{\xi}^{\nu})$ where ν is selected randomly from $\{1, 2 \ldots P\}$ in each step. This variant is frequently used in practice as an alternative to *batch* gradient descent where the sum over all ν is performed [20].

Given a particular example $\vec{\xi}^{\nu}$, the update of prototypes is restricted to the *winners* $\vec{w}_J$ and $\vec{w}_K$:

$$\vec{w}_L \leftarrow \vec{w}_L - \eta_w \frac{\partial e(\vec{\xi}^{\nu})}{\partial \vec{w}_L} \quad \text{for } L = J, K \tag{7}$$

see [3] for details and the full form.

Updates of the matrix Ω are based on single example contributions

$$\frac{\partial e(\vec{\xi}^{\nu})}{\partial \Omega} = \frac{2\, d_K^{\nu}}{\left[d_J^{\nu} + d_K^{\nu}\right]^2} \frac{\partial d_J^{\nu}}{\partial \Omega} - \frac{2\, d_J^{\nu}}{\left[d_J^{\nu} + d_K^{\nu}\right]^2} \frac{\partial d_K^{\nu}}{\partial \Omega} \tag{8}$$

where $d_L^{\nu} = d(\vec{w}_L, \vec{\xi}^{\nu})$, with derivatives as in Eq. (4).

In the stochastic gradient descent procedure the matrix update reads

$$\Omega \leftarrow \Omega - \eta \frac{\partial e(\vec{\xi}^{\nu})}{\partial \Omega}. \tag{9}$$

As demonstrated in Sec. III and shown analytically in Sec. IV, the GMLVQ approach displays a strong tendency to yield singular matrices Λ [3], [18] of very low rank. This effect is advantageous in view of potential over-fitting due to a large number of adaptive parameters. However, the restriction to a single or very few relevant directions can lead to numerical instabilities and might result in inferior classification performance if the distance measure becomes *too simple* [3], [22]. In order to control this behavior a penalty term of the form $-\mu \ln \det \Omega\Omega^{\top}/2$ can be introduced, which is controlled by a Lagrange parameter $\mu > 0$ and prohibits singularity of Λ. The corresponding gradient term [23]

$$\frac{\mu}{2} \frac{\partial \ln \det \Omega\Omega^{\top}}{\partial \Omega} = \mu\, \Omega^{-T} \tag{10}$$

with the shorthand $\Omega^{-T} = \left(\Omega^{-1}\right)^{\top}$, is added to the matrix update, yielding

$$\Omega \leftarrow \Omega - \eta \frac{\partial e(\vec{\xi}^{\nu})}{\partial \Omega} + \eta\, \mu\, \Omega^{-T} \tag{11}$$

in stochastic descent. Note that the extension to rectangular $(N \times M)$-matrices Ω ($M < N$) is also possible [18], [22]: Replacing Ω^{-1} in Eq. (11) by the Moore-Penrose pseudoinverse [23] enforces $\mathrm{rank}(\Lambda) = M$.

In the example applications of GMLVQ presented in the following section, protoytpes were initialized close to the respective class-conditional means, with small random deviations in order to avoid coinciding vectors $\vec{w}_k$. Elements of the initial Ω were drawn independently from a uniform distribution $U(-1, +1)$ with subsequent normalization $\sum_{ij} \Omega_{ij}^2 = 1$.

3) Linear projection of the data set:
As discussed above, the adaptive matrix Ω can be interpreted as to define a linear projection for the intrinsic representation of data and prototypes.

Given a particular matrix Λ, the corresponding Ω is not uniquely defined by Eq. (2). Distance measure and classification performance are, for instance, invariant under rotations or reflections in feature space and $\Lambda = \Omega\Omega^{\top}$ can have many solutions. The actual matrix Ω obtained in the GMVLQ training process will depend on initialization and the randomized sequence of examples, for instance.

Expressing the symmetric Λ in terms of its eigenvectors λ_i and eigenvectors suggests the canonical representation

$$\begin{aligned} \Lambda &= \sum_{i=1}^{N} \lambda_i\, \vec{u}_i\, \vec{u}_i^{\top} = \Omega\Omega^{\top} \quad \text{with} \\ \Omega &= \left[\sqrt{\lambda_1}\vec{u}_1, \sqrt{\lambda_2}\vec{u}_2, \ldots, \sqrt{\lambda_N}\vec{u}_N\right], \end{aligned} \tag{12}$$

in which the rows of $\Omega^{\top}$ are proportional to the eigenvectors of Λ. For a low dimensional representation of the data set, the leading eigenvectors of Λ can be employed, i.e. those corresponding to the largest eigenvalues. The fact that the matrix parameterizes a discriminative distance measure, together with the observation that GMVLQ yields low rank relevance matrices Λ, supports the idea of using this scheme for the visualization of labeled data sets in classification.

B. Linear Discriminant Analysis

We consider Linear Discriminant Analysis (LDA) in the formulation introduced by Fisher [20], [24]–[26]. An alternative approach to LDA is based on Bayes decision theory [25]–[27] and is very similar in spirit. For simplicity we will refer to Fisher's discriminant as LDA, in accordance with imprecise but widespread terminology. Our summary of LDA follows to a large extent the presentation in [20].

Given a data set representing n_c classes, cf. Eq. (5), LDA determines an $(N \times [n_c - 1])$-dim. matrix Γ which defines $(n_c - 1)$ linear projections of the data. It is determined as to maximize an objective function of the form

$$J(\Gamma) = \mathrm{Tr}\left[\left(\Gamma C_w \Gamma^{\top}\right)^{-1} \left(\Gamma C_b \Gamma^{\top}\right)\right]. \tag{13}$$

Here, C_w and C_B are the so-called *within-class* and the *between-class* covariance matrix, respectively:

$$\begin{aligned} C_w &= \sum_{s=1}^{n_c} \sum_{\nu=1}^{P} \delta_{s,\sigma^{\nu}} \left(\vec{\xi}^{\nu} - \vec{m}_s\right)\left(\vec{\xi}^{\nu} - \vec{m}_s\right)^{\top} \\ C_b &= \sum_{s=1}^{n_c} \sum_{\nu=1}^{P} \delta_{s,\sigma^{\nu}} \left(\vec{m}_s - \vec{m}\right)\left(\vec{m}_s - \vec{m}\right)^{\top} \end{aligned} \tag{14}$$

with the Kronecker symbol $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ else. The total mean $\vec{m}$ and the class-conditional means $\vec{m}_s$ are directly

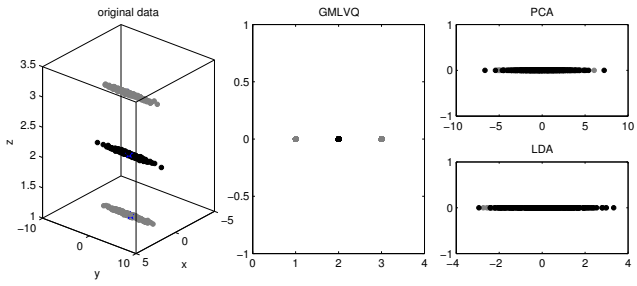

Fig. 1. Projection of the three-dim. artificial two-class data set described in the text, class 1 (2) is represented by grey (black) symbols, respectively. From left to right: original data, projection on the leading eigenvector of Λ in GMLVQ, projection on the first principal component (upper right) and projection by LDA (lower right).

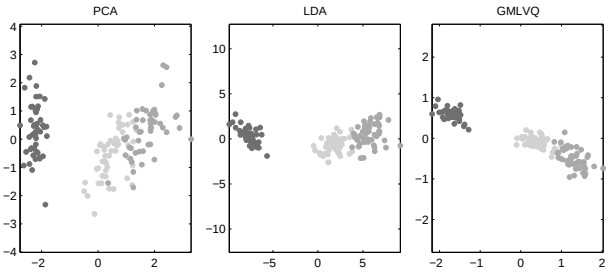

Fig. 2. Two-dimensional visualization of the *Iris* data set as obtained by, from left to right, PCA, LDA, and GMLVQ; see the text for details.

estimated from the given data:

$$\vec{m} = \frac{1}{P}\sum_{\nu=1}^{P}\vec{\xi}^{\nu}, \quad \text{and } \vec{m}_s = \frac{\sum_s \sum_\nu \delta_{\sigma^\nu,s}\,\vec{\xi}^{\nu}}{\sum_s \delta_{\sigma^\nu,s}}. \tag{15}$$

Assuming that the within-class covariance matrix is invertible, one can show that the rows of the optimal Γ correspond to the leading (n_c-1) eigenvectors of $C_w^{-1}C_b$. These can be directly determined from the given data set and, thus, LDA does not require an iterative optimization process.

Note that the between-class covariance matrix C_b is of rank (n_c-1) [20]. Hence, LDA as described above yields a linear mapping to an (n_c-1)-dimensional space. For the purpose of visualization, only the leading eigenvectors of $C_w^{-1}S_b$ are employed. Note that for two-class problems, LDA reduces to the identification of a single direction $\vec{\gamma}$ which maximizes the ratio of within class and between class scatter in terms of the projections $\vec{\xi}^{\nu}\cdot\vec{\gamma}$ [20].

For the results presented in the next section, we have used the implementation of Fisher LDA as it is available in van der Maaten's Toolbox for Dimensionality Reduction [28].

C. Principal Component Analysis

For completeness we also present results obtained by Principal Component Analysis (PCA) [1], [20], [26]. PCA is arguably the most frequently used projection based technique for the exploration and representation of multi-dimensional data sets. Several criteria can be employed as a starting point for deriving PCA, see [1], [20], [26] for examples.

Given a data set of the form (5), PCA determines the eigenvalues and orthonormal eigenvectors of the covariance matrix

$$C = \sum_{\nu=1}^{P}\left(\vec{\xi}^{\nu}-\vec{m}\right)\left(\vec{\xi}^{\nu}-\vec{m}\right)^{\top}. \tag{16}$$

with the total mean $\vec{m}$ given in Eq. (15). The matrix C can also be written as $C = C_w + C_b$, cf. Eq. (14). However, unsupervised PCA does not take class memberships into account at all. Efficient methods for the calculation of the eigenvalue spectrum can be employed in practice. It is, however, very instructive to inspect iterative procedures which relate to Hebbian learning in the context of neural networks [29], [30].

Conveniently, eigenvectors are ordered according to the magnitude of the corresponding eigenvalues. For a normalized eigenvector $\vec{w}_i$ of C with eigenvalue c_i we have

$$c_i = \vec{w}_i^{\top} C\,\vec{w}_i = \sum_{\nu=1}^{P}\left(\vec{w}_i^{\top}\vec{\xi}^{\nu} - \vec{w}_i^{\top}\vec{m}\right)^2.$$

Hence, the eigenvectors mark characteristic directions with variance c_i. The intuitive assumption that a large variance signals high information content of the particular projection can be supported by information theoretic arguments concerning the optimal reconstruction of original vectors from the linear projections [1], [20], [26]. For the purpose of two-dimensional visualization of data sets in the following section, only the two leading eigenvectors were employed.

III. Comparison of methods and illustrative examples

GMLVQ with 1 prototype per class and LDA are, at a glance, very similar in spirit. Clearly, for well separated, nearly isotropic classes represented by single prototypes in their centers one would not expect dramatic differences. However, GLMVQ prototypes are not restricted to positions in the class conditional means which can be advantageous when clusters overlap. More importantly, a different cost function is optimized which appears to be less sensitive to the specific cluster geometry in many cases. In practical applications superior classification performance has been found for GMLVQ, even if two classes are represented by single prototypes, see [31] for a recent example in the biomedical context.

LDA is obviously restricted to data sets which are, at least approximately, separable by single linear decision boundaries. LVQ approaches can implement more complex piecewise linear decision boundaries by using several prototypes, reflecting the cluster geometries and potential multi-modal structures of the classes. In combination with relevance matrix training, LVQ retains this flexibility but at the same time bears the potential to provide a discriminative linear projection of the data.

A first obvious example illustrates this flexibility in terms of an artificial toy data set with two classes in $N = 3$ dimensions,

displayed in Fig. 1 (left panel). Obviously, the data is not linearly separable. Nevertheless, LDA identifies a well-defined direction which maximizes the criterion given in Eq. (13). Due to the elongation of clusters along the y-axis in $\mathbb{R}^3$ and the large distance of the two class 1 clusters along the z-axis, the direction of smallest within class variance corresponds to the x-axis. The largest separation of projected class-conditional means would obviously be obtained along the z-axis. In the actual setting, the former dominates the outcome of LDA. It yields a direction which almost coincides with the x-axis. A slight deviation prevents the prototypes from coinciding in the projection. PCA identifies the direction of largest overall variance, i.e. the y-axis in the example setting.

A GMLVQ analysis with an appropriate set of 3 prototypes, cf. Fig. 1 (center panel), achieves good classification and a discriminative one-dimensional visualization at the same time. Furthermore, its outcome is to a large extent robust with respect to the precise configuration of the clusters.

A classical illustrative example was already considered by Fisher [24]: the well-known *Iris* data set. It is available from the UCI repository [32], for instance. In this simple data set, four features represent properties of 50 individual Iris flowers which are to be assigned to one of three different species.

Here we applied a z-score transformation before processing the data by means of PCA, LDA, and GMLVQ. Hence, the data sets visualized in Figs. 2 and 3 correspond to four features normalized to zero mean and unit variance. Applying unsupervised PCA (left panel) shows already that one of the classes, represented by black symbols, is well separated from the other two Iris species which overlap in the two-dim. projection. Although PCA is unsupervised by definition, the obtained visualization happens to be discriminative to a certain extent: A Nearest Neighbor (NN) classification according to Euclidean distance in the two-dim. space misclassifies 12% of the feature vectors.

For the three class problem, LDA naturally achieves a two-dim. representation, as outlined in Sec. II-B. It results in the visualization shown in the center panel of Fig. 2. The LDA classifier achieves an overall misclassification rate of 2% on the data set. NN-classification in the LDA-projected two-dim. space yields 3.3% error rate, reflecting that its discriminating power is superior compared with unsupervised PCA.

We employed the GMLVQ approach with constant learning rates $\eta_w = 0.25$ and $\eta = 1.25 \cdot 10^{-3}$ in Eq. (7) and (8), respectively. Plain GMLVQ without panelty term achieves an overall Nearest-Prototype error rate of 2%. The rightmost panel in Fig. 2 displays the visualization according to the two leading eigenvectors of the relevance matrix Λ. The corresponding Euclidean NN-classification error is also found to be 2%, reflecting the discriminative power of the projection.

The influence of adding a penalty term, Eq. (10), is illustrated in Fig. 3. The upper left panel corresponds to $\mu = 0$, i.e. original GMLVQ, note the different scaling of axes in comparison with Fig. 2 (right panel). In this case, the first eigenvalue of Λ is approximately 1 and the resulting projection is close to one-dimensional, see Fig. 3 (lower row). The

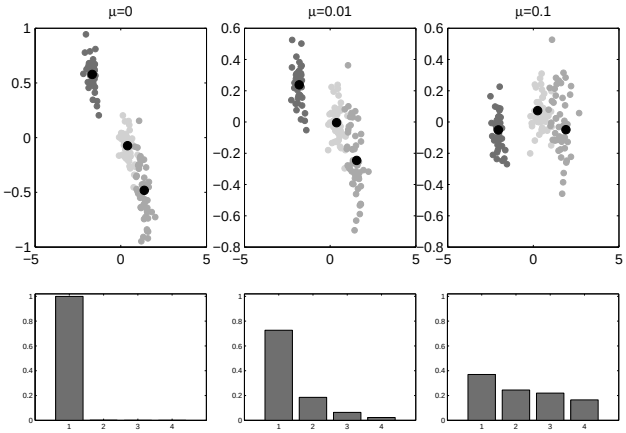

Fig. 3. Influence of the penalty term (10) in GMLVQ on the visualization of the *Iris* data set. Upper panels: The projection of data and protoytpes (black symbols) on the first and second eigenvector of the resulting matrix Λ is displayed for, from left to right, $\mu = 0$, $\mu = 0.01$ and $\mu = 0.1$. Lower panel: the corresponding eigenvalue spectra of the stationary Λ for the three considered values of μ.

presence of a penalty term, $\mu > 0$, enforces non-singular Λ and higher order eigenvalues increase with μ. At the same time the scatter of the data along the second eigendirection of Λ becomes more pronounced. It is interesting to note that, here, the GMLVQ Nearest-Prototype accuracy deteriorates when the penalty is introduced. While for $\mu = 0.01$ the effect is not yet noticeable, we observe an increased error rate of 4% for $\mu = 0.1$. Apparently, optimizing the margin based original cost function (6) is consistent with achieving low error rates in this data set.

The *Landsat* database provides another popular benchmark available at the UCI repository [32]. It consists of 6435 feature vectors in $\mathbb{R}^{36}$, containing spectral intensities of pixels in 3×3 neighborhoods taken from a satellite image. The classification concerns the central pixel in each patch, which is to be assigned to one of 6 classes (red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, very damp grey soil), see [32] for details.

Figure 4 (upper panels) displays the two dimensional visualization of the data set as obtained by PCA and LDA. In the projection restricted to the two leading eigendirections, a Euclidean NN-classification scheme achieves overall misclassification rates of 22.4% in the case of PCA and 27.9% for LDA. It appears counterintuitive that unsupervised PCA should outperform the supervised LDA with respect to this measure. However, one has to take into account that LDA provides a discriminative scheme in $n_c - 1 = 5$ dimensions which is not optimized with respect to the concept of NN-classification. In addition, the restriction to two leading eigendirections limits the discriminative power, obviously. Indeed, the unrestricted LDA system misclassifies only 15.2% of the data.

We also display the outcome of GMLVQ training with $k = 1$ prototype per class and $k = 3$ prototypes per class, respectively, in Fig. 4 (lower panels). Without the penalty term, Eq. (10), the corresponding NN error rates are 27.0%

($k = 1$) and 18.9% ($k = 3$), respectively. Visual inspection also confirms that better discrimination is achieved with more prototypes. Of course, the GMLVQ system is also not optimized with respect to the NN-performance. Indeed, error rates corresponding to Nearest Prototype classification are significantly lower: we obtain 16.3% for $k = 1$ and 15.4% for $k = 3$ prototypes per class, respectively.

Finally, Fig. 5 exemplifies the influence of the penalty term on the GMLVQ system with $k = 3$ prototypes per class. The corresponding eigenvalue spectra are displayed in the rightmost panels. While the leading eigenvalues clearly dominate, the penalty term assigns a certain weight to all eigendirections and Λ remains non-singular.

On the one hand we note that, as higher order directions are contributing more strongly, the NN-classification in two dimensions deteriorates slightly: we obtain error rates of 20.2% for $\mu = 0.01$ and 23.3% with $\mu = 0.1$. On the other hand we observe that the prototype-distance-based classification error varies only slightly with the penalty: we obtain error rates of 14.8% for $\mu = 0.01$ and 15.2% with $\mu = 0.1$, respectively.

The experiments presented here illustrate the tendency of GMLVQ to yield low rank relevance matrices. The results support the idea that this classification scheme can be employed for meaningful visualization of labeled data sets. It is flexible enough to implement complex piecewise linear decision boundaries in high-dimensional multi-class problems, yet it provides discriminative low-dimensional projections of the data, at the same time.

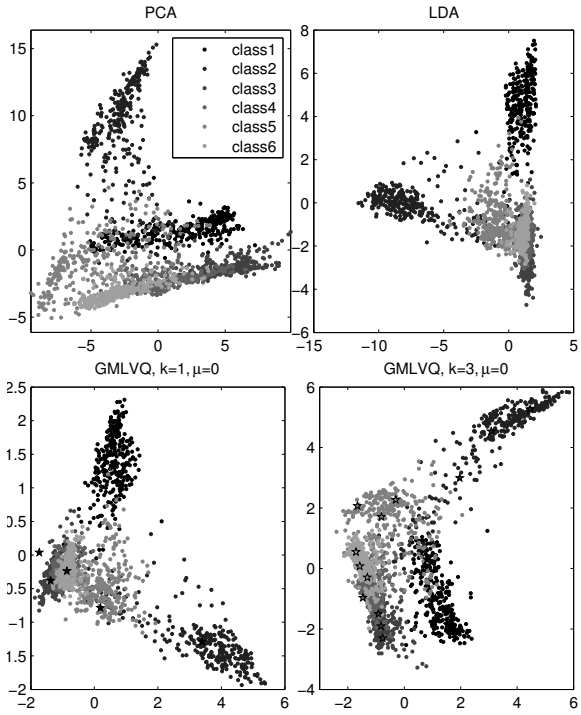

Fig. 4. Two-dimensional visualizations of the *landsat* data set as described in the text. Representaion in terms of the two-dim. projections obtained by PCA (upper left panel), LDA (upper right), GMLVQ with one prototype per class (lower left), and with three prototypes per class (lower right). For the sake of clarity, only 300 randomly selected examples from each class are displayed. Stars mark the projections of GMLVQ protytpes, in addition.

IV. Stationarity of Matrix Relevance Learning

The attractive properties of GMLVQ, as illustrated in the previous section, can be understood theoretically from the generic form of the matrix update, details are presented in a technical report [33]. On average over the random selection of an example $\vec{\xi}^{\nu}$, the stochastic descent update, Eq. (9), can be written as

$$\Omega \leftarrow [I - \eta\, G]\; \Omega \tag{17}$$

with the shorthand $\vec{x}_L = (\vec{\xi} - \vec{w}_L)$. Here, the matrix G is given as a sum over all example data:

$$G = \frac{1}{P} \sum_{\nu=1}^{P} \sum_{m,n=1}^{M} \phi_J(\vec{\xi}^{\nu}, \vec{w}_m)\, \phi_K(\vec{\xi}^{\nu}, \vec{w}_n) \times \left[\frac{d_m^{\nu}}{(d_m^{\nu} + d_n^{\nu})^2} \vec{x}_m^{\nu}\, \vec{x}_m^{\nu\top} - \frac{d_n^{\nu}}{(d_m^{\nu} + d_n^{\nu})^2} \vec{x}_n^{\nu}\, \vec{x}_n^{\nu\top} \right]. \tag{18}$$

where $d_m^{\nu} = d(\vec{\xi}^{\nu}, \vec{w}_m)$. In the sum over pairs of prototypes, the indicator functions $\phi_J = 0, 1$ singles out the closest correct prototype $\vec{w}_J$ with $s_J = \sigma$, while $\phi_K = 0, 1$ identifies the *wrong winner* $\vec{w}_K$ with $s_K \neq \sigma$. Obviously, Eq. (17) can be interpreted as a batch gradient descent step, which coincides with the averaged stochastic update.

It is important to realize that the matrix G in Eq. (17) does change with the LVQ update, in general. Even if prototypes positions are fixed, the assignment of the data to the *winners* as well as the corresponding distances vary with Ω. The following considerations are based on the assumption that in the converging system, i.e. after many training steps, G is reproduced under the update (17). This self-consistency argument implies that the set of prototypes as well as the assignment of input vectors to the $\vec{w}_k$ does not change anymore as Ω is updated. We also have to assume that the individual distances converge smoothly as Ω approaches its stationary state. The potential existence of pathological data sets and configurations which might violate these assumptions will be provided in a forthcoming publication.

We would like to emphasize that G does not have the form of a modified *covariance matrix* since it incorporates label information: Examples are weighted with positive or negative sign. Hence, G is not even positive (semi-) definite, in general. Furthermore, the matrix is given as a sum over all prototypes $\vec{w}_k$ which contribute terms $\propto (\vec{\xi} - \vec{w}_k)(\vec{\xi} - \vec{w}_k)^{\top}$.

To begin with, we assume that an ordering $\rho_1 < \rho_2 \leq \rho_3 \ldots \leq \rho_N$ of the eigenvalues of G exists with a unique smallest eigenvalue ρ_1. We exploit the fact that the set of eigenvectors forms an orthonormal basis $\{\vec{v}_j\}_{j=1}^{N}$ of $I\!R^N$. The influence of degeneracies is discussed below.

An unnormalized update of the form (17) would be dominated by the largest eigenvalue and corresponding eigenvector of the matrix $[I - \eta\, G]$. For sufficiently small η this eigenvalue is $(1 - \eta\rho_1) > 0$. However, the naive iteration of Eq. (17) without normalization would yield either divergent behavior

for $\rho_1 < 0$ or the trivial stationary solution $\Omega \to 0$ for $\rho_1 > 0$. Eq. (17) is reminiscent of the *von Mises* iteration for the determination of leading eigenvalues and eigenvectors [34], where normalization is also required.

Under the constraint that $\sum_{ij} \Omega_{ij}^2 = 1$ and considering the limit of small learning rates $\eta \to 0$, one can show that the stationary solution of Eq. (17) corresponds to a matrix Ω every column of which is a multiple of the eigenvector $\vec{v}_1$:

$$\Omega = [a_1\vec{v}_1, a_2\vec{v}_1, \ldots, a_N\vec{v}_1] \text{ with } \sum_{i=1}^{N} a_i^2 = 1. \tag{19}$$

For a detailed presentation of the argument, see [33]. Exploiting the normalization of the coefficients a_i, we can work out the resulting matrix Λ:

$$\Lambda \; = \; \Omega\,\Omega^\top \; = \vec{v}_1\,\vec{v}_1^\top. \tag{20}$$

Hence, the resulting relevance matrix is given by the eigenvector of G which corresponds to its smallest eigenvalue.

In the case of a k-fold degenerate smallest eigenvalue of G,

$$\rho_1 = \rho_2 = \ldots = \rho_k < \rho_{k+1} \leq \rho_{k+2} \cdots \leq \rho_N,$$

the stationarity condition implies that the columns of Ω are arbitrary vectors from the corresponding eigenspace, see also [33]. It is still possible to construct an orthonormal basis $\{\vec{v}_i\}_{i=1}^k$ of this space and we obtain a stationary

$$\Lambda \; = \; \sum_{i,j=1}^{k} b_{ij}\,\vec{v}_i\,\vec{v}_j^\top \tag{21}$$

where the actual coefficients b_{ij} have to satisfy the symmetry $\Lambda_{mn} = \Lambda_{nm}$ and the normalization $\mathrm{Tr}(\Lambda) = 1$.

The above results are valid in the limit $\eta \to 0$ and for infinitely many training steps. In practice, learning rates $\eta > 0$ and *stopping* after a finite number of updates will result in a matrix Λ with $rank(\Lambda) > 1$, in general. As confirmed in the examples of Sec. (III), Λ is dominated by one leading eigenvector $\vec{v}_1$, generically, but several others also contribute weakly. The incorporation of the penalty term, cf. Eq. (10), prevents Λ from becoming singular, and hence has a similar effect on the resulting eigenvalue spectrum.

Extending the generic update equation (17) by the penalty term gives

$$\Omega \; \propto \; \Omega - \eta\,G\,\Omega + \; \eta\,\mu\,\Omega^{-T}. \tag{22}$$

Its presence complicates the stationarity condition, details of the analysis are presented in [33]. We restrict ourselves to presenting the results with respect to two specific limits:

For very strong penalty, $\mu \to \infty$, one obtains the stationary

$$\Lambda \; = \; \sum_k \vec{v}_k\vec{v}_k^\top \,/N \; = I/N.$$

All eigenvectors contribute equally in this case and the distance reduces to the standard Euclidean measure in original feature space, apart from the normalization factor $1/N$.

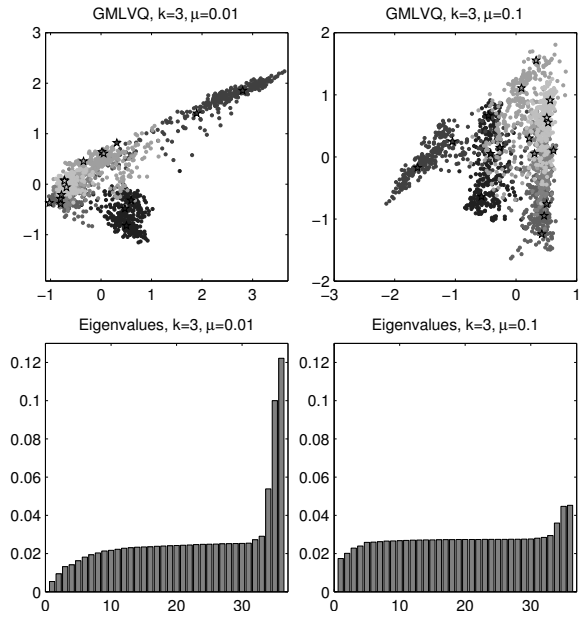

Fig. 5. Influence of the penalty term on the GMLVQ system with three prototypes per class in the landsat data. Upper left panel: visualization for $\mu = 0.01$, upper right: the same with $\mu = 0.1$. For a legend see Fig. 4. Lower left panel: eigenvalues for $\mu = 0.01$, lower right: eigenvalues for $\mu = 0.1$.

The solution becomes particularly transparent for very weak penalization of singularity. As detailed in [33], the self-consisent stationary relevance matrix has the form

$$\Lambda \; = \; \left(1 - \sum_{i\geq 2} \frac{\mu}{\rho_i - \rho_1}\right) \vec{v}_1\vec{v}_1^\top \; + \; \sum_{i\geq 2} \frac{\mu}{\rho_i - \rho_1}\,\vec{v}_i\vec{v}_i^\top \tag{23}$$

in the limit $\mu \to 0$. As expected, the eigendirection corresponding to ρ_1 still dominates the distance measure for small but non-zero μ. The influence of all other eigenvectors $\vec{v}_k$ increases with μ and is inversely proportional to $(\rho_k - \rho_1)$. Here we assumed $\rho_1 < \rho_2$, the extension to degenerate smallest eigenvalues is analogous to the above.

Example results presented in the previous section confirm our theoretical findings qualitatively. More detailed quantitative comparisons will be presented in a forthcoming study.

We have shown that, also in GMLVQ, the obtained projections can be formulated as the solution of a modified eigenvalue problem. In contrast to PCA and LDA, however, neither the corresponding matrix nor the solution can be constructed from a given data set in a straightforward fashion. On the contrary, it results from the iterative process after many steps and depends on initialization and the positioning of prototypes in the course of the training.

V. Conclusion

We have demonstrated that Generalized Matrix Learning Vector Quantization constitutes a powerful method for the visualization of labeled data sets. The framework combines the flexibility and discriminative power of prototype-based

classification with the conceptually simple but versatile low-dimensional representation of feature vectors by means of linear projection. Comparison with classical methods of similar complexity like LDA and PCA illustrate the usefulness and flexibility of the appraoch.

Furthermore, we have presented an analytic treatment of the matrix updates close to stationarity. Like for LDA and PCA, the outcome of GMLVQ can be formulated as a modified eigenvalue problem. However, its characteristic matrix cannot be determined in advance from the data; it depends on the actual training dynamics, including the prototype positions. Consequently, the mathematical treatment of stationarity requires a self-consistency assumption. We have extended the analysis to a variant of GMLVQ which introduces a penalty term as to enforce non-singularity of the projection. It controls the role of higher-order eigenvalues and allows to influence properties of the visualization systematically.

Locally linear extensions of the method which combine global, low rank projections with class-specific relevance matrices defined in the low-dimensional space are currently investigated. This modification can enhance discriminative power significantly, yet retains the conceptual simplicity of the visualization.

Our findings indicate that GMLVQ is a promising tool for discriminative visualization. In particular, we expect a variety of interesting applications in the interactive analysis of complex data sets. In the context of similarity based retrieval, for instance, extensions along the line of the PicSOM approach [35] appear promising. The adaptation of the distance measure based on user-feedback instead of pre-defined labels constitutes another promising application of the approach.

REFERENCES

[1] J. Lee and M. Verleysen, *Nonlinear dimensionality reduction*. Springer, 2007.

[2] K. Bunte, M. Biehl, and B. Hammer, "A general framework for dimensionaliy reducing data visualization mapping," *Neural Computation*, 2012.

[3] P. Schneider, M. Biehl, and B. Hammer, "Adaptive relevance matrices in Learning Vector Quantization," *Neural Computation*, vol. 21, no. 12, pp. 3532–3561, 2009.

[4] K. Bunte, M. Biehl, M. Jonkman, and N. Petkov, "Learning effective color features for content based image retrieval in dermatology," *Pattern Recognition*, vol. 44, pp. 1892–1902, 2011.

[5] M. Biehl, B. Hammer, M. Verleysen, and T. Villmann, Eds., *Similarity based clustering - recent developments and biomedical applications*, ser. Lecture Notes in Artificial Intelligence. Springer, 2009, vol. 5400.

[6] T. Kohonen, *Self-Organizing Maps*, 2nd ed. Berlin, Heidelberg: Springer, 1997.

[7] ——, "Learning Vector Quantization for pattern recognition," Helsinki Univeristy of Technology, Espoo, Finland, Tech. Rep. TKK-F-A601, 1986.

[8] K. Crammer, R. Gilad-Bachrach, A. Navot, and A. Tishby, "Margin analysis of the LVQ algorithm," in *Advances in Neural Information Processing Systems*, vol. 15. MIT Press, Cambridge, MA, 2003, pp. 462–469.

[9] T. Kohonen, "Improved versions of Learning Vector Quantization," in *International Joint Conference on Neural Networks*, vol. 1, 1990, pp. 545–550.

[10] A. Sato and K. Yamada, "Generalized Learning Vector Quantization," in *Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds. Cambridge, MA, USA: MIT Press, 1996, pp. 423–429.

[11] ——, "An analysis of convergence in Generalized LVQ," in *Proceedings of the International Conference on Artificial Neural Networks*, L. Niklasson, M. Bodéén, and T. Ziemke, Eds. Springer, 1998, pp. 170–176.

[12] S. Seo and K. Obermayer, "Soft Learning Vector Quantization," *Neural Computation*, vol. 15, no. 7, pp. 1589–1604, 2003.

[13] S. Seo, M. Bode, and K. Obermayer, "Soft nearest prototype classification," *Transactions on Neural Networks*, vol. 14, pp. 390–398, 2003.

[14] Neural Networks Research Centre, "Bibliography on the Self-Organizing Map (SOM) and Learning Vector Quantization (LVQ)," Helsinki University of Technology.

[15] T. Bojer, B. Hammer, D. Schunk, and K. T. von Toschanowitz, "Relevance determination in Learning Vector Quantization," in *European Symposium on Artificial Neural Networks*, M. Verleysen, Ed., 2001, pp. 271–276.

[16] K. Weinberger, J. Blitzer, and L. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Advances in Neural Information Processing Systems 18*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, pp. 1473–1480.

[17] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.

[18] K. Bunte, P. Schneider, B. Hammer, F.-M. Schleif, T. Villmann, and M. Biehl, "Limited rank matrix learning, discriminative dimension reduction and visualization," *Neural Networks*, vol. 26, pp. 159–173, 2012.

[19] B. Hammer and T. Villmann, "Generalized relevance learning vector quantization," *Neural Networks*, vol. 15, no. 8-9, pp. 1059–1068, 2002.

[20] C. M. Bishop, *Neural Networks for Pattern Recognition*, 1st ed. Oxford University Press, 1995.

[21] C. Darken, J. Chang, J. C. Z, and J. Moody, "Learning rate schedules for faster stochastic gradient search," in *Neural Networks for Signal Processing 2 - Proceedings of the 1992 IEEE Workshop*. IEEE Press, 1992.

[22] P. Schneider, K. Bunte, H. Stiekema, B. Hammer, T. Villmann, and M. Biehl, "Regularization in matrix relevance learning," *IEEE Transactions on Neural Networks*, vol. 21, pp. 831–840, 2010.

[23] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," `http://matrixcookbook.com`, 2008.

[24] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[25] K. Fugunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. Academic Press, 1990.

[26] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. Wiley-Interscience, 2000.

[27] H. Gao and J. Davis, "Why direct LDA is not equivalent to LDA," *Pattern Recognition*, vol. 39, pp. 1002–1006, 2006.

[28] L. van der Maaten, "Matlab toolbox for dimensionality reduction, v0.7.2," `http://homepage.tudelft.nl/19j49/`, 2010.

[29] E. Oja, "Neural networks, principal components, and subspaces," *Journal of Neural Systems*, vol. 1, pp. 61–68, 1989.

[30] T. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network," *Neural Networks*, vol. 2, pp. 459–473, 1989.

[31] W. Arlt, M. Biehl, A. Taylor, S. Hahner, R. Libé, B. Hughes, P. Schneider, D. Smith, H. Stiekema, N. Krone, E. Porfiri, G. Opocher, J. Bertherat, F. Mantero, B. Allolio, M. Terzolo, P. Nightingale, C. Shackleton, X. Bertagna, M. Fassnacht, and P. Stewart, "Urine steroid metabolomics as a biomarker tool for detecting malignancy in adrenal tumors," *Journal of Clinical Endocrinology & Metabolism*, vol. 96, pp. 3775–3784, 2011.

[32] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz, "UCI repository of machine learning databases," `http://archive.ics.uci.edu/ml/`, 1998.

[33] M. Biehl, B. Hammer, F.-M. Schleif, P. Schneider, and T. Villmann, "Stationarity of Relevance Matrix Learning Vector Quantization," University of Leipzig, Machine Learning Reports, Tech. Rep. MLR 01/2009, 2009.

[34] W. Boehm and H. Prautzsch, *Numerical Methods*. Vieweg, 1993.

[35] J. Laakonen, M. Koskela, S. Laakso, and E. Oja, "PicSOM - content-based image retrieval with self-organizing maps," *Pattern Recognition Letters*, vol. 21, pp. 1199–1207, 2000.