LEIBNIZ CENTER FOR RESEARCH IN COMPUTER SCIENCE
TECHNICAL REPORT 2003-34

Learning a Mahalanobis Metric with Side Information

Aharon Bar-Hillel, Tomer Hertz, Noam Shental, and Daphna Weinshall
School of Computer Science and Engineering and the Center for Neural Computation
The Hebrew University of Jerusalem, Jerusalem, Israel 91904
Email: {aharonbh,tomboy,fenoam,daphna}@cs.huji.ac.il

Abstract

Many learning algorithms use a metric defined over the input space as a principal tool, and their performance critically depends on the quality of this metric. We address the problem of learning metrics using side-information in the form of equivalence constraints. Unlike labels, we demonstrate that this type of side-information can sometimes be automatically obtained without the need of human intervention. We show how such side-information can be used to modify the representation of the data, leading to improved clustering and classification.

Specifically, we present the Relevant Component Analysis (RCA) algorithm, which is a simple and efficient algorithm for learning a full ranked Mahalanobis metric. We show that RCA is the solution of an interesting optimization problem, founded on an information theoretic basis. If the Mahalanobis matrix is allowed to be singular, we show that Fisher's linear discriminant followed by RCA is the optimal dimensionality reduction algorithm under the same criterion. Moreover, under certain Gaussian assumptions, RCA can be viewed as an ML estimation of the inner class covariance matrix. We conclude with extensive empirical evaluations of RCA, showing its advantage over alternative methods.

Keywords: clustering, learning from partial knowledge, metric learning, Mahalanobis metric, dimensionality reduction, side information.

1 Introduction

A number of learning problems, such as clustering and nearest neighbor classification, rely on some apriori defined distance function over the input space. In many of these problems it is often the case that selecting a "good" metric critically affects the algorithms' performance. In this paper, motivated by the wish to boost the performance of these algorithms, we study ways to learn a "good" metric using side information.

One difficulty in finding a "good" metric is that its quality may be context dependent. For example, consider an image retrieval application which includes many facial images. Given a query image, the application retrieves the most similar images in the database according to some pre-determined metric. However, when presenting the query image we may be interested in retrieving other images of the same person, or

we may want to retrieve other faces with the same facial expression. It seems difficult for a pre-determined metric to be suitable for two such different tasks.

In order to learn a context dependent metric, the data set must be augmented by some additional information, or side-information, relevant to the task at hand. For example we may have access to the labels of *part* of the data set. In this paper we focus on another type of side-information, in which *equivalence constraints* between a few of the data points are provided. More specifically we assume knowledge about small groups of data points that are known to originate from the same class, although their label is unknown. We term these small groups of points *"chunklets"*.

A key observation is that in contrast to explicit labels that are usually provided by a human instructor, in many unsupervised learning tasks equivalence constraints may be extracted with minimal effort or even automatically. One example is when the data is inherently sequential and can be modeled by a Markovian process. Consider for example movie segmentation, where the objective is to find all the frames in which the same actor appears. Due to the continuous nature of most movies, faces extracted from successive frames in roughly the same location can be assumed to come from the same person. This is true as long as there is no scene change, which can be automatically and robustly detected (Boreczky & Rowe, 1996). Another analogous example is speaker segmentation and recognition, in which the conversation between several speakers needs to be segmented and clustered according to speaker identity. Here, it may be possible to automatically identify small segments of speech which are likely to contain data points from a single yet *unknown* speaker.

A different scenario, in which equivalence constraints are the natural source of training data, occurs when we wish to learn from several teachers who do not know each other and who are not able to coordinate among themselves the use of common labels. We call this scenario 'distributed learning'.[1] To illustrate, assume that you are given a large database of facial images of many people, which cannot be labeled by a small number of people due to its vast size. The database is therefore divided (arbitrarily) into P parts (where P is very large), which are then given to P teachers to annotate. The labels provided by the different teachers may be inconsistent: as images of the same person appear in more than one part of the database, they are likely to be given different names. Coordinating the labels of the different teachers is almost as daunting as labeling the original dataset. However, equivalence constraints can be easily extracted, since points which were given the same tag by a certain teacher are known to originate from the same class.

In this paper we study how to use equivalence constraints in order to learn an optimal Mahalanobis metric between data points. Equivalently the problem can also be posed as learning a good representation function, transforming the data representation by the square root of the Mahalanobis weight matrix. Therefore we shall discuss the two problems interchangeably.

In Section 2 we describe the proposed method – the Relevant Component Analysis (RCA) algorithm. In the subsequent three sections we show how RCA can be derived in parallel from three different perspectives: In Section 3 we describe a novel information theoretic criterion and show that RCA is its optimal solution. Moreover, if dimensionality reduction is permitted, the optimal solution is Fisher's linear discriminant (Fukunaga, 1990) followed by RCA. In Section 4 we show that RCA is also the optimal solution to another optimization problem, seeking to minimize inner class distances. Viewed this way, RCA is directly compared to another recent algorithm for learning Mahalanobis distance from equivalence constraints (proposed in (Xing et al., 2002)). In Section 5 we show that under Gaussian assumptions RCA can be interpreted as the maximum-likelihood (ML) estimator of the within-class covariance matrix. We also provide a bound over the variance of this estimator, showing that it is at most twice the variance of the ML estimator obtained using labeled data.

The successful application of RCA in high dimensional space requires dimensionality reduction, whose

[1] A related scenario (which we call 'generalized relevance feedback'), where users of a retrieval engine are asked to annotate the retrieved set of data points, has similar properties.

details are discussed in Section 6. An online version of the RCA algorithm is presented in Section 7. In Section 8 we describe an extensive empirical evaluation of the RCA method. We focused on two tasks - data retrieval and clustering. We used three types of data: (a) A data set of frontal faces (Belhumeur et al., 1997); this example shows that RCA with partial equivalence constraints typically yields comparable results to supervised algorithms which use fully labeled training data. (b) A large data set of images collected by a real-time surveillance application, where the equivalence constraints are gathered automatically. (c) Several data sets from the UCI repository, which are used to compare between RCA and other competing methods that use equivalence constraints.

Related work

There has been much work on learning representations and distance functions in the supervised learning settings, and we can just briefly mention some examples. (Hastie & Tibshirani, 1996) and (Jaakkola & Haussler, 1998) use labeled data to learn good metrics for classification. In (Thrun, 1996) a distance function (or a representation function) is learned for classification using a "leaning-to-learn" paradigm. In this setting several related classification tasks are learned using several labeled data sets, and algorithms are proposed which learn representations and distance functions in a way that allows for the transfer of knowledge between the tasks. In (Tishby et al., 1999) the joint distribution of two random variables X and Y is assumed to be known, and the problem is reduced to the learning of a compact representation of X which bears high relevance to Y. This work, which is further developed in (Chechik & Tishby, 2002), can be viewed as supervised representation learning. Information theoretic criteria for unsupervised learning in neural networks were first suggested by (Linsker, 1989), and has been used since in several tasks in the neural network literature, e.g., (Bell & Sejnowski, 1995).

In recent years some work has been done on using equivalence constraints as side information. In (Wagstaff et al., 2001) equivalence relations were introduced into the K-means clustering algorithm. Both positive ('a is similar to b') and negative ('a is dissimilar from b') relations were used. An initial description of RCA using positive equivalence constraints was given in (Shental et al., 2002), in the context of image retrieval, and expanded in (Bar-Hilel et al., 2003). Learning a Mahalanobis metric from positive and negative equivalence constraints was addressed in (Xing et al., 2002), in conjunction with the constrained K-means algorithm. We compare this algorithm to our current work in Section 4, and compare our empirical results with the results of both algorithms in Section 8. We have also recently developed a way to introduce both positive and negative equivalence constraints into the EM algorithm for the estimation of a mixture of Gaussian models (Shental et al., 2003).

2 Relevant Component Analysis: the algorithm

Relevant Component Analysis (RCA) is a method that seeks to identify and down-scale global unwanted variability within the data. The method changes the feature space used for data representation, by a global linear transformation which assigns large weights to "relevant dimensions" and low weights to "irrelevant dimensions" (cf. (Tenenbaum & Freeman, 2000)). These "relevant dimensions" are estimated using *chunklets*, i.e, small subsets of points that are known to belong to the same although *unknown* class.

More specifically, points x_1 and x_2 are said to be related by a positive constraint if it is known that both points share the same (unknown) label. If points x_1 and x_2 are related by a positive constraint, and x_2 and x_3 are also related by a positive constraint, then a chunklet $\{x_1, x_2, x_3\}$ is formed. Generally, chunklets are formed by applying transitive closure over the whole set of positive equivalence constraints.

The algorithm is presented below as Algorithm 1. The RCA transformation is intended to reduce clutter, so that in the new feature space, the inherent structure of the data can be more easily unraveled (see illus-

trations in Figs. 1a-f). Thus the whitening transformation W (in step 3 of Alg. 1) assigns lower weights to directions of large variability, since this variability is mainly due to within class changes and is therefore "irrelevant" for the task of classification. RCA can be used as a preprocessing step for both unsupervised clustering of the data and nearest neighbor classification.

Algorithm 1 The RCA algorithm

Given a dataset $\{x_i\}_{i=1}^{N}$ and k chunklets $C_j = \{x_{ji}\}_{i=1}^{n_j} \quad j = 1 \dots k, \quad$ do

1. For each chunklet C_j, subtract the chunklet's mean from all the points it contains (Fig. 1d).

2. Compute the covariance matrix of all the centered data-points in chunklets (Fig. 1d). Assume a total of p points in k chunklets, where chunklet C_j consists of points $\{x_{ji}\}_{i=1}^{n_j}$ and its mean is $\hat{m}_j$. RCA computes the following matrix:

$$\hat{C} = \frac{1}{p} \sum_{j=1}^{k} \sum_{i=1}^{n_j} (x_{ji} - \hat{m}_j)(x_{ji} - \hat{m}_j)^t \tag{1}$$

3. Compute the whitening transformation $W = \hat{C}^{-\frac{1}{2}}$ associated with this covariance matrix (Fig. 1e), and apply it to the original data points: $x_{new} = W x$ (Fig. 1f). Alternatively, use the inverse of $\hat{C}$ as a Mahalanobis distance.

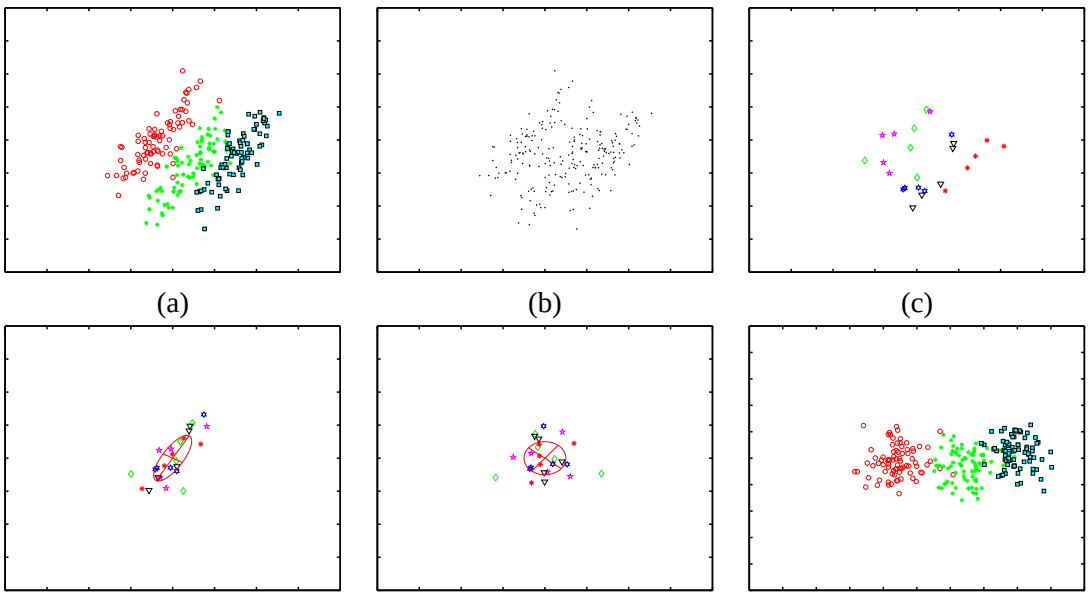

Figure 1: An illustrative example of the RCA algorithm applied to synthetic Gaussian data. (a) The fully labeled data set with 3 classes. (b) Same data unlabeled; clearly the classes' structure is less evident. (c) The set of chunklets that are provided to the RCA algorithm (points that share the same color and marker type form a chunklet). (d) The centered chunklets, and their empirical covariance. (e) The whitening transformation applied to the chunklets. (f) The original data after applying the RCA transformation.

The following sections present theoretical justifications for the RCA algorithm.

3 Information maximization with chunklet constraints

How do we find the optimal transformation of the data which uses chunklets to improve its representation value? In Section 3.1 we state the problem for general families of transformations, presenting an information theoretic formulation. In section 3.2 we restrict the family of transformation to non-singular linear maps, showing that the optimal solution is given by RCA. In section 3.3 we widen the family of permitted transformations to include non-invertible linear transformations, showing (for normally distributed data) that the optimal transformation is given by Fisher's Linear Discriminant (FLD) followed by RCA.

3.1 An information theoretic perspective

Following (Linsker, 1989), an information theoretic criterion states that an optimal transformation of the input X into its new representation Y, should seek to maximize the mutual information $I(X,Y)$ between X and Y under suitable constraints. In the general deterministic case a set $X = \{x_l\}_{l=1}^n$ of data points in $\mathcal{R}^N$ is transformed into the set $Y = \{f(x_l)\}_{l=1}^n$ of points in $\mathcal{R}^M$. We wish to find a function $f \in F$ that maximizes $I(X,Y)$, where F is the family of permitted transformation functions (a "hypotheses family").

First, note that $I(X,Y)$ is the *differential* mutual information as X and Y are continuous variables. Moreover since f is deterministic, maximizing $I(X,Y)$ is achieved by maximizing the entropy $H(Y)$ alone. To see this, recall that

$$I(X,Y) = H(Y) - H(Y|X)$$

Since f is deterministic, the uncertainty concerning Y when X is known is minimal, thus $H(Y|X)$ achieves its lowest possible value at $-\infty$.[2] However, as noted in (Bell & Sejnowski, 1995), $H(Y|X)$ does not depend on f and is constant for every finite quantization scale. Hence maximizing with respect to f can be done by considering only the first term $H(Y)$.

Second, note also that $H(Y)$ can be increased by simply 'stretching' the data space (for example by choosing $f = \lambda x$, where $\lambda > 1$). Therefore, in order to avoid the trivial solution $\lambda \to \infty$, we constrain the distance between points contained in a single chunklet, not to exceed a fixed threshold. This gives us the following optimization problem:

$$\max_{f \in F} I(X,Y) \quad s.t. \ \frac{1}{p}\sum_{j=1}^{p}\sum_{i=1}^{n_j} ||y_{ji} - \hat{m}_j^y||^2 \leq K \tag{2}$$

where $\{y_{ji}\}_{j=1,i=1}^{p\ ,\ n_j}$ denote the set of points in p chunklets after the transformation, $\hat{m}_j^y$ denotes the mean of the points in chunklet j after the transformation, and K denotes some constant (the fixed threshold). From the discussion above, (2) can be simplified to the following optimization problem, which henceforth we will try to solve:

$$\max_{f \in F} H(Y) \quad s.t. \ \frac{1}{p}\sum_{j=1}^{p}\sum_{i=1}^{n_j} ||y_{ji} - \hat{m}_j^y||^2 \leq K \tag{3}$$

3.2 RCA: the solution to the optimization problem

Consider the problem posed in (3) for the family F of invertible linear transformations. Since f is invertible, the connection between the densities of $Y = f(X)$ and X is expressed by $p_y(y) = \frac{p_x(x)}{|J(x)|}$, where $|J(x)|$ is

[2] This non-intuitive divergence is a result of the generalization of information theory to continuous variables, i.e., it is the result of ignoring the discretization constant in the definition of differential entropy.

the Jacobian of the transformation. From $p_y(y)dy = p_x(x)dx$, it follows that $H(Y)$ and $H(X)$ are related as follows:

$$H(Y) = -\int_y p(y)\log p(y)dy = -\int_x p(x)\log\frac{p(x)}{|J(x)|}dx = H(X) + \langle\log|J(x)|\rangle_x$$

For the linear map $Y = AX$ the Jacobian is constant and equals $|A|$, and it is the only term in $H(Y)$ that depends on the transformation A. Hence problem (3) is reduced to

$$\max_A log|A| \quad s.t.\ \frac{1}{p}\sum_{j=1}^{k}\sum_{i=1}^{n_j}||y_{ji} - \hat{m}_j^y||^2 \leq K \tag{4}$$

Let $B = A^tA$; since B is positive definite and $\log|A| = \frac{1}{2}\log|B|$, (4) can be rewritten as

$$\max_{B>0} log|B| \quad s.t.\ \frac{1}{p}\sum_{j=1}^{k}\sum_{i=1}^{n_j}||x_{ji} - \hat{m}_j||_B^2 \leq \mathrm{K} \tag{5}$$

where $||.||_B$ denotes the Mahalanobis distance with weight matrix B.

The optimization problem in (5) can be solved easily, since the constraint is linear in B. The solution is $B = \frac{K}{N}\hat{C}^{-1}$, where $\hat{C}$ is the average chunklet covariance matrix (1) and N is the dimensionality of the data space. This solution is identical to the Mahalanobis matrix in RCA up to a global scale factor,[3] and hence RCA is a scaled solution of (5).

3.3 Dimensionality reduction

We now solve the optimization problem in (5) with the family of general linear transformations, i.e., $Y = AX$ where $A \in \mathcal{M}_{M\times N}$ and $M \leq N$. To simplify the analysis, we assume that the distribution of X is multivariate Gaussian. Since X is assumed to be Gaussian, Y is also Gaussian with the following entropy

$$H(Y) \quad = \quad \frac{d}{2}\log 2\pi e + \frac{1}{2}\log|\Sigma_y| = \frac{d}{2}\log 2\pi e + \frac{1}{2}\log|A\Sigma_x A^t|$$

Now (3) becomes

$$\max_A \log|A\Sigma_x A^t| \quad s.t.\ \frac{1}{p}\sum_{j=1}^{k}\sum_{i=1}^{n_j}||x_{ji} - m_j||_{A^tA}^2 \leq K \tag{6}$$

For a given target dimensionality M, the solution to the problem is Fisher linear discriminant (FLD) followed by RCA in the reduced space. A sketch of the proof is given in Appendix 10. Notice that after the FLD the inner covariance matrix in the reduced space is diagonal, and so RCA only scales each dimension separately. The computation of FLD based on equivalence constraints (cFLD) is described in Section 6.

4 RCA and the minimization of inner class distances

In order to gain some intuition to the solution provided by the information maximization criterion of Eq. (3), let us look at the optimization problem obtained by reversing the roles of the maximization term and the

[3] Such a global scale constant is not important in most clustering and classification tasks, which essentially rely on relative distances.

constraint term in (5):

$$\min_B \frac{1}{p} \sum_{j=1}^{k} \sum_{i=1}^{n_j} ||x_{ji} - m_j||_B^2 \quad s.t. \ |B| \geq 1 \tag{7}$$

We interpret (7) as follows: a Mahalanobis distance B is sought, which minimizes the sum of all inner chunklet squared distances, while $|B| \geq 1$ prevents the solution from being achieved by "shrinking" the entire space. Using the Kuhn-Tucker theorem, we can reduce (7) to

$$\min_B \sum_{j=1}^{k} \sum_{i=1}^{n_j} ||x_{ji} - m_j||_B^2 - \lambda \log |B| \quad s.t. \ \lambda \geq 0, \ \ \lambda \log |B| = 0 \tag{8}$$

Differentiating the Lagrangian above shows that the minimum is given by $B = |\hat{C}|^{\frac{1}{d}} \hat{C}^{-1}$, where C is the average chunklet covariance matrix. Once again, the solution is identical to the Mahalanobis matrix in RCA up to a scale factor.

It is interesting, in this respect, to compare RCA with the method proposed recently by (Xing et al., 2002). They also consider the problem of learning a Mahalanobis distance using side information in the form of pairwise constraints.[4] They assume the knowledge of a set S of pairs of points known to be similar, and a set D of pairs of points known to be dissimilar. Given these sets, they pose the following optimization problem.

$$\min_B \sum_{(x_1,x_2) \in S} ||x_1 - x_2||_B^2 \qquad s.t. \sum_{(x_1,x_2) \in D} ||x_1 - x_2||_B \geq 1, \quad B \geq 0 \tag{9}$$

This problem is solved using gradient ascent and iterative projection methods.

To allow a clearer comparison of RCA with Eq. (9), we reformulate the argument of (7) in terms of inner chunklet pairwise distances. For each point x_{ji} in chunklet j we have:

$$x_{ji} - m_j = x_{ji} - \frac{1}{n_j} \sum_{k=1}^{n_j} x_{jk} = \frac{1}{n_j} \sum_{\substack{k=1 \\ k \neq i}}^{n_j} (x_{ji} - x_{jk})$$

Problem (7) can now be rewritten as

$$\min_B \sum_{j=1}^{k} \frac{1}{n_j^2} \sum_{i=1}^{n_j} || \sum_{k \neq i} (x_{ji} - x_{jk}) ||_B^2 \quad s.t. \ |B| \geq 1 \tag{10}$$

When only chunklets of size 2 are given (as in the case studied in (Xing et al., 2002)), the problem reduces to

$$\min_B \frac{1}{2} \sum_{j=1}^{k} ||x_{j1} - x_{j2}||_B^2 \quad s.t. \ |B| \geq 1 \tag{11}$$

Clearly the minimization terms in problems (11) and (9) are identical up to a constant ($\frac{1}{2}$). The difference between the two problems lies in the constraint term: the constraint proposed in (Xing et al., 2002) uses pairs of dissimilar points, whereas the constraint in the RCA formulation affects global scaling so that the 'volume' of the Mahalanobis neighborhood is not allowed to shrink indefinitely. This difference has the immediate effect that the algorithm described in (Xing et al., 2002) to solve (9) is substantially slower and does not always converge. In contrast, the RCA distance computation is simple and fast (requiring a single matrix inversion) without the need for costly iterative procedures.

[4] Chunklets of size > 2 are not considered in (Xing et al., 2002).

5 RCA and Maximum Likelihood: the effect of chunklet size

We now consider the case where the data consists of several normally distributed classes sharing the same covariance matrix. Under the assumption that the chunklets are sampled i.i.d. and that points within each chunklet are also sampled i.i.d., the likelihood of the chunklets' distribution can be written as:

$$\prod_{j=1}^{k}\prod_{i=1}^{n_j}\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\tfrac{1}{2}(x_{ji}-m_j)^t\Sigma^{-1}(x_{ji}-m_j)\right) \tag{12}$$

Writing the log-likelihood while neglecting constant terms and denoting $B=\Sigma^{-1}$, we obtain:

$$\sum_{j=1}^{k}\sum_{i=1}^{n_j}||x_{ji}-m_j||_B^2 - p\log|B| \tag{13}$$

where p is the total number of points in chunklets. Maximizing the log-likelihood is equivalent to minimizing (13), whose minimum is obtained when B equals the RCA Mahalanobis matrix from Eq. (1). Note, moreover, that (13) is rather similar to the Lagrangian in (8), where the Lagrange multiplier is replaced by the constant p. Hence, under Gaussian assumptions, the solution of problem (8) is probabilistically justified by a maximum likelihood formulation.

Under Gaussian assumptions, we can define an *unbiased* version of the RCA estimator. Assume for simplicity that there are p constrained data points divided into n chunklets of size k each. The *unbiased* RCA estimator can be written as:

$$\hat{C}(n,k)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{k-1}\sum_{j=1}^{k}(x_i^j-\hat{m}_i)(x_i^j-\hat{m}_i)^t \tag{14}$$

where x_i^j denotes the data point j in chunklet i, and $\hat{m}_i$ denotes the empirical mean of chunklet i. $\hat{C}(n,k)$ in (14) is the empirical mean of the covariance estimators produced by each chunklet. It is shown in Appendix 10 that the variance of the elements $\hat{C}_{ij}$ of the estimating matrix is bounded by

$$Var(\hat{C}_{ij}(n,k)) \le (1+\frac{1}{k-1})Var(\hat{C}_{ij}(1,nk)) \tag{15}$$

where $\hat{C}_{ij}(1,nk)$ is the estimator when all the $p=nk$ points are known to belong to the same class, thus forming the best estimate possible from p points. This bound shows that the variance of the RCA estimator rapidly converges to the variance of the best estimator, even for small chunklets.

6 Dimensionality reduction

As noted in Section 2 the first step in RCA is usually dimensionality reduction. We now turn to address this issue in detail. The RCA algorithm decreases the weight of principal directions along which the inner class covariance is relatively high, and increases the weight of directions along which it is low. This intuition can be made precise in the following sense:

Denote by $\{\lambda_i\}_{i=1}^{D}$ the eigenvalues of the inner covariance matrix, and consider the squared distance between two points from the same class $||x^1-x^2||^2$. We can diagonalize the inner covariance matrix using an orthonormal transformation which does not change the distance. Therefore let us assume without loss of generality that the inner covariance is diagonal.

Before whitening, the average squared distance is $E[||x^1 - x^2||^2] = 2\sum_{j=1}^{D}\lambda_j$ and the average squared distance in direction i is $E[(x_i^1 - x_i^2)^2] = 2\lambda_i$. After whitening these values are $2D$ and 2, respectively. Let us define the weight of dimension i, $W(i) \in [0,1]$, as

$$W(i) = \frac{E[(x_i^1 - x_i^2)^2]}{E[||x^1 - x^2||^2]}$$

Now the ratio between the weight of each dimension before and after whitening is given by

$$\frac{W_{before}(i)}{W_{after}(i)} = \frac{\lambda_i}{\frac{1}{D}\sum_{j=1}^{D}\lambda_j} \tag{16}$$

In (16) the weight of each principal dimension increases if its initial variance was lower than the average, and vice versa. When there is high irrelevant noise along several dimensions, the algorithm will indeed scale down noise dimensions. However, when the irrelevant noise is scattered among many dimensions with low amplitude in each of them, Whitening will amplify these noisy dimensions, which is potentially harmful. Therefore, when the data is initially embedded in high dimensional space, dimensionality reduction must precede RCA to avoid this problem.

We have seen in section 3.3 that FLD is the dimensionality reduction technique which maximizes the mutual information under Gaussian assumptions. Traditionally FLD is computed from fully labeled training data, and the method therefore falls within supervised learning. However, it is easy to extend classical FLD to the case of partial supervision in the form of equivalence constraints. Specifically, let S_t and S_w denote the total covariance and the inner class covariance respectively. FLD maximizes the following determinant ratio

$$\max_{A \in M_{k \times D}} \frac{A^t S_t A}{A^t S_w A}$$

by solving the appropriate generalized eigenvector problem. With chunklets, we can use the inner chunklet covariance matrix from (1) to approximate S_w, and compute the projection matrix is the usual way. We term this FLD variant cFLD (Constraints based FLD).

The cFLD dimensionality reduction can only be used if the rank of the inner chunklet covariance matrix is higher than the dimensionality of the initial data space. If this condition doesn't hold, we use PCA to reduce the original data dimensionality as needed. The detailed procedure is summarized below in Algorithm 2. We compare this dimensionality reduction scheme with simple PCA in Section 8.1.4.

Algorithm 2 RCA's dimensionality reduction

Denote by d the original data dimensionality.
Given a set of chunklets $\{C_i\}_{i=1}^{K}$ do

1. Compute the rank of the estimated covariance matrix $R = \sum_{i=1}^{K}(|c_i| - 1)$.
2. If ($d > R$), apply PCA to reduce the data dimensionality to αR, where $0 < \alpha < 1$ (to ensure that cFLD provides stable results).
3. Apply cFLD to achieve the target data dimensionality.

7 Online implementation of RCA

The standard RCA algorithm presented in Section 2 is a batch algorithm which assumes that all the equivalence constraints are available at once. Here we briefly present an alternative online implementation of RCA, suitable for a neural-network-like architecture. In this implementation a weight matrix $W \in M_{D\times D}$, initiated randomly, is gradually developed to become the RCA transformation matrix. The algorithm is presented below in Algorithm 3.

Algorithm 3 Online RCA

Input: a stream of pairs of points (x_1, x_2), where the points in a pair are known to belong to the same class.
initialize W to a symmetric random matrix with $||W|| << 1$.
For t=1:T do

- receive pair x_1^t, x_2^t;
- take the difference between the points, $h = x_1 - x_2$;
- transform h using W, to get $y = Wh$;
- update $W = W + \eta(W - yy^tW)$.

where η determines the step size.

The steady state of this stochastic process can be found by equating the mean update to 0, where the mean is taken over the next example pair (x_1, x_2):

$$\begin{aligned} & E[\eta(W - yy^tW)] = 0 \\ \Rightarrow \quad & E[I - yy^t] = I - W^tE[hh^t]W = 0 \\ \Rightarrow \quad & W = E[hh^t]^{-\frac{1}{2}}P \end{aligned}$$

Where P is an orthogonal marix $P^tP = 1$. The steady state W is a whitening transformation of the correlation matrix of h. Since $h = 2(x_1 - \frac{(x_1+x_2)}{2})$, it is equivalent (up to the constant 2) to the distance of a point from the center of its chunklet. The correlation matrix of h is therefore equivalent to the inner chunklet covariance matrix. Thus W converges to the RCA transformation of the input population up to an orthogonal transformation. The resulting transformation is geometrically equivalent to RCA, since the orthogonal transformation P preserves vector norms and angles.

8 Experimental Results

The success of the RCA algorithm can be measured directly by measuring neighborhood statistics, or indirectly by measuring whether it improves clustering results. In the following we tested RCA on three different applications using both direct and indirect evaluations.

The RCA algorithm uses only partial information about the data labels. In this respect it is interesting to compare its performance to unsupervised and supervised methods for data representation. Section 8.1 compares RCA to the unsupervised PCA and the fully supervised FLD on a facial recognition task, using the YaleB data set (Belhumeur et al., 1997). In this application of face recognition, RCA appears very efficient in eliminating irrelevant variability caused by varying illumination. We also used this data set to test the effect of dimensionality reduction using cFLD, and the sensitivity of RCA to average chunklet size and the total amount of points in chunklets.

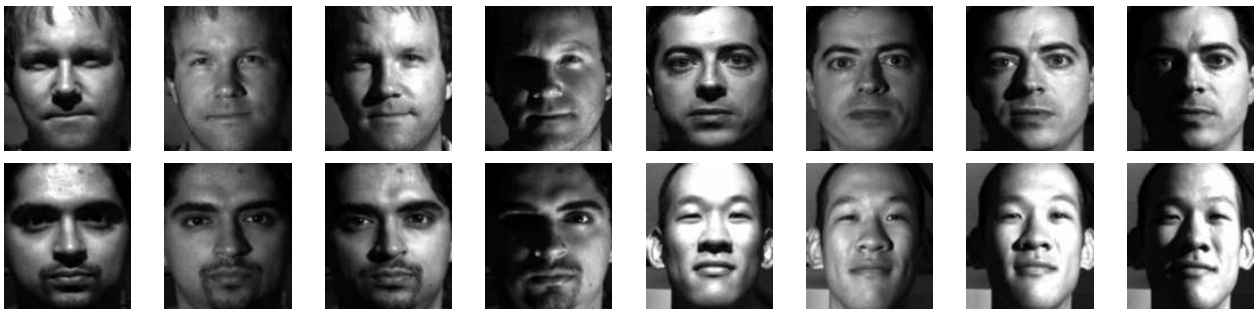

Figure 2: A subset of the YaleB database which contains 1920 frontal face images of thirty individuals taken under different lighting conditions.

Section 8.2 presents a more realistic surveillance application in which equivalence constraints are gathered automatically from a Markovian process. We conclude our experimental validation by comparing RCA with other methods which make use of equivalence constraints in a clustering task, using a few benchmark data sets from the UCI repository (Blake & Merz, 1998).

8.1 Applying RCA to facial recognition

The task here is to classify facial images with respect to the person photographed. In these experiments we consider a retrieval paradigm reminiscent of nearest neighbor classification, in which a query image leads to the retrieval of its nearest neighbor or its k-nearest neighbors in the dataset. Using a facial image database, we begin by evaluating nearest neighbor classification with the RCA distance, and compare its performance to supervised and unsupervised learning methods. We then move on to address specific issues regarding RCA: In Section 8.1.4 we test RCA with various dimensionality reduction procedures, and in Section 8.1.5 we evaluate the effect of different chunklets sizes. Finally, in Section 8.1.6 we show that RCA can also be successfully used to augment (standard) FLD.

8.1.1 The data set

We used a subset of the yaleB data set (Belhumeur et al., 1997), which contains facial images of 30 subjects under varying lighting conditions. The dataset contains a total of 1920 images, including 64 frontal pose images of each subject. The variability between images of the same person is mainly due to different lighting conditions. These factors caused the variability among images belonging to the same subject to be greater than the variability among images of different subjects (Adini et al., 1997). As preprocessing, we first automatically centered all the images using optical flow. Images were then converted to vectors, and each image was represented using its first d PCA coefficients (we experimented with $d = 30, 60, 100, 200$). Fig. 2 shows a few images of four subjects.

8.1.2 Obtaining equivalence constraints

We simulated the *'distributed learning'* scenario presented in Section 1 in order to obtain equivalence constraints. In this scenario, we obtain equivalence constraints using the help of T teachers. Each teacher is given a random selection of K data points from the data set, and is asked to give his own labels to all the points, effectively partitioning the dataset into equivalence classes. The constraints provided by the teachers are gathered and used as equivalence constraints. The total number of points in chunklets grows linearly

with TK, the number of data points seen by a teacher. This amount, which gives a loose bound on the number of points in chunklets, is controlled by varying the number of teachers T. We tested a range of values of T for which TK is 10%, 30%, or 75% of the points in the data set.[5]

In appendix 10 we show that the distribution of chunklet size is controlled by the ratio $r = \frac{K}{M}$ where M is the number of classes in the data. In all our experiments we have used $r = 2$. For this value the expected chunklet size is roughly 2.9 and we typically obtain many small chunklets. Fig. 3 shows a histogram of typical chunklet sizes, as obtained in our experiments.[6]

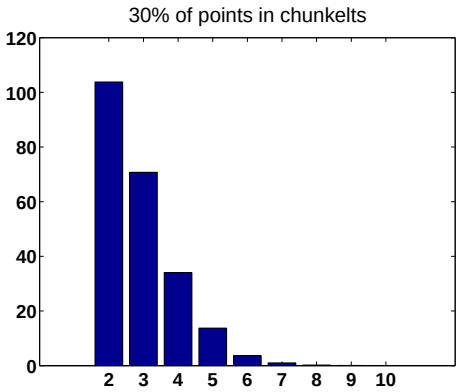

Figure 3: Sample chunklet size distribution obtained using the distributed learning scenario on a subset of the yaleB dataset with 1920 images from $M = 30$ classes, with $r = \frac{K}{M} = 2$. The histogram is plotted for 30% of the data in constraints (chunklets).

8.1.3 RCA on the continuum between supervised and unsupervised learning

The goal of our main experiment in this section was to assess the relative performance of RCA as a semi-supervised method in a face recognition task. To this extent we compared the following methods:

- Eigenfaces (Turk & Pentland, 1991): this unsupervised method reduces the dimensionality of the data using PCA, and compares the images using a Euclidean metric in the reduced space. Images were normalized to have zero mean and unit variance, and the number of dimensions was varied until optimal performance was obtained.
- Fisherfaces (Belhumeur et al., 1997): this supervised method starts by applying dimensionality reduction as in the Eigenfaces method. It then uses all the data labels to compute the FLD transformation (Fukunaga, 1990), and transforms the data accordingly.
- RCA: RCA with the dimensionality reduction procedure described in Section 6. We varied the amount of data in constraints provided to RCA, using the *distributed learning* paradigm described above.

Fig. 4 shows the results of the different methods. The left graph in Fig. 4 shows the relative performance of the three methods using nearest neighbor classification. As can be seen, RCA sometimes achieves error rates which are even lower than the error rates of the fully supervised Fisherface method, while relying only

[5] In this scenario one usually obtains mostly 'negative' equivalence constraints, which are pairs of points that are known to originate from different classes. Note that RCA does *not* use these 'negative' equivalence constraints.

[6] But we used a different sampling scheme in the experiments which address the effect of chunklet size (Section 8.1.5).

on fragmentary chunklets with unknown class labels. This somewhat surprising result stems from the fact that combining RCA with cFLD usually yields better performance than using FLD alone. (In Section 8.1.6 we discuss RCA as an extension to the fully labeled FLD.)

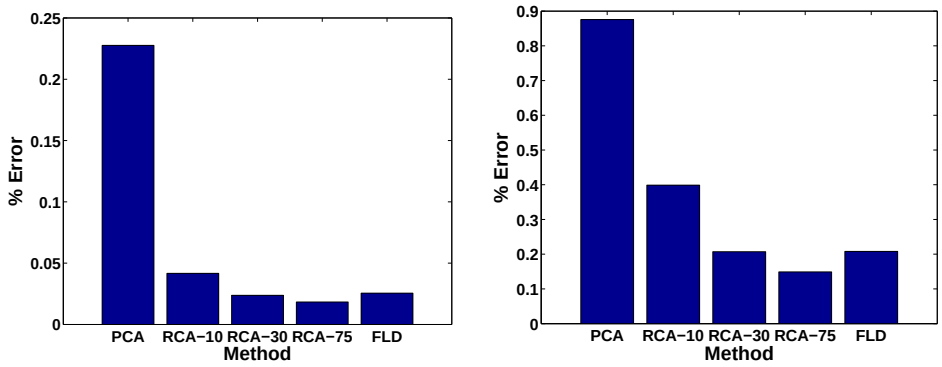

Figure 4: Classification Error rates for (1) Eigenface (PCA), (2) RCA 10%, (3) RCA 30%, (4) RCA 75% and (5) Fisherface (FLD). RCA was used in 3 conditions, with chunklets obtained from a distribution of a fraction of the total number of points, from small (10% and 30%) to large (75%). The left graph shows results of nearest neighbor classification. The right graph shows the mean error rate on all neighbors. Results are averaged over 50 chunklet realizations. Note that when using large amounts of equivalence constraints the performance of RCA is better than the fully labeled Fisherface, which is not followed by RCA.

Another interesting measure of performance is the total number of errors on all neighbors. For each image x_i, we compute the $|C_i| - 1$ nearest neighbors of that image where $|C_i|$ is the size of the class to which x_i belongs. We then compute the fraction of neighbors which were retrieved incorrectly. The right graph in Fig. 4 shows the mean error rate on all neighbors using the three methods. The pattern of results is similar to the one obtained for the first neighbor error. Even with small amounts of equivalence constraints (10% and 30%) RCA achieves error rates which are far better than the unsupervised PCA, and as good as the fully supervised FLD.

In order to visualize the effect of RCA in this task we also created some RCAfaces (following (Belhumeur et al., 1997)): We ran RCA on the images after applying PCA, and then reconstructed the images. Fig. 5 shows a few images and their reconstruction. Clearly RCA dramatically reduces the effect of varying lighting conditions, and the reconstructed images of the same individual look very similar to each other.

In another experiment we compared the performance of (Xing et al., 2002) to RCA on the YaleB dataset using code obtained from the author's website. The experimental setup was the one described in Section 8.1.2, with 30% of the data points distributed into chunklets. Results are shown in Fig. 6.

8.1.4 Dimensionality reduction with cFLD vs. PCA

In this experiment we compare PCA to cFLD as the mechanism to reduce dimensionality prior to RCA. Results over the YaleB data set are shown in Fig. 7, with *cumulative neighbor purity* plots of the two methods. *Cumulative neighbor purity* measures the percentage of correct neighbors up to the K-th neighbor, averaged over all the datapoints. Not surprisingly, we can see that even with small amounts of constraints, cFLD gives better performance.

8.1.5 The effect of different chunklet sizes

In Section 5 we showed that RCA typically provides an inner class covariance estimator which is not very sensitive to the chunklets' sizes. In order to empirically test the effect of chunklets' size, we fixed the number

Figure 5: Top: Several facial images of two subjects under different lighting conditions. Bottom: the same images from the top row after applying PCA and RCA and then reconstructing the images. Clearly RCA dramatically reduces the effect of different lighting conditions, and the reconstructed images of each person look very similar to each other.

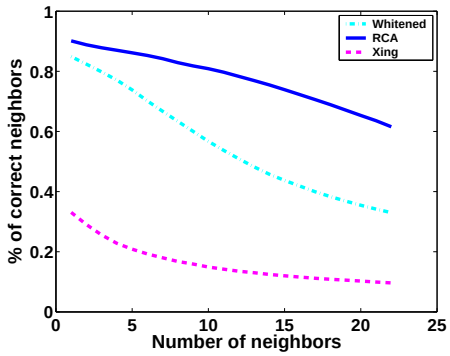

Figure 6: Comparison of Xing's method (Xing et al., 2002) with RCA on the YaleB facial image dataset. Results are presented using neighbor purity plots.

of equivalence constraints, and varied the size of the chunklets S in the range $\{2-10\}$. The chunklets were obtained by randomly selecting 30% of the data (total of $N = 1920$ points) and dividing it into chunklets of size S.[7]

The results can be seen in Fig. 8. As expected the performance of RCA improves as the size of the chunklets increases. However, most of the gain in performance is obtained with chunklets of size $S = 3$, in agreement with our theoretical analysis.

8.1.6 RCA in a supervised learning scenario

RCA can also be used when given a fully labeled training set. In this case, chunklets correspond uniquely and fully to classes, and the cFLD algorithm for dimensionality reduction is equivalent to the standard FLD. In this setting RCA can be viewed as an augmentation of standard FLD. Fig. 9 shows comparative results of FLD with and without RCA in the fully labeled case.

[7] When necessary, the remaining $mod(N, S)$ points were gathered into an additional smaller chunklet.

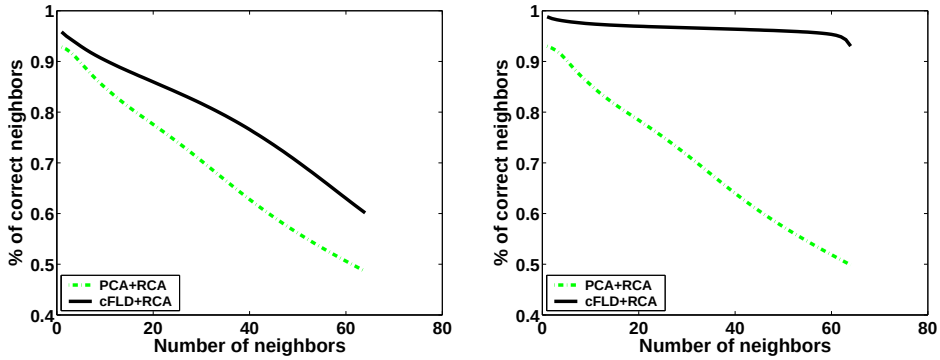

Figure 7: Neighbor purity plots of RCA after dimensionality reduction using PCA and cFLD. When using cFLD each image was first represented using its first 100 and 200 PCA coefficients respectively. We then used cFLD to reduce the dimensionality to $M = 30$, the number of classes in the data (as described in Section 6). When using PCA the dimensionality was reduced to 30 PCA coefficients directly. Left: 10% of the data in constraints. Right: 75% of the data in constraints. As can be seen, even with a small number of equivalence constraints, the combination of cFLD+RCA performs better than the combination PCA+RCA.

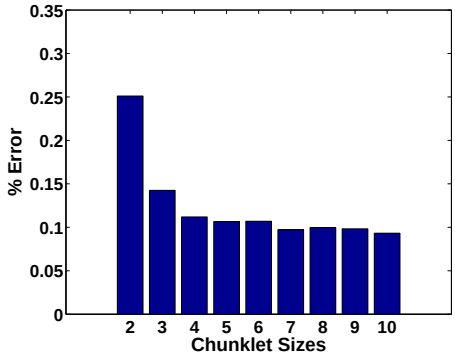

Figure 8: Mean error on all neighbors on the yaleB dataset when using 30% of the data in chunklets. In this experiment we varied the chunklet sizes while fixing the total amount of points in chunklets. As expected, as the size of the chunklet increases performance improves. However, note that most of the performance gain is obtained using chunklets of size 3.

8.2 Surveillance application

In this application, a stationary indoor surveillance camera provided short video clips whose beginning and end were automatically determined by the appearance and disappearance of a moving target. The database therefore included many clips, each displaying only one person of unknown identity. Effectively each clip provided a chunklet. The task in this case was to cluster together all clips in which a certain person appeared.

The task and our approach: The video clips were highly complex and diversified, for several reasons. First, they were entirely unconstrained: a person could walk everywhere in the scene, coming closer to the camera or walking away from it. Therefore the size and resolution of each image varied dramatically. In addition, since the environment was not constrained, images included varying occlusions, reflections and (most importantly from our perspective) highly variable illumination. In fact, the illumination changed

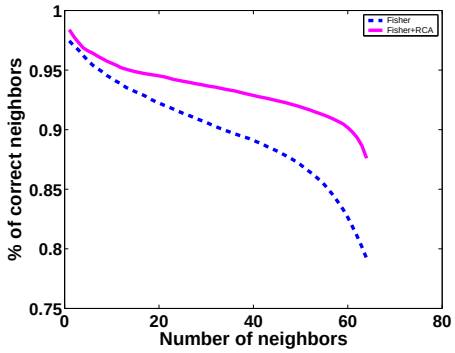

Figure 9: Neighbor purity plots on the yaleB dataset using FLD with and without RCA. Here RCA dramatically enhances the performance obtained by FLD.

!

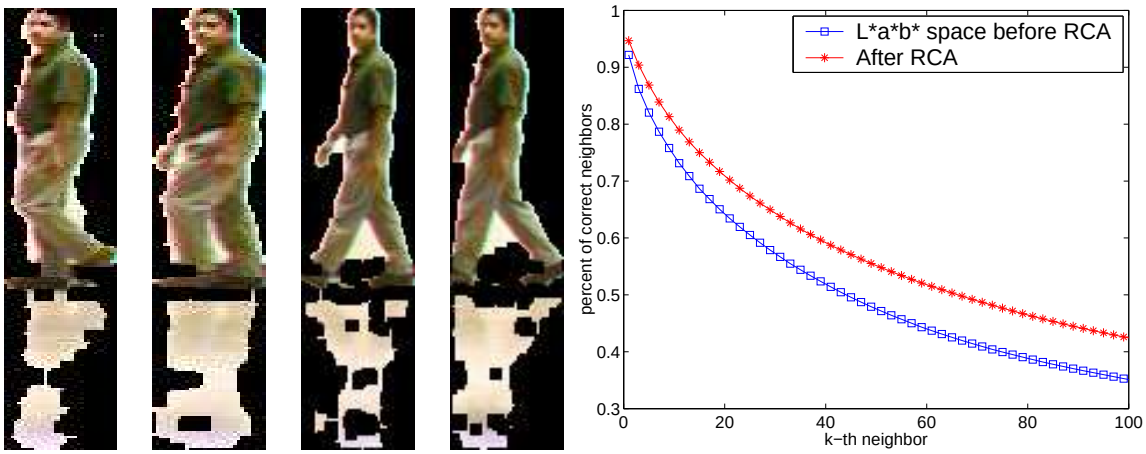

Figure 10: Left: several images from a video clip of one intruder. Right: percent of correct retrievals as a function of the number of retrieved images.

dramatically across the scene both in intensity (from brighter to darker regions), and in spectrum (from neon light to natural lighting). Fig. 10 shows several images from one input clip.

We sought to devise a representation that would enable the effective clustering of clips, focusing on color as the only low-level attribute that could be reliably used in this application. Therefore our task was to accomplish some sort of color constancy, i.e., to overcome the irrelevant variability due to the varying illumination.

Image representation and RCA Each image in a clip was represented by its color histogram in $L^*a^*b^*$ space (we used 5 bins for each dimension). We used the clips as chunklets in order to compute the RCA transformation. We then computed the distance between pairs of images using two methods: *L1* and RCA (Mahalanobis), and sorted the neighbors of each image according to both distances. We used over 6000 images from 130 clips (chunklets) of 20 different people. Fig. 10 shows the percentage of 'correct' neighbors up to the k'th neighbor over all 6000 images. One can see that RCA makes a significant contribution by bringing 'correct' neighbors closer to each other (relative to other images).

8.3 RCA and clustering

In this section we evaluate RCA's contribution to clustering, and compare it to alternative algorithms that use equivalence constraints. We used six data sets from the UCI repository. For each data set we randomly selected a set S of pairwise similarity constraints (or chunklets of size 2). We compared the following clustering algorithms:

1. K-means using the default Euclidean metric (using no side-information) (Fukunaga, 1990).
2. Constrained K-means + Euclidean metric: A K-means version suggested by (Wagstaff et al., 2001), in which a pair of points $(x_i, x_j) \in S$ is always assigned to the same cluster.
3. Constrained K-means + the metric proposed in (Xing et al., 2002): Constrained K-means using the Mahalanobis metric proposed in (Xing et al., 2002), which is learned from the constraints in S.
4. Constrained K-means + RCA: Constrained K-means using the RCA Mahalanobis metric learned from S.
5. EM: Expectation Maximization of a Gaussian Mixture model (using no side-information).
6. Constrained EM: EM using side-information in the form of equivalence constraints (Shental et al., 2003), when using the RCA distance metric as the initial metric.

The clustering algorithms numbered 4 and 6 are our own. Clustering algorithms 1 and 5 are unsupervised and provide respective lower bounds for comparison with our algorithms. Clustering algorithms 2 and 3 compete fairly with our algorithm 4, using the same kind of side information.

Experimental setup To ensure fair comparison with (Xing et al., 2002), we used exactly the same experimental setup as it affects the gathering of equivalence constraints and the evaluation score used. We tested all methods using two conditions, with: (i) "little" side-information S, and (ii) "much" side-information. The set of pairwise similarity constraints S was generated by choosing a random subset of all pairs of points sharing the same class identity c_i. In the case of little side-information, the size of the subset was chosen so that the total number of different connected components K_c (using transitive closure over pairs) was roughly 90% of the size of the original dataset. In case of much side information this was reduced to 70% (where fewer connected components imply that the components are larger, which implies in turn more information).

Following (Xing et al., 2002) we used a normalized accuracy score (the "Rand index") to evaluate the partitions obtained by the different clustering algorithms. More formally, with binary labels (or two clusters), the accuracy measure can be written as:

$$\sum_{i>j} \frac{1\{1\{c_i = c_j\} = 1\{\hat{c}_i = \hat{c}_j\}\}}{0.5m(m-1)}$$

where $1\{\dot{}\}$ denotes the indicator function $(1\{True\} = 1), 1\{False\} = 0)$, $\{\hat{c}_i\}_{i=1}^m$ denotes the cluster to which point x_i is assigned by the clustering algorithm, and c_i denotes the "correct" (or desirable) assignment. The score above is equivalent to computing the probability that the algorithm's assignment $\hat{c}$ of two randomly drawn points x_i and x_j agrees with the "true" assignment c.[8]

[8] As noted in (Xing et al., 2002), this score should be normalized when the number of clusters is larger than 2. Normalization is achieved by sampling the pairs x_i and x_j from the same cluster (as determined by $\hat{c}$) with probability 0.5 and from different clusters with probability 0.5, so that "matches" and "mismatches" are given the same weight.

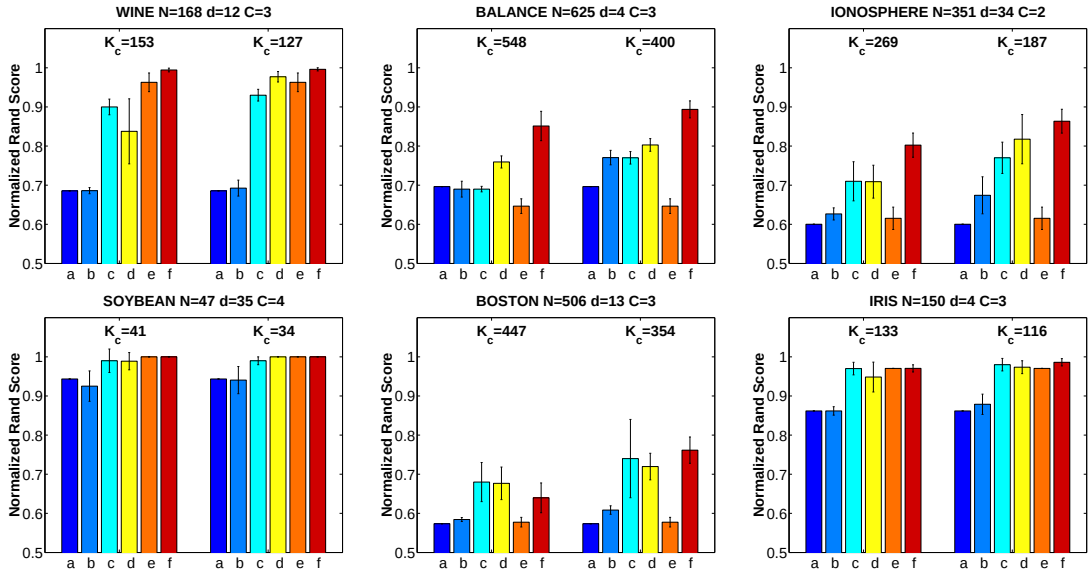

Figure 11: Clustering accuracy on 6 UCI datasets. In each panel, the six bars on the left correspond to an experiment with "little" side-information, and the six bars on the right correspond to "much" side-information. From left to right the six bars correspond respectively to the algorithms described in the text, as follows: (a) K-means over the original feature space (without using any side-information). (b) Constrained K-means over the original feature space. (c) Constrained K-means over the feature space suggested by (Xing et al., 2002). (d) Constrained K-means over the feature space created by RCA. (e) EM over the original feature space (without using any side-information). (f) Constrained EM (Shental et al., 2003) over the feature space created by RCA. Also shown are N - the number of points, C - the number of classes, d - the dimensionality of the feature space, and K_c - the mean number of connected components. The results were averaged over 20 realizations of side-information. In all experiments we used K-means with multiple restarts as in (Xing et al., 2002).

Fig. 11 shows comparative results using six different UCI data sets. Clearly the RCA metric significantly improved the results over the original K-means algorithms (both the constrained and unconstrained version). Generally in this context, we see that using equivalence constraints to find a better metric improves results much more than using this information to constrain the algorithm. RCA achieves better or similar results to those reported in (Xing et al., 2002). However, while the RCA metric involves a single step efficient computation, the method presented in (Xing et al., 2002) requires gradient descent and iterative projections and is also sensitive to the initial conditions used.

The comparisons in Fig. 11 involve six different clustering algorithms. The last two algorithms use the EM algorithm to compute a generative Gaussian Mixture Model, and are therefore much more complex computationally. We have added these comparisons because EM implicitly changes the distance function over the input space in a linear way (i.e., like a Mahalanobis distance). It therefore may appear that EM can do everything that RCA does and more, without any modification. The histogram bins marked by (e) in Fig. 11 clearly show that this is not the case. Only when we add constraints to the EM, and preprocess the data with RCA, do we get improved results as shown by the histogram bins marked by (f) in Fig. 11.

9 Discussion

In order to obtain strict probabilistic justification for RCA, we listed in Section 5 the following assumptions:

- The classes have multi normal distributions.
- All the classes share the same covariance matrix.
- The chunklets points are an i.i.d sample from the class.

What happens when these assumptions do not hold?

The first assumption gives RCA its probabilistic justification, but even without it (i.e., a distribution-free model), RCA is justified as the optimal transformation by two criteria: mutual information maximization, and inner chunklet distance minimization. These criteria are reasonable as long as the classes are approximately convex (as implied by the use of the distance between chunklet's points and chunklet's means).

The second assumption justifies RCA's main computational step, which uses the empirical average of all the chunklets covariance matrices in order to estimate the global inner class covariance matrix. When this assumption fails, RCA effectively extracts the shared component of all the classes covariance matrices, if such component exists.

The third assumption may break down in many practical applications, when chunklets are automatically collected and the points within a chunklet are no longer independent of one another. As a result chunklets may be composed of points which are rather close to each other, and whose distribution does not reflect all the typical variance of the true distribution. In this case RCA's performance is not guaranteed to be optimal.

10 Conclusion

We have presented an algorithm which uses side-information in the form of equivalence constraints, in order to learn a Mahalanobis metric. We have shown that our method is optimal under several criteria. Our empirical results show that RCA reduces irrelevant variability in the data and thus leads to considerable clustering improvements. RCA matlab code can be downloaded from the authors' site.

Appendix A: Information Maximization in the case of non invertible linear transformation

Here we sketch the proof of the claim made in Section 3.3. As before, we denote by C the average covariance matrix of the chunklets. We can rewrite the constrained expression as:

$$\frac{1}{p}\sum_{j=1}^{k}\sum_{i=1}^{n_j}(x_{ji}-m_j)^t A^t A(x_{ji}-m_j) = tr(A^t AC) = tr(A^t CA)$$

Hence the Lagrangian can be written as:

$$\log|A\Sigma_x A^t| - \lambda(tr(ACA^t) - K)$$

Differentiating the Lagrangian w.r.t A gives

$$\Sigma_x A(A^t\Sigma_x A)^{-1} = \lambda CA \tag{17}$$

Multiplying by A^t and rearranging terms, we get: $\frac{I}{\lambda} = A^t C A$. Hence as in RCA, A must whiten the data with respect to the chunklet covariance C in a yet to be determined subspace. Using $\lambda \neq 0$ it then follows that the inequality constraint in (6) is an equality, which can be used to find λ.

$$\begin{aligned} tr(ACA^t) = tr(\frac{I}{\lambda}) = \frac{M}{\lambda} = K &\implies \lambda = \frac{M}{K} \\ &\implies ACA^t = \frac{K}{M} I \end{aligned}$$

where M is the dimensionality of the projection subspace.

Next, since in our solution space $ACA^t = \frac{K}{M} I$, it follows that $\log|ACA^t| = M \log \frac{K}{M}$ holds for all points. Hence we can modify the maximization argument as follows

$$\log|A\Sigma_x A^t| \quad = \quad \log \frac{|A\Sigma_x A^t|}{|ACA^t|} + M \log \frac{K}{M}$$

Now the optimization argument has a familiar form. It is known (Fukunaga, 1990) that maximizing the determinant ratio can be done by projecting the space on the span of the first M eigenvectors of $C^{-1}\Sigma_x$. Denote by B the solution matrix for this unconstrained problem. In order to enforce the constraints we define the matrix $A = \sqrt{\frac{K}{M}} \Lambda_1^{-0.5} B$ and claim that A is the solution of the constrained problem. Notice that the value of the maximization argument does not change when we switch from A to B since A is a product of B and another full ranked matrix. It can also be shown that A satisfies the constraints and is thus the solution of the problem presented in Eq. (6).

Appendix B:Variance bound on the RCA covariance estimator

In this appendix we prove the inequality 15 from section 5. Assume we have $p = nk$ data points $\mathbf{X} = \{x_i^j\}_{i=1,j=1}^{n,k}$ in n chunklets of size k each. We assume that all chunklets are drawn independently from Gaussian sources with the same covariance matrix. Denoting by $\hat{m}_i$ the empirical mean of chunklet i, the unbiased RCA estimator of this covariance matrix is

$$\hat{C}(n,k) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{k-1} \sum_{j=1}^{k} (x_i^j - \hat{m}_i)(x_i^j - \hat{m}_i)^T$$

Let U denote the diagonalization transformation of the covariance matrix C of the Gaussian sources, i.e., $U^t C U = \Lambda$ where Λ is a diagonal matrix with $\{\lambda_i\}_{i=1}^n$ on the diagonal. Let $\mathbf{Y} = \mathbf{X}U$ denote the transformed data. Denote its covariance matrix by $\hat{C}^u(n,k) = U^t \hat{C}(n,k) U$, and denote the empirical chunklet means by $\hat{m}_i^u = U\hat{m}_i$. We can analyze the variance of $\hat{C}^u$ as follows:

$$\begin{aligned} var(\hat{C}^u(n,k)) &= var[\frac{1}{n} \sum_{i=1}^{n} \frac{1}{k-1} \sum_{j=1}^{k} (y_i^j - \hat{m}_i^u)(y_i^j - \hat{m}_i^u)^T] \\ &= \frac{1}{n} var[\frac{1}{k-1} \sum_{j=1}^{k} (y_1^j - \hat{m}_1^u)(y_1^j - \hat{m}_1^u)^T] \end{aligned}$$

The last equality holds since the summands of the external sum are sample covariances of independent chunklets drawn from sources with the same covariance matrix.

The variance of the sample covariance for diagonalized Gaussian data is known to be (Fukunaga, 1990)

$$var(\hat{C}_{ii}) = \frac{2\lambda_i^2}{k-1}; var(\hat{C}_{ij}) = \frac{\lambda_i\lambda_j}{k}; cov(\hat{C}_{ij}, \hat{C}_{kl}) = 0$$

and we get the following variance for the RCA estimator:

$$var(\hat{C}^u_{ii}) = \frac{2\lambda_i^2}{n(k-1)}; var(\hat{C}^u_{ij}) = \frac{\lambda_i\lambda_j}{nk}; cov(\hat{C}^u_{ij}, \hat{C}^u_{kl}) = 0$$

As $p = nk$, we can write

$$var(\hat{C}^u_{ii}) = \frac{2\lambda_i^2}{p(1-\frac{1}{k})}; var(\hat{C}^u_{ij}) = \frac{\lambda_i\lambda_j}{p}; cov(\hat{C}^u_{ij}, \hat{C}^u_{kl}) = 0$$

and for the diagonal terms $\hat{C}^u_{ii}$

$$var(\hat{C}^u(\frac{p}{k}, k)_{ii}) = \frac{2\lambda_i^2}{p(1-\frac{1}{k})} = \frac{k}{k-1}\frac{2\lambda_i^2}{p} \leq \frac{k}{k-1}\frac{2\lambda_i^2}{p-1} = var(\hat{C}^u(1,p)_{ii})$$

This inequality trivially holds for the off-diagonal covariance elements.

Getting back to the original data covariance, we note that in matrix elements notation $\hat{C}_{ij} = \sum_{q,r=1}^{d} \hat{C}^u_{qr} U_{iq} U_{jr}$ while d is the data dimensionality. Therefore

$$\frac{var[\hat{C}_{ij}(n,k)]}{var[\hat{C}_{ij}(1,nk)]} = \frac{\sum_{q,r=1}^{d} var[\hat{C}^u(n,k)_{qr} U_{iq} U_{jr}]}{\sum_{q,r=1}^{d} var[\hat{C}^u(1,nk)_{qr} U_{iq} U_{jr}]} \leq \frac{\sum_{q,r=1}^{d} \frac{k}{k-1} var[\hat{C}^u(1,nk)_{qr} U_{iq} U_{jr}]}{\sum_{q,r=1}^{d} var[\hat{C}^u(1,nk)_{qr} U_{iq} U_{jr}]} = \frac{k}{k-1}$$

where the first equality holds because $cov(\hat{C}^u_{ij}, \hat{C}^u_{kl}) = 0$.

Appendix C:The expected chunklet size in the distributed learning paradigm

Here we estimate the expected chunklet size obtained when using the distributed learning paradigm introduced in Section 8. In this scenario, we use the help of T teachers, each of which is provided with a random selection of K data points. Let us assume that the data contains M equiprobable classes, and that the size of the data set is large relative to K. We can define random variables x_i^j as the number of points from class i seen by teacher j. Due to the symmetry between classes and teachers, the distribution of x_i^j is the same for each i, j. It can be well approximated by a Bernoulli variable $x\ B(K, \frac{1}{M}$. This variable may take values of 0 or 1, which are not allowed when we wish to describe the distribution of chunklet size. The distribution of chunklets with sizes bigger than 1 is given by the trimmed Bernoulli distribution

$$p(x = i | x \neq 0, 1) = \frac{1}{1 - p(x=0) - p(x=1)} \left(\begin{array}{c} q \\ i \end{array} \right) (\frac{1}{m})^i (1 - \frac{1}{m})^{q-i} \quad i = 2, 3, ..$$

We can approximate $p(X = 0)$ and $p(X = 1)$ as

$$p(X = 0) \approx e^{-\frac{K}{m}} \quad p(X = 1) \approx \frac{K}{M} e^{-\frac{K}{m}}$$

Using these approximations, the expected chunklet size is determined by the ratio $r = \frac{K}{M}$ through the formula

$$E(x|x \neq 0, x \neq 1) \simeq \frac{r(1-e^{-r})}{1-(r+1)e^{-r}}$$

References

Adini, Y., Moses, Y., & Ullman, S. (1997). Face recognition: The problem of compensating for changes in illumination direction. *proc. of IEEE PAMI* (pp. 721–732).

Bar-Hilel, A., Hertz, T., Shental, N., & Weinshall, D. (2003). Learning distance functions using equivalence relations. *The 20th International Conference on Machine Learning.*.

Belhumeur, P., Hespanha, J., & Kriegman, D. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE PAMI 8*, *19(7)*, 711–720.

Bell, A., & Sejnowski, T. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation, 7*, 1129–1159.

Blake, C., & Merz, C. (1998). UCI repository of machine learning databases.

Boreczky, J. S., & Rowe, L. A. (1996). Comparison of video shot boundary detection techniques. *SPIE Storage and Retrieval for Still Images and Video Databases IV*, *2664*, 170–179.

Chechik, G., & Tishby, N. (2002). Extracting relevant structures with side information. *NIPS*, *15*.

Fukunaga, K. (1990). *Statistical pattern recognition*. San Diego: Academic Press. 2nd edition.

Hastie, T., & Tibshirani, R. (1996). Discriminant adaptive nearest neighbor classification and regression. *Advances in Neural Information Processing Systems* (pp. 409–415). The MIT Press.

Jaakkola, T., & Haussler, D. (1998). Exploiting generative models in discriminative classifiers.

Linsker, R. (1989). An application of the principle of maximum information preservation to linear systems. *NIPS* (pp. 186–194). Morgan Kaufmann.

Shental, N., Hertz, T., Bar-Hilel, A., & Weinshall, D. (2003). Computing gaussian mixture models with EM using equivalence constraints.

Shental, N., Hertz, T., Weinshall, D., & Pavel, M. (2002). Adjustment learning and relevant component analysis. *Computer Vision - ECCV*.

Tenenbaum, J., & Freeman, W. (2000). Separating style and content with bilinear models. *Neural Computation, 12*, 1247–1283.

Thrun, S. (1996). Is learning the n-th thing any easier than learning the first? *Advances in Neural Information Processing Systems* (pp. 640–646). The MIT Press.

Tishby, N., Pereira, F., & Bialek, W. (1999). The information bottleneck method. *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing* (pp. 368–377).

Turk, M., & Pentland, A. (1991). Face recognition using eigenfaces. *Proc. of IEEE CVPR* (pp. 586–591).

Wagstaff, K., Cardie, C., Rogers, S., & Schroedl, S. (2001). Constrained K-means clustering with background knowledge. *Proc. 18th International Conf. on Machine Learning* (pp. 577–584). Morgan Kaufmann, San Francisco, CA.

Xing, E., Ng, A., Jordan, M., & Russell, S. (2002). Distance metric learnign with application to clustering with side-information. *Advances in Neural Information Processing Systems*. The MIT Press.