Informative Discriminant Analysis

Samuel Kaski SAMUEL.KASKI@HUT.FI
Jaakko Peltonen JAAKKO.PELTONEN@HUT.FI
Helsinki University of Technology, Neural Networks Research Centre, P.O. Box 9800, FIN-02015 HUT, Finland

Abstract

We introduce a probabilistic model that generalizes classical linear discriminant analysis and gives an interpretation for the components as informative or relevant components of data. The components maximize the predictability of class distribution which is asymptotically equivalent to (i) maximizing mutual information with the classes, and (ii) finding principal components in the so-called learning or Fisher metrics. The Fisher metric measures only distances that are relevant to the classes, that is, distances that cause changes in the class distribution. The components have applications in data exploration, visualization, and dimensionality reduction. In empirical experiments the method outperformed a Renyi entropy-based alternative and linear discriminant analysis.

1. Introduction

Classical Linear Discriminant Analysis (LDA; see Timm, 2002) searches for directions or components in multivariate continuous-valued data that discriminate between classes. The components have traditionally been used for classification. They construct class borders that are Bayes-optimal for that purpose (in the two-class case), assuming the classes are normally distributed and share the same covariance matrix.

LDA has been used for visualizing multivariate data by projecting them to planes spanned by main discriminant direction pairs. An example will be presented on visualizing relationships in the behavior of genes of different functional classes in a set of knock-out mutation experiments. Another example could be to collect financial indicators from a set of companies, and visualize relationships of companies that will go bankrupt after 1 year, 2 years, etc. Such visualizations may reveal new subclasses or outliers. Our view to why such visualizations are useful is that discriminant analysis finds, intuitively speaking, directions that are *relevant* to or informative of the classification.

In data projection applications the classical LDA has two problems: (i) It is not optimal unless the data fulfills the restrictive assumption of normal distribution with equal covariance matrices in each class, and (ii) even if the assumption holds, the model is only optimal for *classification*, not for data projection.

Torkkola (Torkkola & Campbell, 2000; Torkkola, 2003) introduced an alternative method which relaxes the normality assumption. The optimization criterion is changed from discriminative power to the more flexible informativeness, measured by Renyi entropy-based mutual information. The work utilizes elegant formulas in (Fisher & Principe, 1998; Principe, Fisher & Xu, 2000) . Renyi-based formalism was claimed to be more suitable than the traditional Shannon entropy.

The Renyi-based projection has two potential weaknesses that we aim at removing. First, it is defined for probability distributions instead of data sets. We introduce a generative model for which standard probabilistic inference can be applied. This should be more justified for small data sets. The second potential weakness stems from using the Renyi entropy instead of Shannon's. We show that our criterion asymptotically maximizes the standard Shannon mutual information, which implies that it is not necessary to revert to Renyi entropy for computational reasons. The relative goodness of Shannon and Renyi entropy for projection will then be studied empirically.

The second goal of the paper, in addition to generalizing classical discriminant analysis, is to define more rigorously what it means for a projection to be relevant to the classes. The components of the proposed method can be asymptotically interpreted as kinds of principal components in so-called learning or Fisher

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

metrics, a framework used earlier for clustering and self-organizing maps (Kaski, Sinkkonen & Peltonen, 2001; Kaski & Sinkkonen, in press; Sinkkonen & Kaski, 2002).

2. Theory

Let $\mathbf{x}$ be multivariate vectors in the vector space $\mathbb{R}^n$. We seek to transform the $\mathbf{x}$ to smaller-dimensional vectors $\mathbf{y} = \mathbf{f}(\mathbf{x}) = \mathbf{W}^T\mathbf{x} \in \mathbb{R}^d$, where $\mathbf{W}$ is the orthogonal transformation matrix to be optimized. The columns $\mathbf{w}_i$ of $\mathbf{W}$ decompose the data, in the subspace they span, into *components* $\mathbf{w}_i^T\mathbf{x}$. The transformation is learned from a set of sample pairs $(\mathbf{x}, c)$, where the $\mathbf{x}$ are the primary data and the c are their classes.

The two key assumptions are that (i) analysis of the primary data is of main interest, and (ii) the classes are well-chosen such that variation in the $\mathbf{x}$ is relevant only to the extent it causes changes in the c. The goal is to make the transformation as informative as possible of the classes. The columns of the estimated transformation matrix $\mathbf{W}$ then represent the "informative" components of the primary data.

Note that the transformation does not depend on the classes. Once optimized, it can transform primary data without known classification.

2.1. Objective Function

Informativeness will be measured by predictive power, as the log-likelihood of a generative probabilistic model of c in the projection subspace. The prediction given the projected value $\mathbf{f}(\mathbf{x})$ is denoted by $\hat{p}(c|\mathbf{f}(\mathbf{x}))$. Maximizing the log-likelihood

$$L = \sum_{(\mathbf{x},c)} \log \hat{p}(c|\mathbf{f}(\mathbf{x})) \ , \tag{1}$$

is a well-defined criterion for fitting the transformation to the finite data set $\{(\mathbf{x}, c)\}$. The function L is to be maximized with respect to the projection matrix $\mathbf{W}$. In case parametric estimators of $\hat{p}$ are used, their parameters need to be optimized as well. Any parametric or non-parametric estimator in the projection space can be used; their relative goodness can be evaluated with standard methods of probabilistic model selection. In this paper we use non-parametric Parzen estimators.

A method for optimizing L for a reasonably general class of estimators $\hat{p}$ is presented in Section 3.

2.2. Properties of the Method

Connection to mutual information. Asymptotically, as the number of samples N increases (here $\mathbf{y} = \mathbf{f}(\mathbf{x})$), the objective function (1) becomes

$$\begin{aligned} &\frac{1}{N} L \underset{N\to\infty}{\longrightarrow} \sum_c \int p(c, \mathbf{x}) \log \hat{p}(c|\mathbf{f}(\mathbf{x})) d\mathbf{x} \\ &= I(C, Y) - E_{p(\mathbf{y})}[D_{KL}(p(c|\mathbf{y}), \hat{p}(c|\mathbf{y}))] - H(C) \ . \end{aligned} \tag{2}$$

(The constant $1/N$ on the first line has no effect on optimization.) The first term on the second line is the (true) mutual information between the auxiliary variable C having the values c and the projected primary variable Y having the values $\mathbf{y}$. The second term is the average estimation error, measured by the Kullback-Leibler divergence D_{KL}, of the classes after the projection. The term $H(C)$, the entropy of C, is constant.

Hence, maximization of the objective function of the generative model (1) is asymptotically equivalent to maximizing the mutual information and simultaneously minimizing the estimation error. The estimation error vanishes asymptotically for consistent estimators such as the Parzen estimators.

Connection to maximization of Renyi entropy. Torkkola and Campbell (2000) have introduced a closely related method, denoted here by MRMI for Maximization of Renyi Mutual Information. The main difference from our method is that instead of Shannon entropy, Torkkola and Campbell use Renyi quadratic entropy in defining the mutual information.

The second difference is in the estimation of the projection. We define the model for finite data as a generative (conditional) probability density model well suited for rigorous probabilistic inference. The connection to mutual information is asymptotic, which in our opinion is natural since mutual information is defined in terms of the (asymptotic) distributions.

By contrast, in MRMI an estimator (e.g. Parzen) of the projected joint density of the data is constructed, and the estimated mutual information of the projection and the class distribution is maximized. The possible theoretical problem seems to be that the objective function used for estimating the density is not directly related to the overall modeling goal, that is, maximization of mutual information. For Parzen estimators this problem is minimal, however. In Section 4 we compare the methods empirically.

Connection to linear discriminant analysis. In classical LDA, each class is assumed to be multinormal with the same covariance matrix in each class. For a two-class problem the direction in the data space that maximizes within-class variance while minimizing between-class variance is sought. The solution can be found by estimating the within- and between-class covariance matrices, and it is asymptotically optimal for classification if the assumptions hold. The solution can be generalized to multiple classes, by still maximizing between-class and minimizing within-class variance.

If the classes are multinormal, our method finds the same projection as LDA, at least under two assumptions in addition to the normal LDA assumptions: (i) the class centers reside within a d-dimensional subspace of the original space, if a d-dimensional projection is sought, and (ii) there is enough data, i.e., the result is asymptotic.

The proof is simple with these assumptions; a sketch is presented here. It is known (see for example Hastie and Tibshirani, 1996) that LDA maximizes the likelihood of a joint density model for the data and the classes, in the original data space. Each class is modeled by a separate Gaussian density. It is then evident that the conditional class density $p(c|\mathbf{x})$ of the optimal LDA model (and asymptotically of the data as well) is constant orthogonal to the d-dimensional subspace containing the class centers. This can be seen by factoring the density into two parts; the first depends only on the important d dimensions and the second only on the other dimensions. Our method, by comparison, builds a model $\hat{p}(c|\mathbf{f}(\mathbf{x}))$ for the conditional distribution that only varies within d dimensions, and the optimal solution clearly is to pick the dimensions where the densities really vary. The correct solution is reached if the probability estimator $\hat{p}(c|\mathbf{f}(\mathbf{x}))$ asymptotically finds the true distribution in the projection space, which holds at least for the nonparametric estimator we have used.

Connection to learning metrics. The learning metrics principle (Kaski, Sinkkonen & Peltonen, 2001; Kaski & Sinkkonen, in press; Sinkkonen & Kaski, 2002) formulates the idea of metrics where differences between data points are relevant only to the extent they cause changes in auxiliary data c, here the classes.

Distances d_L in the learning metric are asymptotically defined in terms of the conditional distributions of auxiliary data: local distances are Kullback-Leibler divergences D_{KL} between the distributions,

$$d_L^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) \equiv D_{KL}(p(c|\mathbf{x}), p(c|\mathbf{x} + d\mathbf{x})) \ , \qquad (3)$$

and global distances are minimal path integrals of the local distances.

Principal component analysis (PCA) minimizes the average reconstruction error, the (squared) Euclidean distance between the original data sample and its reconstruction. The proposed projection model has an (asymptotic) interpretation as a similar component model, but with a differently defined reconstruction and distance measure. It can be shown (Appendix A) that as the number of data grows, the mutual information in (2) approaches

$$I(C, Y) \approx -\int p(\mathbf{x}) d_L^2(\mathbf{x}, \mathbf{r}(\mathbf{f}(\mathbf{x}))) d\mathbf{x} + \text{const.} \ ,$$

the average squared distance d_L in the learning metric from samples $\mathbf{x}$ to their reconstructions $\mathbf{r}(\mathbf{f}(\mathbf{x}))$.

For consistent estimators $\hat{p}$ the second term in (2) asymptotically vanishes. Hence, optimizing the proposed cost function (1) is asymptotically approximately equivalent to minimizing the reconstruction error in learning metrics.

The definition and computation of the reconstruction is somewhat involved; it is not needed in practice and presented here only to define "relevance." The reconstruction $\mathbf{r}(\mathbf{f}(\mathbf{x}))$ of $\mathbf{f}(\mathbf{x})$ is defined to be the point in the primary data space that projects to $\mathbf{f}(\mathbf{x})$, and whose auxiliary distribution best matches that of the projection. Best match is defined by the smallest Kullback-Leibler divergence.

3. Optimization

To optimize the projection we need an estimator of $p(c|\mathbf{f}(\mathbf{x}))$, the conditional probability of auxiliary data in the projection space. The likelihood (1) can then be optimized by standard non-linear optimization methods, here stochastic approximation. We will next present a fairly general class of estimators and derive update rules for the parameters of a linear projection.

3.1. Estimation of Conditional Densities

In this paper we use standard Parzen estimators with Gaussian kernels for $p(c|\mathbf{f}(\mathbf{x}))$ but present them in a more general form:

$$\hat{p}(c|\mathbf{f}(\mathbf{x})) = \frac{G(\mathbf{f}(\mathbf{x}), c)}{\sum_{c'} G(\mathbf{f}(\mathbf{x}, c'))} \ . \qquad (4)$$

Here $G(\mathbf{f}(\mathbf{x}), c) = \sum_{m=1}^{M} \psi_{mc} g(\mathbf{f}(\mathbf{x}), m)$ is a weighted sum of M spherical Gaussian kernels

$$g(\mathbf{f}(\mathbf{x}), m) = \frac{1}{(2\pi\sigma^2)^{d/2}} e^{-\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{r}_m)\|^2 / 2\sigma^2} \ . \qquad (5)$$

The number of kernels M, the width σ of the Gaussians, the location parameters $\mathbf{r}_m$ and the weights ψ_{mc}

are parameters to be optimized. The weights must satisfy $0 < \psi_{mc} < 1$ and $\sum_{m,c} \psi_{mc} = 1$.

Both Parzen-type estimators with Gaussian windows and mixtures of Gaussians can be expressed with (4). For a Parzen-type estimator the probabilities are directly based on a learning data set $\{(\mathbf{x}_i, c_i)\}_{i=1}^N$ where the $\mathbf{x}_i$ are the primary data and the c_i the classes. Parzen estimators result from setting $M = N$, weights $\psi_{mc} = \delta_{c_m,c}/N$, where $\delta_{c_m,c}$ is one if $c_m = c$ and zero otherwise, and for the locations $\mathbf{r}_m = \mathbf{x}_m$. The only free parameter then is the width σ of the Gaussians which we will choose using a validation set.

For a mixture of Gaussians, M can be either fixed or validated. The ψ_{mc} and the $\mathbf{r}_m$ are to be optimized, and σ is either optimized or validated.

We have used nonparametric Parzen, although it can be slow for large data sets (using a subset helps). It has two advantages: (i) it is a *consistent* estimator of the conditional density that approaches the true value as the number of data grows and σ decreases, and (ii) there is no need to re-estimate when the projection changes. Mixtures of Gaussians would need to be re-estimated.

3.2. Optimization of the Projection by Stochastic Approximation

We optimize the likelihood (1) with respect to the projection $\mathbf{f}(\mathbf{x})$ by stochastic approximation. It is applicable to objective functions that are averages of another function. Here the average is taken of $L(\mathbf{x}, c) \equiv \log \hat{p}(c|\mathbf{f}(\mathbf{x}))$, that is, over the discrete distribution of the paired samples: $\frac{1}{N}L(\mathbf{W}) = \frac{1}{N}\sum_{(\mathbf{x},c)} L(\mathbf{x}, c; \mathbf{W})$. Under certain mild assumptions (Kushner & Yin, 1997) $L(\mathbf{W})$ can be optimized by iteratively moving towards the sample-specific gradient. At step t the update is

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \alpha(t)\frac{\partial L(\mathbf{x}, c; \mathbf{W})}{\partial \mathbf{W}} .$$

The step size has to fulfill the conditions $\sum \alpha(t) = \infty$ and $\sum \alpha^2(t) < \infty$. In practice the number of steps is finite and only an approximation to the optimum is obtained. It can be shown that the gradient is

$$\begin{aligned} &\frac{\partial}{\partial \mathbf{W}} L(\mathbf{x}, c; \mathbf{W}) \\ &= \frac{1}{\sigma^2}\left(E_{\xi(m|\mathbf{f}(\mathbf{x}))}\{(\mathbf{x} - \mathbf{r}_m)(\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{r}_m))^T\}\right. \\ &\left. - E_{\xi(m|\mathbf{f}(\mathbf{x}),c)}\{(\mathbf{x} - \mathbf{r}_m)(\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{r}_m))^T\}\right) . \quad (6) \end{aligned}$$

This is a difference between two cross-correlation-like matrices, one conditioned on the auxiliary data and the other without it. The operators $E_{\xi(m|\mathbf{f}(\mathbf{x}))}$ and $E_{\xi(m|\mathbf{f}(\mathbf{x}),c)}$ denote weighted sums over mixture components m, with respective weights

$$\xi(m|\mathbf{f}(\mathbf{x})) = \frac{\sum_{c'} \psi_{mc'} g(\mathbf{f}(\mathbf{x}), m)}{\sum_k \sum_{c'} \psi_{kc'} g(\mathbf{f}(\mathbf{x}), k)} \quad (7)$$

$$\xi(m|\mathbf{f}(\mathbf{x}), c) = \frac{\psi_{mc} g(\mathbf{f}(\mathbf{x}), m)}{\sum_k \psi_{kc} g(\mathbf{f}(\mathbf{x}), k)} . \quad (8)$$

The weighted sums are functionally similar to expectations over conditional distributions of m. However, the weights need not correspond to a maximum likelihood probability estimate.

For Parzen-type estimation, we made a minor improvement: if the stochastic sample has index m', this index is excluded from the sums over m in (6), (7), and (8). This results in a kind of implicit leave-one-out validation during learning. A similar adjustment does not affect the update of the MRMI method.

3.3. Orthogonality by Reparameterization

To keep the projection orthonormal, a straightforward addition is to reparameterize the matrix by Givens rotations. This yields $(n-d)d$ rotation angles to optimize, compared to nd matrix elements. A similar reparameterization was used by Torkkola & Campbell (2000).

The reparameterization is given by $\mathbf{W} = \mathbf{W}_0\left(\prod_{i=1}^d \left(\prod_{j=d+1}^n \mathbf{G}_{ij}\right)\right)\mathbf{W}_1$, where $\mathbf{G}_{ij}$ is a rotation matrix in the ij plane by an angle λ_{ij}. The angles are initially zero, and $\mathbf{W}_0$ is an initial rotation matrix whose first columns are in this paper an orthogonalized LDA projection. The matrix $\mathbf{W}_1$ simply selects the first d components after the rotations.

The (stochastic) gradient of a rotation angle is

$$\frac{\partial}{\partial \lambda_{ij}} L(\mathbf{x}, c; \mathbf{W}) = \sum_{k,l} \frac{\partial L}{\partial w_{kl}} \frac{\partial w_{kl}}{\partial \lambda_{ij}} , \quad (9)$$

where $\frac{\partial L}{\partial w_{kl}}$ is the element (k, l) of the matrix given by (6) and $\frac{\partial w_{kl}}{\partial \lambda_{ij}}$ can be easily calculated from the definition of the reparameterization.

3.4. The Algorithm

The stochastic approximation algorithm is finally as follows: after initializing the projection with LDA, repeat the following step for a fixed number of iterations t. At iteration t, pick a sample $(\mathbf{x}, c)$ at random and adjust the rotation angles by

$$\lambda_{ij}(t+1) = \lambda_{ij}(t) + \alpha(t)\frac{\partial}{\partial \lambda_{ij}} L(\mathbf{x}, c; \mathbf{W}) , \quad (10)$$

where the gradient on the right hand side is computed by (9). We used a piecewise linearly decreasing schedule for the $\alpha(t)$.

4. Comparisons

In this section we compare the proposed new projection method to the two other most closely related linear projections, MRMI and LDA. PCA is also included to provide a baseline; unlike the other methods, PCA does not use the classes.

4.1. Data

We compared the methods on five real-world data sets. LVQ_PAK denotes Finnish acoustic phoneme data from LVQ-PAK (Kohonen et al., 1996), and TIMIT denotes phoneme data from (TIMIT 1998); Landsat, Isolet and Multiple Features (MFeat) are from UCI Machine Learning Repository (Blake & Merz, 1998). Dimensionality of Isolet was reduced to 30 with PCA, and Fourier coefficient features were selected from MFeat.

We sought 5-dimensional projections except for Landsat having only six classes; for it the projection was 3-dimensional. Methods for choosing the "optimal" dimensionality will be studied later.

4.2. Quality Measure

A fair performance measure is needed to compare the methods. It cannot of course be the objective function of any of the methods. Since all aim to be discriminative we chose the classification error, measured with the simple non-parametric K *nearest neighbor* (KNN) classifier, working in the projection space. Note that minimization of the classification error is not the primary goal of any of the methods; hence the results give only indirect evidence.

To be precise, results were evaluated by error rate of KNN classification ($K = 5$) computed for a projected leave-out set with the neighbors picked from the learning set. 'Ties' yield partial error, e.g. 4/5 if the correct class and 4 others are tied.

4.3. Experimental Set-up

The quantitative comparison required three steps. First, value ranges were chosen for the initial learning rate α and width σ of the Gaussians: the α range was either hand-picked or expanded to find a local minimum of classification error on one of the data sets. The range of σ was logarithmic, roughly from the order of the nearest neighbor distance to the order of the largest pairwise distance. Second, the precise values that gave best validation results were picked. Lastly, the methods were compared with a cross-validation test where the data sets were redivided into 10 folds.

In all of the experiments, both our method and MRMI were optimized by 20,000 steps of stochastic approximation, starting from an LDA initialization.

Table 1. Difference of performance of the methods. The figures are average classification error rates over the ten folds, in percentages; the best result for each data set is shown in boldface. If a result is underlined, $P < 0.05$ for a two-tailed paired t-test between it and the boldfaced method, and $P < 0.01$ if doubly underlined.

Method	Landsat	LVQ_PAK	Isolet	MFeat	TIMIT
New	$\underline{14.70}$	**8.51**	**17.74**	**17.06**	**59.6**
MRMI	**13.34**	$\underline{\underline{10.25}}$	$\underline{\underline{29.44}}$	$\underline{\underline{20.89}}$	59.6
LDA	13.62	$\underline{\underline{10.51}}$	$\underline{\underline{28.79}}$	$\underline{\underline{21.08}}$	59.6
PCA	13.96	$\underline{9.60}$	$\underline{\underline{40.15}}$	$\underline{19.60}$	$\underline{\underline{64.1}}$

4.4. Quantitative Comparison

The statistical significance of the difference between our method and the best competitor was evaluated for each data set by a t-test of the 10-fold cross-validation results (Table 1).

Our method achieved the best average result for four data sets. The difference from the next best method was significant for three of the sets.

For Landsat data, all the other methods had similar performance; ours was surprisingly bad here, possibly because of noise in parameter validation.

5. Analysis of Gene Expression Data

In this section we demonstrate one way of using the extracted components for exploratory analysis of yeast gene expression.

The data set (Hughes et al., 2000) consists of measurements of the expression of each yeast gene in 300 knock-out mutation experiments. After leaving out all genes and experiments without significant expression, and genes with no known function, the resulting data set contained 702 genes measured in 179 experiments. The 46 functional classes were chosen from a standard MIPS functional classification.

It is well known that such large expression data sets are hard to analyze because they contain lots of both biological and measurement noise, and distortions in the measurements. Hence, this is a kind of worst-case case study.

The goals of the analysis are (i) to visualize mutual similarities and substructures of the functional classes, and (ii) to perform *feature exploration* to discover how the gene expression levels differentiate between functional classes of genes. Note that the goal is not to classify; it is known that hardly any classes are separable in such data. Instead, we want to explore properties and overlap of the classes.

To facilitate visualization on paper, we sought two components, with the dispersion parameter of the density estimator chosen using a validation set. Genes may belong to several classes; we treated such genes by dividing their "probability mass" equally to each class both in estimation and validation. In fact, almost all genes belong to many classes, which is reflected in the seemingly very high error rate: 205.5 of 234 validation samples (the lowest possible is here 109.9). LDA and PCA are still worse, with respective error rates 212.3 and 214.6, and even classification without dimensionality reduction only yields 191.3. The proposed method is significantly better (by the McNemar test) than a simple classifier predicting the largest class, which is already a result for this worst-case data.

Visualization of data by a scatter plot reveals properties of the class structure, such as mutual similarities between the functioning of genes. For example, since nitrogen and sulfur metabolism and amino acid metabolism-related genes are located relatively compactly and close to each other, they behave similarly in this set of experiments (Fig. 1**a**). Carbohydrate metabolism completes the continuum, although being even more widely distributed. The effect is not strong but the classes are clearly non-randomly distributed.

In contrast, the function of some classes such as organization of cytoplasm (Fig. 1**c**) is hardly reflected at all in this set of measurements. We verified with a standard KNN classifier that there do not exist components that could discriminate this class well: A 5-dimensional projection decreased the error rate of this class only marginally, from 94.4% to 92.5%, and even using all the components improved the error rate only to 88.1%.

Some classes have multimodal structure (Fig. 1**b**), suggesting existence of subclasses. Protein synthesis-related genes became divided into a cluster at the top, and more scattered data towards the bottom. All genes in the topmost cluster turned out to be mito-

(a)
(b)
(c)

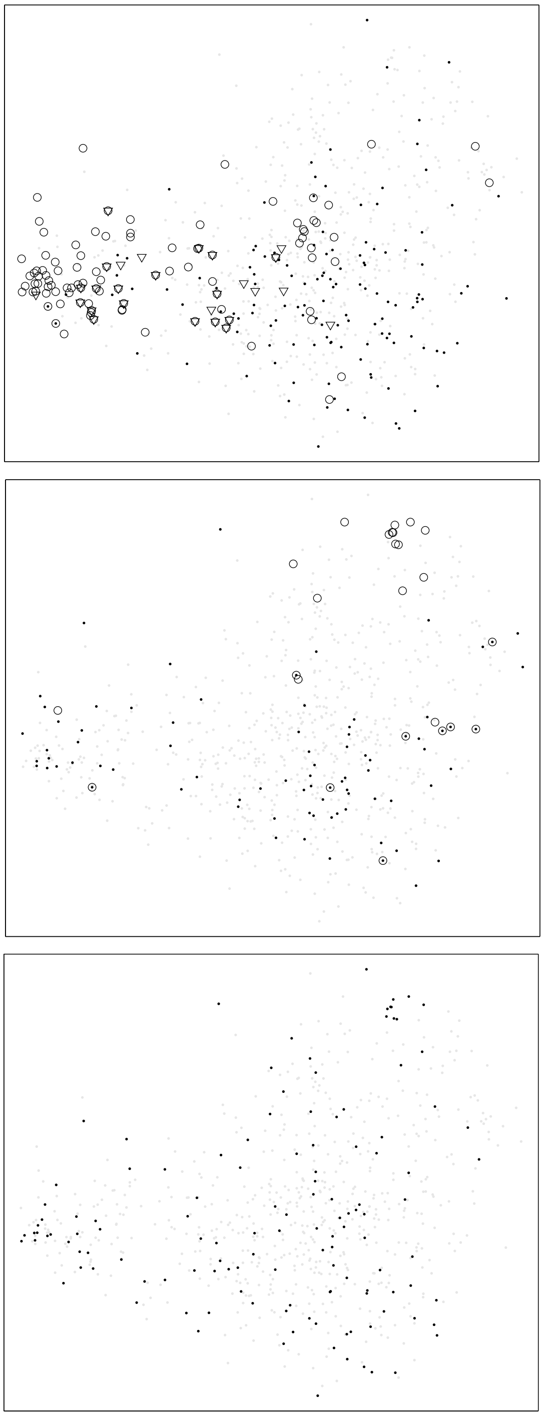

Figure 1. Two-dimensional discriminative projection of yeast gene expression data. Each gene is represented by a dot, and genes of different functional classes have been highlighted in different images. **(a)** C-compound and carbohydrate metabolism (black dot), amino-acid metabolism (circle) and nitrogen and sulfur metabolism (triangle). **(b)** mitochondrial organization (black dot) and protein synthesis (circle). **(c)** organization of cytoplasm (black dot).

chondrial ribosomal proteins. Different behavior of these and cytoplasmic ribosomal proteins (most of the rest) is biologically meaningful and known.

Feature exploration to characterize the classes. If a component discriminates one class from the others, it summarizes properties of the class. For example, the vertical axis in Figure 1**a** seems to characterize amino acid metabolism genes. The set of experiments (here dimensions) "contributing" most to the vertical component can then be listed (not shown) to summarize the behavior of these genes.

Both axes are required to characterize the protein synthesis class (Fig. 1**b**). The organization of cytoplasm class (Fig. 1**c**) is a different kind of example: The sought two components cannot describe it. Possible reasons for this are that (i) the dependency is non-linear, which would require an extension of the method, (ii) more than two components would be required, or (iii) the class is more or less randomly distributed in this data. For the difficult gene expression data the option (iii) is quite possible, as indicated by the classification errors of this class reported above.

In summary the visualizations, complemented with measuring the classification accuracy, suggest that the broad functional classes are not strongly differentially expressed in this knock-out data. However, although the effects are not strong some structure and meaningful overlap of classes can be found.

6. Discussion

Classical Linear Discriminant Analysis (LDA) was generalized to a linear probabilistic model for generating the class distribution. In contrast to LDA, normality assumptions about the class distribution are not needed. The model was shown to asymptotically maximize mutual information with the classes.

The model is applicable to dimensionality reduction, visualization, data exploration, and feature exploration. In such applications it is questionable whether the classification accuracy, the original goal of LDA, is a good criterion. An alternative is to measure relevance to the classes. It was argued that the proposed predictive power is a suitable measure of relevance, because (i) of its connection to mutual information, and (ii) the resulting components can be asymptotically interpreted as kinds of principal components in so-called learning or Fisher metrics.

In experiments, the model outperformed both the classical LDA and a Renyi entropy-based method. The method was lastly applied to gene expression analysis for visualizing overlap and substructures of functional classes of genes.

Only linear components relevant to (discrete) classes were considered. Extensions to non-linear projections and continuous auxiliary data in place of the classes will be studied. We suggest coining the more general task of finding components relevant to auxiliary data *relevant component analysis.*

The task is related to clustering discrete-valued data to maximize mutual information with another discrete variable by the information bottleneck principle (Tishby, Pereira & Bialek, 1999) and its extension to continuous-valued data (Sinkkonen & Kaski, 2002). In this paper a continuous projection is sought instead of the clustering. The work could in principle be extended to incorporate another random variable that indicates non-relevant variation such as normal biological noise, and minimize relevance to that variable, along the lines of (Chechik & Tishby, 2002).

We found out about a different kind of related work by Zhu and Hastie (2003) too late to include comprehensive comparisons here. They extend classical LDA by maximizing the likelihood ratio between class-specific and class-independent models. For non-parametric estimators the method is very close to ours. More generally, however, the difference is that we do not need an estimator of primary data densities—the conditional class probabilities suffice.

Acknowledgements

This work has been supported by the Academy of Finland, grants 50061 and 52123.

References

Blake, C. L., & Merz, C. J. (1998). UCI repository of machine learning databases. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Chechik, G., & Tishby, N. (2002). Extracting relevant structures with side information. In *Advances in neural information processing systems, NIPS 2002.* MIT Press. To appear.

Fisher III, J. W., & Principe, J. (1998). A methodology for information theoretic feature extraction. In A. Stuberud (Ed.), *Proceedings of IJCNN'98, International Joint Conference on Neural Networks*, vol. 3, 1712–1716. Piscataway, NJ: IEEE Service Center.

Hastie, T., & Tibshirani, R. (1996). Discriminant analysis by Gaussian mixtures. *Journal of the Royal Statistical Society B58*, 155–176.

Hughes, T. R., Marton, M. J., Jones, A. R., Roberts, C. J., Stoughton, R., Armour, C. D., Bennett, H. A., Coffrey, E., Dai, H., He, Y. D., Kidd, M. J., King, A. M., Meyer, M. R., Slade, D., Lum, P. Y., Stepaniants, S. B., Shoemaker, D. D., Gachotte, D., Chakraburtty, K., Simon, J., Bard, M., & Friend, S. H. (2000). Functional discovery via a compendium of expression profiles. *Cell*, *102*, 109–126.

Kaski, S., & Sinkkonen, J. (in press). Principle of learning metrics for data analysis. *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology, Special Issue on Data Mining and Biomedical Applications of Neural Networks*.

Kaski, S., Sinkkonen, J., & Peltonen, J. (2001). Bankruptcy analysis with self-organizing maps in learning metrics. *IEEE Transactions on Neural Networks*, *12*, 936–947.

Kohonen, T., Hynninen, J., Kangas, J., Laaksonen, J., & Torkkola, K. (1996). *LVQ_PAK: The learning vector quantization program package* (Technical Report A30). Helsinki University of Technology, Laboratory of Computer and Information Science, FIN-02150 Espoo, Finland.

Kushner, H. J., & Yin, G. G. (1997). *Stochastic approximation algorithms and applications*. New York: Springer.

Principe, J. C., Fisher III, J. W., & Xu, D. (2000). Information-theoretic learning. In S. Haykin (Ed.), *Unsupervised adaptive filtering*. New York, NY: Wiley.

Sinkkonen, J., & Kaski, S. (2002). Clustering based on conditional distributions in an auxiliary space. *Neural Computation*, *14*, 217–239.

TIMIT 1998. CD-ROM prototype version of the DARPA TIMIT acoustic-phonetic speech database.

Timm, N. H. (2002). *Applied multivariate analysis*. New York, NY: Springer.

Tishby, N., Pereira, F. C., & Bialek, W. (1999). The Information Bottleneck Method. *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing* (pp. 368–377). University of Illinois.

Torkkola, K. (2003). Feature Extraction by Non-Parametric Mutual Information Maximization. *Journal of Machine Learning Research*, *3*, 1415–1438.

Torkkola, K., & Campbell, W. (2000). Mutual information in learning feature transformations. *Proceedings of ICML'2000, the 17th International Conference on Machine Learning* (pp. 1015–1022). Stanford, CA, USA: Morgan Kaufmann.

Zhu, M., & Hastie, T. (2003) Feature extraction for non-parametric discriminant analysis. *Journal of Computational and Graphical Statistics*, *12*, 101–120.

Appendix A: Sketch of Proof of a Connection to Learning Metrics

The assumptions are that local approximations to the metrics are sufficient, and that good reconstruction points exist (for details, see the end).

Given a projected point $\mathbf{y}$, define $p(c,\mathbf{y}) = \int p(c,\mathbf{x})d\mathbf{x}$, where the integration is over all points projected to $\mathbf{y}$. Define the *reconstruction* $\mathbf{r}(\mathbf{y})$ to be the point projected to $\mathbf{y}$ that minimizes $D_{KL}(p(c|\mathbf{y}), p(c|\mathbf{r}(\mathbf{y})))$.

Asymptotically, omitting a few intermediate forms,

$$
\begin{aligned}
I(C,Y) &= \int p(\mathbf{y}) \sum_c p(c|\mathbf{y}) \log p(c|\mathbf{y}) d\mathbf{y} + H(C) \\
&= -E_{p(\mathbf{x})}[D_{KL}(p(c|\mathbf{x})||p(c|\mathbf{r}(\mathbf{y})))] \\
&+ E_{p(\mathbf{y})}[D_{KL}(p(c|\mathbf{y}), p(c|\mathbf{r}(\mathbf{y})))] + H(C) - H(C|X) \ .
\end{aligned}
$$

Assuming distances are local, this further equals

$$
\begin{aligned}
I(C,Y) \approx & -E_{p(\mathbf{x})}[d_L^2(\mathbf{x}, \mathbf{r}(\mathbf{f}(\mathbf{x})))] \\
& + E_{p(\mathbf{y})}[D_{KL}(p(c|\mathbf{y}), p(c|\mathbf{r}(\mathbf{y})))] + I(C,X) \ , \quad (11)
\end{aligned}
$$

where the last term is constant.

The divergence term (middle term in eqn 11) is zero trivially if the distribution of auxiliary data is constant in the direction orthogonal to the subspace. This holds approximately if the data is local in the orthogonal direction, which is likely to hold if the projection dimensionality is large.

If the middle term is not (nearly) constant, the proposed algorithm does not minimize the reconstruction error. If the goal is not to maximize mutual information but to minimize the reconstruction error, it can in principle be done by minimizing

$$
\begin{aligned}
E_{p(\mathbf{x})}[d_L^2(\mathbf{x}, \mathbf{r}(\mathbf{f}(\mathbf{x})))] \approx & -I(C,Y) \\
& + E_{p(\mathbf{y})}[D_{KL}(p(c|\mathbf{y}), p(c|\mathbf{r}(\mathbf{y})))] + \text{const.}
\end{aligned}
$$