Discriminatory Data Mapping by Matrix-Based Supervised Learning Metrics

M. Strickert[1,⋆], P. Schneider[2], J. Keilwagen[1],
T. Villmann[3], M. Biehl[2], and B. Hammer[4]

[1] Leibniz Institute of Plant Genetics and Crop Plant Research Gatersleben
[2] Institute for Mathematics and Computing Science, University of Groningen
[3] Research group Computational Intelligence, University of Leipzig
[4] Institute of Computer Science, Technical University of Clausthal
stricker@ipk-gatersleben.de

Abstract. Supervised attribute relevance detection using cross-comparisons (SARDUX), a recently proposed method for data-driven metric learning, is extended from dimension-weighted Minkowski distances to metrics induced by a data transformation matrix $\boldsymbol{\Omega}$ for modeling mutual attribute dependence. Given class labels, parameters of $\boldsymbol{\Omega}$ are adapted in such a manner that the inter-class distances are maximized, while the intra-class distances get minimized. This results in an approach similar to Fisher's linear discriminant analysis (LDA), however, the involved distance matrix gets optimized, and it can be finally utilized for generating discriminatory data mappings that outperform projection pursuit methods with LDA index. The power of matrix-based metric optimization is demonstrated for spectrum data and for cancer gene expression data.

Keywords: Supervised feature characterization, adaptive matrix metrics, attribute dependence modeling, projection pursuit, LDA.

1 Introduction

Learning metrics constitute one of the most exciting topics in machine learning research [11,17,18]. The potential of metric adaptation needs exploration for facing challenges connected to the curse of dimensionality in high-throughput biomedical data sets. Mass spectra, gene expression arrays, or 2D electrophoretic gels, given as vectors of real-value measurements, are often characterized by only a low number of available experiments as compared to their huge dimensionality. Data-driven adaptation of a data metric can be used in many helpful ways. Applications of metric optimization range from attribute weighting via dimension reduction to data transformations into task-specific spaces.

The adaptive Euclidean distance $\mathrm{d}_{\boldsymbol{\lambda}}(\boldsymbol{x},\boldsymbol{y}) = (\sum_{i=1}^{q} \lambda_i (x_i - y_i)^2)^{1/2}$, for example, relates attribute characterization to the choice of attribute scaling factors λ_i beneficial for the separation of labeled and unlabeled data in supervised and

⋆ Corresponding author.

L. Prevost, S. Marinai, and F. Schwenker (Eds.): ANNPR 2008, LNAI 5064, pp. 78–89, 2008.

unsupervised manners. This aim is shared with projection pursuit methods for which matrix parameters of a linear projection mapping are optimized with respect to criteria of data spreading and clusterability of the low-dimensional projections [4]. In this work, evaluation takes place in a space of original dimensionality where only the comparison criterion, the metric, is changed. If desired, attribute-related parameters with low impact, for example expressed as low scaling factors, can be pruned for dimension reduction, after adaptation. For the parametric Euclidean distance, small attribute scaling factors λ_i would indicate negligible attributes. Scaling factors can be also used for transforming the data to the non-adapted Euclidean space for further utilization of standard Euclidean methods. This kind of attribute characterization is different from many other methods for feature extraction [6], such as the recently suggested Iterative Relief algorithm [16] for which the attribute weights do not coincide with a canonic rescaling of the data space.

Matrix-based metrics help to extend the view of individual attribute processing to a model of dependence between pairs of attributes. Generally, matrix methods can be used for optimizing linear data transformations aiming at criteria related to the data spreading. In the unsupervised case, interesting transforms include sphering of the data covariance matrix to the unity matrix, or the projection of data to directions of maximum variance (PCA) or to directions along maximum non-Gaussianity (ICA) [10]. The projection pursuit method [4] is a very flexible approach to extract projections of interest by optimizing a target function, called the index of the projection. Such indices exist for unsupervised cases aiming at mappings to continuous or sharply clustered views. In addition, there are supervised indices like projection entropy and class separability according to linear discriminant analysis (LDA) criteria. A good environment for the study of projection pursuit and other matrix methods is, for example, provided by the free R statistical language with rGGobi and classPP packages [2,12], an application of classPP for the visualization of gene expression data, is provided in [3].

An alternative view on seeking optimum data transformations is data-driven adaptation of the data metric or, more generally, of a data similarity measure. Learning vector quantization (LVQ), for example, can implement metric adaptation for better data classification by boosting class-separating attributes between data prototype vectors. The generalized relevance LVQ method (GRLVQ) realizes such metric adaptation by using a misclassification cost function – minimized by gradient descent – making use of data labels for attribute rescaling [8]. For Euclidean distances, large-margin optimization is realized, but also non-Euclidean similarity-measures profit from parameter adaptation [7,14]. Recently, matrix learning has been integrated into the GRLVQ framework for modeling attribute-attribute dependencies by generalized Mahalanobis distance [13]. This allows to express scalings of the data space along arbitrary directions, and very good classification accuracies are obtained on difficult classification problems ranging from spectrum classification to image segmentation.

The success of matrix metric adaptation in GRLVQ classifiers initiated the present work. Here, no classifier will be build though, but the data space will

be transformed: directions in the data space relevant to data label separation will be emphasized, while within-class variations will be damped. After all, relevant combinations of attributes, trained and expressed in form of a matrix, are identified for the discrimination task. Since only few data samples per class can be expected in costly and time-consuming biomedical studies, prototype-based data abstraction, like provided by GRLVQ, is avoided in order to keep maximum information. For the analyzed data sets it turned out that the transform matrix could be effectively compressed to only a few prominent eigenvectors, possibly only one, without significant loss of metric structure. After all, we are able to compute relatively compact discriminatory data models that allow hypotheses generation for supporting biomedical experts.

2 Method

Data. The q-dimensional row input vector $\mathbf{x} \in \mathbb{R}^{1\times q}$ is taken from a data set containing n data vectors $\{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n\}$. The proposed metric adaptation requires that each vector $\mathbf{x}^k$ is labeled with one class-specific index $\mathrm{c}(k)$, assuming at least two unique classes in the whole data set.

Metric. Most essential is the definition of the matrix-based metric $\mathrm{d}^{ij}_{\boldsymbol{\Omega}} \in [0;\infty)$ between data vectors $\mathbf{x}^i$ and $\mathbf{x}^j$:

$$\mathrm{d}^{ij}_{\boldsymbol{\Omega}} = \mathrm{d}_{\boldsymbol{\Omega}}(\mathbf{x}^i, \mathbf{x}^j) = (\mathbf{x}^i - \mathbf{x}^j) \cdot \boldsymbol{\Lambda} \cdot (\mathbf{x}^i - \mathbf{x}^j)^{\mathsf{T}}\,, \quad (\boldsymbol{\Lambda} = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}) \in \mathbb{R}^{q\times q}\,. \tag{1}$$

Choosing the identity matrix $\boldsymbol{\Lambda} = \boldsymbol{\Omega} = \boldsymbol{I}$ induces the special case of the squared Euclidean distance; other diagonal matrices yield weighted squared Euclidean distances as discussed in [15]. Generally, metrics are obtained for arbitrary positive-definite matrices $\boldsymbol{\Lambda}$. Then the value expressed by $\boldsymbol{\Delta} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{\Delta}^{\mathsf{T}} \geq 0$, getting zero only for trivial difference vectors $\boldsymbol{\Delta} = \mathbf{0}$, is a metric. It is known that in context of metrics non-symmetric positive-definite matrices can be replaced by equivalent symmetric positive-definite matrices. Since any symmetric positive-definite matrix $\boldsymbol{\Lambda}$ can be decomposed by Cholesky decomposition into a product of a lower triangular matrix and its transposed, it is in principle sufficient to learn a lower triangular matrix $\boldsymbol{\Omega}$ for expressing $\boldsymbol{\Lambda}$. Alternatively, symmetric positive-definite $\boldsymbol{\Lambda}$ can be represented by the self-product $\boldsymbol{\Lambda} = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}$ of a symmetric $\boldsymbol{\Omega}$ [13]. Here, we consider products $\boldsymbol{\Lambda} = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$ with arbitrary $\boldsymbol{\Omega} \in \mathbb{R}^{q\times q}$. These full matrices $\boldsymbol{\Omega}$, possess more adaptive matrix elements than degrees of freedom needed for expressing the product solution space of $\boldsymbol{\Lambda}$. For the data sets discussed, the interaction of matrix element pairs Ω_{ij} and Ω_{ji} leads to a faster convergence of $\boldsymbol{\Lambda}$ during optimization, compared to the convergence properties obtained for symmetric or triangular matrices $\boldsymbol{\Omega}$.

Note that for some the data $\boldsymbol{\Lambda}$ might become positive-semidefinite during optimization, i.e. $\boldsymbol{\Delta} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{\Delta}^{\mathsf{T}} = 0$ with difference vectors $\boldsymbol{\Delta} \neq \mathbf{0}$. Then, the metric property gets relaxed to a mathematical distance with vanishing self-scalar

product $(\boldsymbol{\Delta} \cdot \boldsymbol{\Omega}) \cdot (\boldsymbol{\Delta} \cdot \boldsymbol{\Omega})^\intercal = \langle \boldsymbol{\Delta} \cdot \boldsymbol{\Omega}, \boldsymbol{\Delta} \cdot \boldsymbol{\Omega} \rangle = 0$ becoming zero for certain configurations of $\boldsymbol{\Omega}$ with $\boldsymbol{\Delta} \cdot \boldsymbol{\Omega} = \mathbf{0}$, else positive.

Adaptation. Driven by the goal to minimize within-class differences while maximizing between class differences, the following cost function is minimized over pairs of all n data items:

$$\mathrm{s}(\boldsymbol{\Omega}) := \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} \mathrm{d}_{\boldsymbol{\Omega}}(\mathbf{x}^i, \mathbf{x}^j) \cdot \delta_{ij}}{\sum_{i=1}^{n} \sum_{j=1}^{n} \mathrm{d}_{\boldsymbol{\Omega}}(\mathbf{x}^i, \mathbf{x}^j) \cdot (1 - \delta_{ij})} = \frac{\mathrm{d}_C}{\mathrm{d}_D}, \quad \delta_{ij} = \begin{cases} 0 : \mathrm{c}(i) \neq \mathrm{c}(j) \\ 1 : \mathrm{c}(i) = \mathrm{c}(j) \end{cases} \tag{2}$$

Distances $\mathrm{d}_{\boldsymbol{\Omega}}^{ij}$ between data vectors $\mathbf{x}^i$ and $\mathbf{x}^j$ depend on the adaptive matrix parameters $\boldsymbol{\Omega} = (\Omega_{kl})_{\substack{k=1\ldots q \\ l=1\ldots m}}$ of interest. The numerator represents within-class data scatter, which should be small; the denominator is related to inter-class distances, which should be large. Thus, optimization of $\mathrm{s}(\boldsymbol{\Omega})$ handles both parts of the fraction simultaneously. Compromise solutions must be found in cases when within-class variation, potentially caused by outliers, needs compression, while inter-class separability would require inflation.

Using the chain rule, the cost function $\mathrm{s}(\boldsymbol{\Omega})$ is iteratively optimized by gradient descent $\boldsymbol{\Omega} \leftarrow \boldsymbol{\Omega} - \gamma \cdot \frac{\partial \mathrm{s}(\boldsymbol{\Omega})}{\partial \boldsymbol{\Omega}}$, which requires adaptation of the matrix $\boldsymbol{\Omega}$ in small steps γ into the direction of steepest gradient

$$\frac{\partial \mathrm{s}(\boldsymbol{\Omega})}{\partial \boldsymbol{\Omega}} = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial \mathrm{s}(\boldsymbol{\Omega})}{\partial \mathrm{d}_{\boldsymbol{\Omega}}^{ij}} \cdot \frac{\partial \mathrm{d}_{\boldsymbol{\Omega}}^{ij}}{\partial \boldsymbol{\Omega}} \,. \tag{3}$$

The quotient rule applied to the fraction $\mathrm{s}(\boldsymbol{\Omega}) = \mathrm{d}_C / \mathrm{d}_D$ in Eqn. 2 yields

$$\frac{\partial \mathrm{s}(\boldsymbol{\Omega})}{\partial \mathrm{d}_{\boldsymbol{\Omega}}^{ij}} = \frac{\delta_{ij} \cdot \mathrm{d}_D}{\mathrm{d}_D^2} + \frac{(\delta_{ij} - 1) \cdot \mathrm{d}_C}{\mathrm{d}_D^2} = \begin{cases} 1/\mathrm{d}_D \,:\, \mathrm{c}(i) = \mathrm{c}(j) \\ -\mathrm{d}_C/\mathrm{d}_D^2 \,:\, \mathrm{c}(i) \neq \mathrm{c}(j) \end{cases} . \tag{4}$$

The right factor in Eqn. 3 is obtained by matrix derivative of Eqn. 1:

$$\frac{\partial \mathrm{d}_{\boldsymbol{\Omega}}^{ij}}{\partial \boldsymbol{\Omega}} = 2 \cdot (\mathbf{x}^i - \mathbf{x}^j)^\intercal \cdot (\mathbf{x}^i - \mathbf{x}^j) \cdot \boldsymbol{\Omega} \,. \tag{5}$$

If desired, adaptation can be restricted to certain structures of $\boldsymbol{\Omega}$, such as to the lower triangular elements. In that case, undesired elements must be initially masked out by zeros in $\boldsymbol{\Omega}$. Additionally, the same zero masking pattern must be applied to the matrix resulting from Eqn. 5, because the equation calculates $\partial \mathrm{d}_{\boldsymbol{\Omega}}^{ij} / \partial \boldsymbol{\Omega}$ correctly only for full adaptive matrices $\boldsymbol{\Omega}$. By consistent masking operations, though, the matrix of derivatives is mathematically correct. In practice, the gradient from Eqn. 3 is computed and reused as long the cost function decreases. Potential increase of $\mathrm{s}(\boldsymbol{\Omega})$ triggers a recomputation of the gradient. The step size γ is dynamically determined as the initial size γ_0, being exponentially cooled down by rate η, divided by the maximum absolute element in the matrix $\partial \mathrm{s}(\boldsymbol{\Omega}) / \partial \boldsymbol{\Omega}$.

Initialization. Empirically, the initial step size γ_0 can be chosen from the interval $[0.05; 1)$, such as 0.75 in the conducted experiments. The number of iterations

should be set to a value between 50 and 1000, depending on the saturation characteristics of the cost function. The exponential cooling rate should diminish the original step size by some orders of magnitude during training, for example, set to $\eta = 0.995$ for 1000 iterations.

The initialization of matrix $\boldsymbol{\Omega}$ is of particular interest. If chosen as identity matrix $\boldsymbol{\Omega} = \boldsymbol{I}$, the algorithm starts from the usual squared Euclidean distance. For data sets with strong mutual attribute dependencies, i.e. prominent non-diagonal elements, the uniform structure of the identity matrix might lead to unnecessary iterations required for the symmetry breaking, as often encountered in neural network adaptations. Therefore, the alternatively proposed method is random matrix element sampling from uniform noise in the interval $[-0.5; 0.5]$. This noise matrix $\boldsymbol{A} \in \mathbb{R}^{q \times q}$ is broken by QR-decomposition into $\boldsymbol{A} = \boldsymbol{Q} \cdot \boldsymbol{R}$, of which the $\boldsymbol{Q}$-part is known to form an orthonormal basis with $\boldsymbol{Q} \cdot \boldsymbol{Q}^\top = \boldsymbol{I}$. This makes $\boldsymbol{\Omega} = \boldsymbol{Q}$ our preferred initial candidate.

Relation to LDA. At first glance, the proposed cost function looks quite similar to the inverse fraction of the LDA cost function for C classes that is maximized:

$$\mathsf{S}_{LDA} = \frac{\boldsymbol{v} \cdot \left[\sum_{i=1}^{C} n_i \cdot (\boldsymbol{\mu_i} - \boldsymbol{\mu})^\top \cdot (\boldsymbol{\mu_i} - \boldsymbol{\mu})\right] \cdot \boldsymbol{v}^\top}{\boldsymbol{v} \cdot \left[\sum_{i=1}^{C} \boldsymbol{\Sigma}_i\right] \cdot \boldsymbol{v}^\top}\,, \quad n_i = |\{\mathbf{x}^j : \mathrm{c}(j) = i\}|\,. \quad (6)$$

The numerator contains the between-class variation as the squared difference between class centers $\boldsymbol{\mu_i}$ of all vectors $\mathbf{x}^j$ belonging to class i and the overall center $\boldsymbol{\mu} = 1/n \cdot \sum_{k=1}^{n} \mathbf{x}^k$. The denominator describes the within-class variation over all classes i expressed by the sum of squared differences from class centers $\boldsymbol{\mu_i}$ contained in the covariance matrices $\boldsymbol{\Sigma}_i = \sum_{j:\mathrm{c}(j)=i} (\mathbf{x}^j - \boldsymbol{\mu_i})^\top \cdot (\mathbf{x}^j - \boldsymbol{\mu_i})$.

LDA seeks an optimum direction vector $\boldsymbol{v}$ representing a good compromise of being collinear along the class centers (numerator, separating) and orthogonal to maximum within-class variation (denominator, compressing).

If multiple directions $\boldsymbol{V} = (\boldsymbol{v}_k)^\top$ are computed simultaneously, the products in the numerator and denominator of Eqn. 6, involving the matrices in square brackets, become matrices as well. In order to circumvent the problem of valid ratio calculation with matrices, determinants of the obtained matrices can be taken, as discussed in the LDA-based projection pursuit approach [12]. As a result, the LDA ratio optimizes low-dimensional projections onto discriminatory directions.

Our approach is structurally different, because the (inverse) LDA ratio in Eqn. 2 operates in the original data space, subject to the dynamically optimized metric. This explains the higher computational demands compared to LDA for which covariance matrices and class centers can be initially computed and then reused. As a benefit of the new approach, numerator and denominator of the new ratio in Eqn. 2 naturally contain sums of real-valued distances, which avoids problems of handling singular determinants in low-rank matrices.

3 Experiments

3.1 Tecator Spectral Data Set

The benchmark spectral data set, taken from the UCI repository of machine learning [1], contains 215 samples of 100-dimensional infrared absorbance spectra recorded on a Tecator Infratec Food and Feed Analyzer working in the wavelength range 850–1050nm by the Near Infrared Transmission (NIT) principle. The original regression problem accompanying the data set is reformulated as attribute identification task for explaining the separation of 183 samples with low fat content and 77 high fat meat probes.

View 1. An exploratory data view is obtained from the left panel of Fig. 1 and from the PCA projection shown in the left scatter plot of Fig. 2. As expected, the strong spectrum overlap cannot be resolved by PCA projection. After application of the matrix learning all spectra were transformed according to $\boldsymbol{z} = \mathbf{x} \cdot \boldsymbol{\Omega}$, which realizes the left transformation part of the metric given in Eqn. 1; the right part is just $\boldsymbol{z}^\intercal$. The result of this data transformation leads to a good separation with

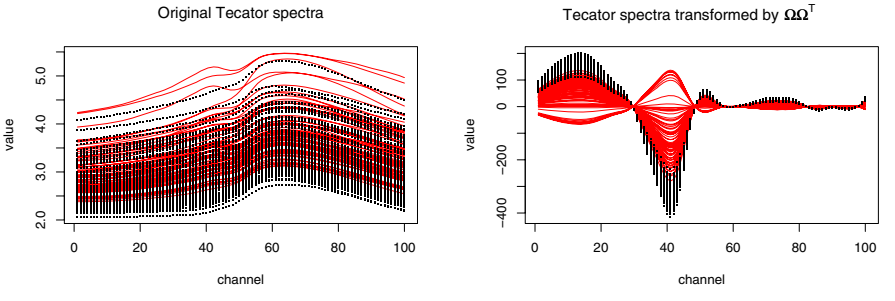

Fig. 1. Tecator spectra, raw (left) and transformed (right). Low fat content is reflected by dashed lines, high fat content by solid lines.

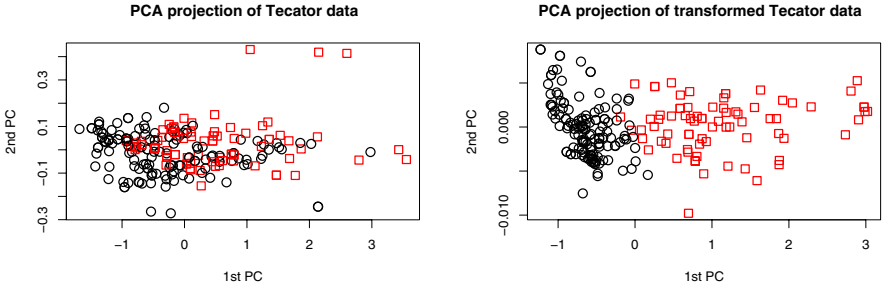

Fig. 2. Scatter plots of Tecator data. Bullets (○) denote low-fat samples, squares (□) high fat content. Left: PCA projection of original data. Right: PCA projection of data transformed by $\boldsymbol{\Omega}$.

almost no overlap in the PCA projection. This is shown in the right panel of Fig. 2.

View 2. By reformulating the metric definition in Eqn. 1 according to

$$\mathrm{d}^{ij}_{\boldsymbol{\Omega}} = (\mathbf{x}^i - \mathbf{x}^j) \cdot \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^\mathsf{T} \cdot (\mathbf{x}^i - \mathbf{x}^j)^\mathsf{T} = \langle \mathbf{x}^i \cdot \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^\mathsf{T} - \mathbf{x}^j \cdot \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^\mathsf{T}, \mathbf{x}^i - \mathbf{x}^j \rangle \,. \quad (7)$$

another interesting perspective on the data is obtained. This is a formal metric decomposition into a static part of difference vectors of the original data (right part of the scalar product) and a dynamically adapted transformation space of the data (left argument of the scalar product). A look into this space is obtained by the transformation to $\mathbf{x}^* = \mathbf{x} \cdot \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^\mathsf{T}$. The resulting transformed spectra with their amazingly separated attributes are shown in the right panel of Fig. 1.

The learned metric can be nicely presented by the matrices $\boldsymbol{\Omega}$ and $\boldsymbol{\Lambda}$ shown in the left and right panel of Fig. 3, respectively. As displayed for $\boldsymbol{\Lambda}$, attribute dependence is most prominent in the channel range 35–45. Strong emphasis of these channels around the diagonal is accompanied by simultaneous repression of the off-diagonal channels 5–30.

Matrix reduction. Since full matrices are quite big models, the study of their compressibility is important. Eigen decomposition of $\boldsymbol{\Lambda} = \boldsymbol{S} \cdot \mathbf{W} \cdot \mathbf{W}^{-1}$ into the diagonal eigenvalue matrix $\boldsymbol{S}$ and the eigenvectors matrix $\mathbf{W}$ helps to reach substantial compressions. In the current case, the highest eigenvalue contributes an amount of 95.3%, thus most variation in the learned matrix $\boldsymbol{\Omega}$ can be explained by the corresponding eigenvector $\boldsymbol{w}$, a column vector. Therefore, up to a scaling factor, a very good reconstruction of $\boldsymbol{\Lambda}$ by $\boldsymbol{w} \cdot \boldsymbol{w}^\mathsf{T}$ is obtained, as confirmed in the left matrix plot of Fig. 4. If the spectra are projected onto $\boldsymbol{w}$, still a very good class separation is obtained, as demonstrated by the corresponding class-specific box plot in the right panel of Fig. 4.

The computational demands are quite high, though, requiring roughly one hour for 1000 updates of the matrix gradient. In contrast to that, the classPP [12] package is much faster, if only a class-separating projection is desired. In principle, classPP takes only several seconds or minutes, depending on the choice of

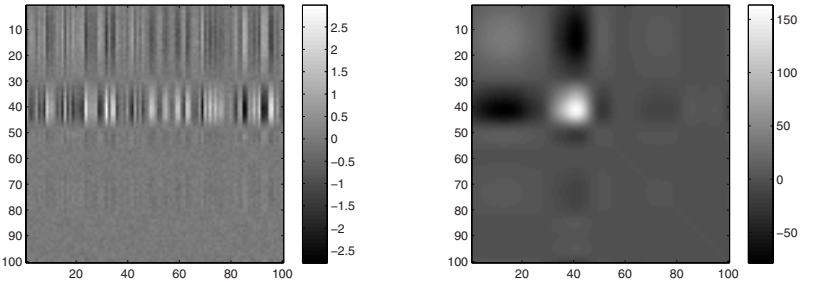

Fig. 3. Matrix representation of optimum metric for the 100-D Tecator data set. The learned matrix $\boldsymbol{\Omega}$ is shown on the left, its squared counterpart $\boldsymbol{\Lambda} = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^\mathsf{T}$ on the right. Interesting dependencies are found around channel index 40.

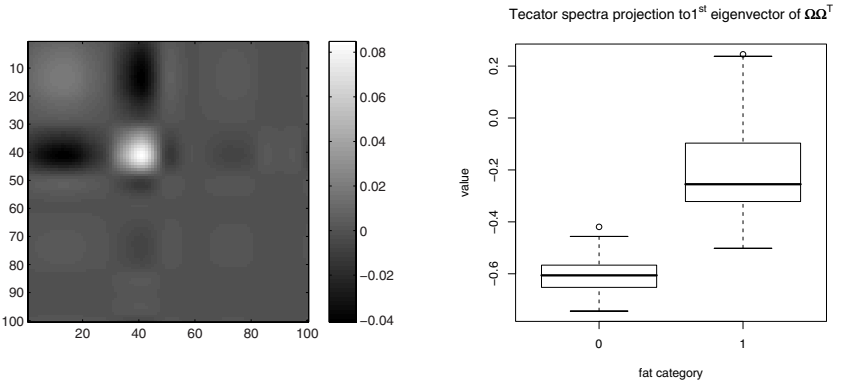

Fig. 4. Representation of $\boldsymbol{\Lambda}$ by its first eigenvector. Left: plot of reconstructed matrix. Right projection of Tecator spectra to the eigenvector.

annealing parameters. However, no stable solution could be obtained, because of the degeneration of the projection vectors, probably caused by near-singular matrix determinants during the computation. Training with our proposed method showed very stable results, converging to the presented solution for different random initializations of $\boldsymbol{\Omega}$. Projection displays are just a by-product of our method. It is important to remember that the original dimensionality of the data space is preserved by the transformation, enabling further utilization with any classification or projection method.

3.2 Gene Expression Analysis of AML/ALL Cancer

Many well-documented and deeply investigated data sets are freely available in cancer research. Since its publication in 1999 the leukemia gene expression dataset [5] used here has become a quasi-benchmark for testing feature selection methods. The original research aimed at the identification of the most informative genes for modeling and classification of two cancer types, acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). The training data covers 7129 genes by 27 cases of ALL and 11 cases of AML. The test data set contains 20 cases of ALL and 14 cases of AML.

The projection of the complete set of training and test data to the first two principal components yields the scatter plot shown in the left panel of Fig. 5. It is worth noticing that a systematic difference between AML training set and test set is indicated by the unbalanced distribution of closed and open bullets. Thus, a training set specific bias is induced during training. Matrix learning is computationally very expensive, because $\boldsymbol{\Omega}$ is a 7129x7129 matrix. Thus, it takes roughly 40 hours on a 2.4 GHz system in order to achieve 500 gradient changes expressed by Eqn. 5. Yet, gradients are reused until first cost function degradation, creating several thousand updates, after all. Since experiment preparation in lab requires much more time, a two day calculation period is no principal problem.

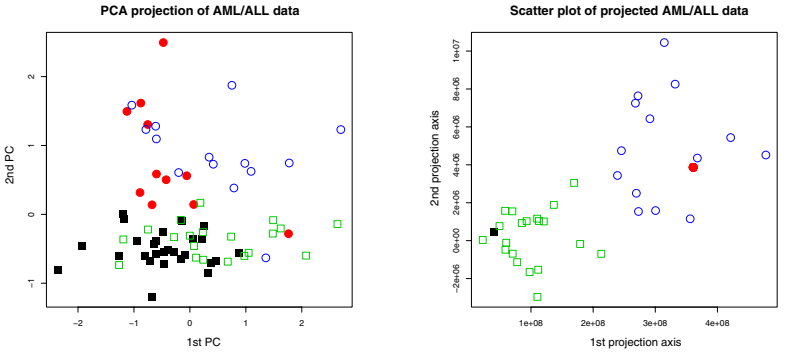

Fig. 5. Scatter plots of AML/ALL gene expression data. Bullets (○) denote expression samples of AML, squares (□) are related to ALL cancer. Closed symbols indicate training data, open symbols test data. Left panel: principal component projection without distinction between training and test data. Right panel: data projected to the first two eigenvectors of the trained interaction matrix $\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$. The training data is perfectly arranged, being very distinct and almost contracted to points.

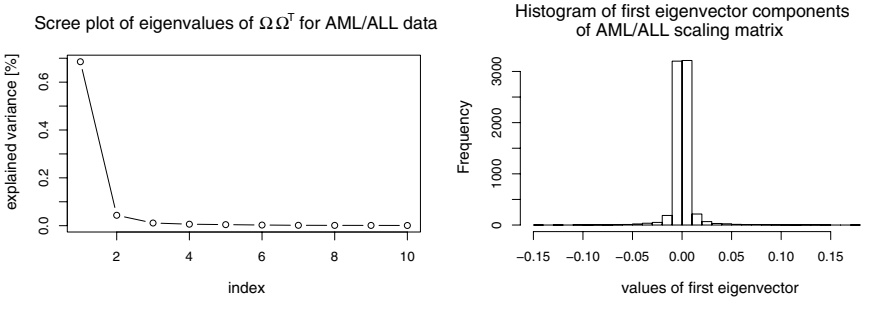

Fig. 6. Eigenvalues of the interaction matrix $\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$ (left) for the AML/ALL data, and the distribution of values in the most prominent eigenvector (right)

Several interesting results are obtained after training. The top ten eigenvalues of $\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$, displayed in the left panel of Fig. 6, point out a very strong explicative power of roughly 70% explained variance of the first eigenvector, dominating the all other eigenvectors. The right panel in Fig. 6 further indicates that only a small fraction of extreme values in that first eigenvector really contains interesting attribute magnifications. In case of data projection, many other attributes are transformed to near zero values, i.e. only few differentiating genes become emphasized.

The projection to the first two eigenvectors of the interaction matrix already yields a perfect separation of the training data into AML and ALL, as shown in the right panel of Fig. 5. Also a very strong compression of the within-class scatter almost to points is obtained. The test data, projected the same way and added to the plot, is still well-separated, but shows much larger variability. This is a clear indication of over-fitting. Sure, a huge 7129x7129 model has been

trained; yet, the displayed projection only contains several hundred effective parameters, because the first two eigenvectors are much dominated by the few most prominent entries of the first eigenvector. A simple center-of-gravity model of the data, for example, would already be much larger.

For comparison, LDA indices of the projections shown in the right panel of Fig. 5 were calculated according to Eqn. 6 using the classPP package. An almost perfect near-one value of $\mathsf{S}_{LDA} = 1 - 7.894 \cdot 10^{-12}$ was achieved for the projected training data and good value of $\mathsf{S}_{LDA} = 0.8298$ for the test data. Using the built-in simulated annealing strategy, the classPP package itself reported a best seen index value of $\mathsf{S}_{LDA} = 1 - 2.825 \cdot 10^{-6}$ for the optimized projections of the training data. Since the corresponding projection matrix is not returned correctly from the classPP package, an application to the test set was not possible. The authors of classPP have identified internal rounding errors in their package.

Individual gene variances correspond to gene-specific scaling factors, compensated by the cost function by adaptation of the related matrix entries. As a consequence, components in the most prominent eigenvector show low correlation with the variance in the original data set and, accordingly, systematically separating low-variance genes were able to gain high rankings in the eigenvector.

The real benefit of the proposed method is the possibility to infer putative gene-gene interactions responsible for cancer type separation. For that purpose the indices i, j corresponding to the most extreme (high and low) values in the matrix $\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$ are extracted and associated with the genes i and j. Because of symmetry, only the lower triangular matrix, including diagonal, is considered. The top 100 pairs extracted this way are compiled in Tab. 1. After all, 14 prominent self-dependent genes are detected as individual factors on the diagonal, three of them are coinciding with the list of Golub et al. of 50 genes. Three more genes of that study are detected on non-diagonal elements as dependent.

Table 1. Table of genes specific to separation of AML/ALL cancer in alphabetic reading order. The listed genes correspond to the 100 most extreme entries in the lower triangular part of the obtained symmetric matrix $\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$. As single genes participate multiple times in combination with others, only 30 different out of 200 possible genes appear in the table. Numbers indicate the frequencies of occurrence. Underlined genes appear also on the diagonal, stressing their individual importance. For illustration purposes, bold face genes are those acting in combination with M19507 which is the overall top-ranked gene. Asterisks mark genes coinciding with top-rated genes from the study of Golub et al.

D49824	**HG3576-HT3779**	L06797	L20688	L20941	M11147
6	13	3	1	1	1
M14328	M17733	**M19507**	**M24485**	**M27891***	**M28130_rna1***
1	1	**26**	1	3	1
M33600	**M69043***	**M77232_rna1**	**M91036_rna1**	**M91438**	**M96326_rna1***
4	1	1	13	1	14
S73591	**U01317_cds4**	**U14968**	V00594	**X14046**	**X17042***
1	13	1	1	1	12
X78992	**Y00433**	**Y00787***	**Z19554**	**Z48501**	**Z70759**
13	14	14	14	11	13

The most prominent gene found by the new method is M19507, which is not mentioned in the Golub study. Yet, the gene is confirmed as relevant in more recent publications, such as [9]. As this gene is connected to more than 20 other top-rated genes, its central role in the discriminatory transcriptome is clearly pointed out. Yet, the whole potential of the analysis, including proper interpretation of the findings, must be thoroughly worked out together with biological experts.

4 Conclusions and Outlook

A data-driven metric in flavor of a generalized Mahalanobis distance has been proposed that makes use of label information for emphasizing or repressing class-specific attribute combinations. Similar to LDA, metric optimization of $\boldsymbol{\Lambda} = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega}^{\mathsf{T}}$ seeks improved inter-class separation with simultaneous minimization of within-class variation. In contrast to LDA, it is not the low-dimensional projection to be optimized, but a transformation in the data space. The new method is not primarily designed for visual projection or classification, but it is a first step towards, because the resulting transformed data can be used as a preprocessing step for subsequent standard methods. As illustrated, visual data exploration is easily possible by projecting the data to the most prominent eigenvectors of $\boldsymbol{\Lambda}$. No sophisticated optimization method is required, simple gradient descent works very reliably on the inverse LDA-like cost function. Both investigated data sets led to convergence to very useful label-specific metrics for different initializations of $\boldsymbol{\Omega}$. The main drawback of the new method is its long runtime for handling the potentially large matrices. Yet, as the discussed cases showed a strong dominance of only the first principal direction, future work will focus on the development of a sparse learning scheme for computing only the k most prominent eigenvectors instead of the whole matrix. This will help to reduce the model size and to speed up the optimization procedure. Finally, a better control of intra- and inter-class contributions to the cost function will be investigated.

Acknowledgment

The work is supported by grant XP3624HP/0606T, Ministry of Culture Saxony-Anhalt.

References

1. Blake, C., Merz, C.: UCI repository of machine learning databases (1998)
2. Cook, D., Swayne, D.: Interactive and Dynamic Graphics for Data Analysis with R and GGobi. Springer, Heidelberg (2007)
3. Faith, J., Mintram, R., Angelova, M.: Targeted projection pursuit for visualizing gene expression data classifications. Bioinformatics 22(21), 2667–2673 (2006)

4. Friedman, J.: Exploratory projection pursuit. Journal of the American Statistical Association 82, 249–266 (1987)
5. Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., Lander, E.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439), 531–537 (1999)
6. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.: Feature Extraction: Foundations and Applications. Springer, Berlin (2006)
7. Hammer, B., Strickert, M., Villmann, T.: Supervised neural gas with general similarity measure. Neural Processing Letters 21(1), 21–44 (2005)
8. Hammer, B., Villmann, T.: Generalized relevance learning vector quantization. Neural Networks 15, 1059–1068 (2002)
9. Hu, S., Rao, J.: Statistical redundancy testing for improved gene selection in cancer classification using microarray data. Cancer Informatics 2, 29–41 (2007)
10. Hyvärinen, A., Oja, E.: Independent component analysis: Algorithms and applications. Neural Networks 13(4–5), 411–430 (2000)
11. Kaski, S.: From learning metrics towards dependency exploration. In: Cottrell, M. (ed.) Proceedings of the 5th International Workshop on Self-Organizing Maps (WSOM), pp. 307–314 (2005)
12. Lee, E., Cook, D., Klinke, S., Lumley, T.: Projection pursuit for exploratory supervised classification. Journal of Computational and Graphical Statistics 14(4), 831–846 (2005)
13. Schneider, P., Biehl, M., Hammer, B.: Adaptive relevance matrices in learning vector quantization (Submitted to Machine Learning) (2008)
14. Strickert, M., Seiffert, U., Sreenivasulu, N., Weschke, W., Villmann, T., Hammer, B.: Generalized relevance LVQ (GRLVQ) with correlation measures for gene expression data. Neurocomputing 69, 651–659 (2006)
15. Strickert, M., Witzel, K., Mock, H.-P., Schleif, F.-M., Villmann, T.: Supervised attribute relevance determination for protein identification in stress experiments. In: Proc. of Machine Learning in Systems Biology (MLSB), pp. 81–86 (2007)
16. Sun, Y.: Iterative relief for feature weighting: Algorithms, theories, and applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(6), 1035–1051 (2007)
17. Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Advances in Neural Information Processing Systems 18, pp. 1473–1480. MIT Press, Cambridge (2006)
18. Xing, E., Ng, A., Jordan, M., Russell, S.: Distance metric learning, with application to clustering with side-information. In: Becker, S., Thrun, S., Obermayer, K. (eds.) Advances in Neural Information Processing Systems 15 (NIPS), pp. 505–512. MIT Press, Cambridge (2003)