Journal of Machine Learning Research 6 (2005) 937–965 Submitted 7/04; Revised 4/05; Published 6/05

Learning a Mahalanobis Metric from Equivalence Constraints

Aharon Bar-Hillel AHARONBH@CS.HUJI.AC.IL
Tomer Hertz TOMBOY@CS.HUJI.AC.IL
Noam Shental FENOAM@CS.HUJI.AC.IL
Daphna Weinshall DAPHNA@CS.HUJI.AC.IL
School of Computer Science & Engineering and Center for Neural Computation
The Hebrew University of Jerusalem
Jerusalem, Israel 91904

Editor: Greg Ridgeway

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 Mahalanobis metric. We show that RCA is the solution of an interesting optimization problem, founded on an information theoretic basis. If dimensionality reduction is allowed within RCA, we show that it is optimally accomplished by a version of Fisher's linear discriminant that uses constraints. Moreover, under certain Gaussian assumptions, RCA can be viewed as a Maximum Likelihood estimation of the within class covariance matrix. We conclude with extensive empirical evaluations of RCA, showing its advantage over alternative methods.

Keywords: clustering, metric learning, dimensionality reduction, equivalence constraints, side information.

1. Introduction

A number of learning problems, such as clustering and nearest neighbor classification, rely on some a priori defined distance function over the input space. 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 faces 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 modelled 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 robustly detected (Boreczky and 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] For example, assume that you are given a large database of facial images of many people, which cannot be labelled by a small number of teachers 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 labelling the original data set. 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. Although some of the interesting results can only be proven using explicit Gaussian assumptions, the optimality of RCA can be shown with some relatively weak assumptions, restricting the discussion to linear transformations and the Euclidean norm. Specifically, in Section 3 we describe a novel information theoretic criterion and show that RCA is its optimal solution. If Gaussian assumptions are added the result can be extended to the case where dimensionality reduction is permitted, and the optimal solution now includes Fisher's linear discriminant (Fukunaga, 1990) as an intermediate step. In Section 4 we show that RCA is also the optimal solution to another optimization problem, seeking to minimize within class distances. Viewed this way, RCA is directly compared to another recent algorithm for learning Mahalanobis distance from equivalence constraints, proposed by 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 the fully labelled data.

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.

The successful application of RCA in high dimensional spaces requires dimensionality reduction, whose details are discussed in Section 6. An online version of the RCA algorithm is presented in Section 7. In Section 8 we describe extensive empirical evaluations of the RCA algorithm. We focus on two tasks–data retrieval and clustering, and use 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 labelled 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.

1.1 Related Work

There has been much work on learning representations and distance functions in the supervised learning settings, and we can only briefly mention a few examples. Hastie and Tibshirani (1996) and Jaakkola and Haussler (1998) use labelled data to learn good metrics for classification. Thrun (1996) learns a distance function (or a representation function) for classification using a "leaning-to-learn" paradigm. In this setting several related classification tasks are learned using several labelled 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 the work of Tishby et al. (1999) the joint distribution of two random variables X and Z is assumed to be known, and one seeks a compact representation of X which bears high relevance to Z. This work, which is further developed in Chechik and Tishby (2003), can be viewed as supervised representation learning.

As mentioned, RCA can be justified using information theoretic criteria on the one hand, and as an ML estimator under Gaussian assumptions on the other. Information theoretic criteria for unsupervised learning in neural networks were studied by Linsker (1989), and have been used since in several tasks in the neural network literature. Important examples are self organizing neural networks (Becker and Hinton, 1992) and Independent Component Analysis (Bell and Sejnowski, 1995)). Viewed as a Gaussian technique, RCA is related to a large family of feature extraction techniques that rely on second order statistics. This family includes, among others, the techniques of Partial Least-Squares (PLS) (Geladi and Kowalski, 1986), Canonical Correlation Analysis (CCA) (Thompson, 1984) and Fisher's Linear Discriminant (FLD) (Fukunaga, 1990). All these techniques extract linear projections of a random variable X, which are relevant to the prediction of another variable Z in various settings. However, PLS and CCA are designed for regression tasks, in which Z is a continuous variable, while FLD is used for classification tasks in which Z is discrete. Thus, RCA is more closely related to FLD, as theoretically established in Section 3.3. An empirical investigation is offered in Section 8.1.3, in which we show that RCA can be used to enhance the performance of FLD in the fully supervised scenario.

In recent years some work has been done on using equivalence constraints as side information. Both positive ('a is similar to b') and negative ('a is dissimilar from b') equivalence constraints were considered. Several authors considered the problem of semi-supervised clustering using equivalence constraints. More specifically, positive and negative constraints were introduced into the complete linkage algorithm (Klein et al., 2002), the K-means algorithm (Wagstaff et al., 2001) and the EM of a Gaussian mixture model (Shental et al., 2004). A second line of research, to which this work belongs, focuses on learning a 'good' metric using equivalence constraints. Learning a Mahalanobis metric from both positive and negative constraints was addressed in the work of Xing et al. (2002),

presenting an algorithm which uses gradient ascent and iterative projections to solve a convex non linear optimization problem. We compare this optimization problem to the one solved by RCA in Section 4, and empirically compare the performance of the two algorithms in Section 8. The initial description of RCA was given in the context of image retrieval (Shental et al., 2002), followed by the work of Bar-Hillel et al. (2003). Recently Bilenko et al. (2004) suggested a K-means based clustering algorithm that also combines metric learning. The algorithm uses both positive and negative constraints and learns a single or multiple Mahalanobis metrics.

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" (see Tenenbaum and Freeman, 2000). These "relevant dimensions" are estimated using *chunklets*, that is, small subsets of points that are known to belong to the same although *unknown* class. The algorithm is presented below as Algorithm 1 (Matlab code can be downloaded from the authors' sites).

Algorithm 1 The RCA algorithm

Given a data set $X = \{x_i\}_{i=1}^{N}$ and n chunklets $C_j = \{x_{ji}\}_{i=1}^{n_j}$ $j = 1 \dots n$, do

1. Compute the within chunklet covariance matrix (Figure 1d)

$$\hat{C} = \frac{1}{N} \sum_{j=1}^{n} \sum_{i=1}^{n_j} (x_{ji} - m_j)(x_{ji} - m_j)^t, \tag{1}$$

   where m_j denotes the mean of the j'th chunklet.

2. If needed, apply dimensionality reduction to the data using $\hat{C}$ as described in Algorithm 2 (see Section 6).

3. Compute the whitening transformation associated with $\hat{C}$: $W = \hat{C}^{-\frac{1}{2}}$ (Figure 1e), and apply it to the data points: $X_{new} = WX$ (Figure 1f), where X refers to the data points after dimensionality reduction when applicable. Alternatively, use the inverse of $\hat{C}$ in the Mahalanobis distance: $d(x_1, x_2) = (x_1 - x_2)^t \hat{C}^{-1} (x_1 - x_2)$.

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 RCA transformation is intended to reduce clutter, so that in the new feature space, the inherent structure of the data can be more easily unravelled (see illustrations in Figure 1a-f). To this end, the algorithm estimates the within class covariance of the data $cov(X|Z)$ where X and Z describe the data points and their labels respectively. The estimation is based on positive equivalence constraints

only, and does not use any explicit label information. In high dimensional data, the estimated matrix can be used for semi-supervised dimensionality reduction. Afterwards, the data set is whitened with respect to the estimated within class covariance matrix. The whitening transformation W (in Step 3 of Algorithm 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.

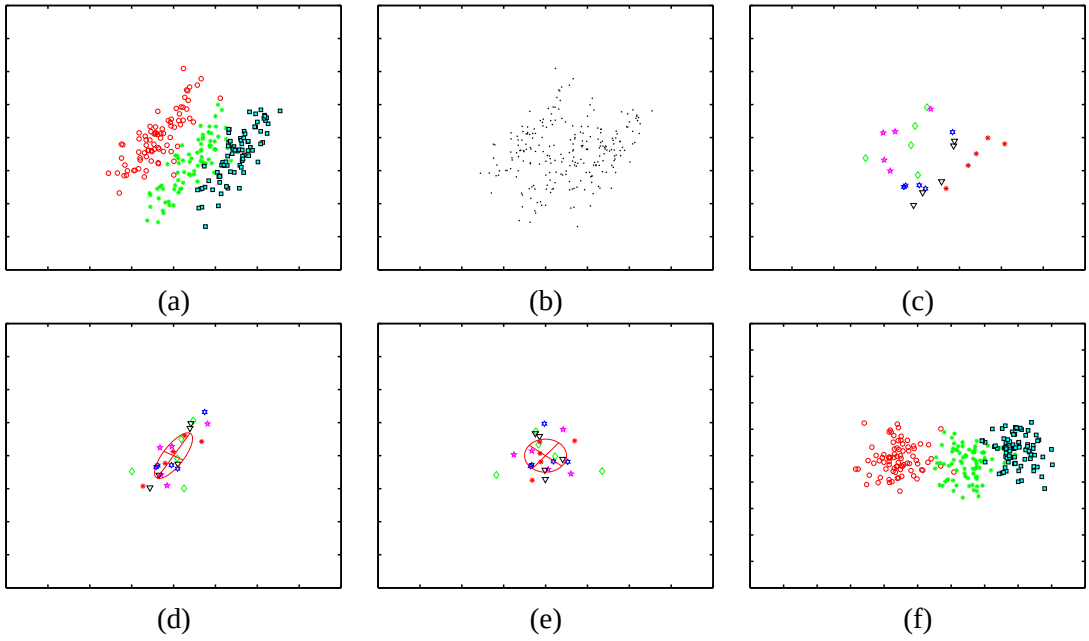

Figure 1: An illustrative example of the RCA algorithm applied to synthetic Gaussian data. (a) The fully labelled data set with 3 classes. (b) Same data unlabelled; clearly the classes' structure is less evident. (c) The set of chunklets that are provided to the RCA algorithm (points that share the same 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 theoretical justifications for the RCA algorithm are given in Sections 3-5. In the following discussion, the term 'RCA' refers to the algorithm either with or without dimensionality reduction (optional Step 2). Usually the exact meaning can be readily understood in context. When we specifically discuss issues regarding the use of dimensionality reduction, we may use the explicit terms 'RCA with (or without) dimensionality reduction'.

RCA does not use negative equivalence constraints. While negative constraints clearly contain useful information, they are less informative than positive constraints (see counting argument below). They are also much harder to use computationally, due partly to the fact that unlike positive constraints, negative constraints are not transitive. In our case, the naïve incorporation of negative constraints leads to a matrix solution which is the difference of two positive definite matrices, and as a results does not necessarily produce a legitimate Mahalanobis metric. An alternative approach, which modifies the optimization function to incorporate negative constraints, as used for example by Xing et al. (2002), leads to a non-linear optimization problem with the usual associated drawbacks

of increased computational load and some uncertainty about the optimality of the final solution.[2] In contrast, RCA is the closed form solution of several interesting optimization problem, whose computation is no more complex than a single matrix inversion. Thus, in the tradeoff between runtime efficiency and asymptotic performance, RCA chooses the former and ignores the information given by negative equivalence constraints.

There is some evidence supporting the view that positive constraints are more informative than negative constraints. Firstly, a simple counting argument shows that positive constraints exclude more labelling possibilities than negative constraints. If for example there are M classes in the data, two data points have M^2 possible label combinations. A positive constraint between the points reduces this number to M combinations, while a negative constraint gives a much more moderate reduction to $M(M-1)$ combinations. (This argument can be made formal in information theoretic terms.) Secondly, empirical evidence from clustering algorithms which use both types of constraints shows that in most cases positive constraints give a much higher performance gain (Shental et al., 2004; Wagstaff et al., 2001). Finally, in most cases in which equivalence constraints are gathered automatically, only positive constraints can be gathered.

Step 2 of the RCA algorithm applies dimensionality reduction to the data if needed. In high dimensional spaces dimensionality reduction is almost always essential for the success of the algorithm, because the whitening transformation essentially re-scales the variability in all directions so as to equalize them. Consequently, dimensions with small total variability cause instability and, in the zero limit, singularity.

As discussed in Section 6, the optimal dimensionality reduction often starts with Principal Component Analysis (PCA). PCA may appear contradictory to RCA, since it eliminates principal dimensions with small variability, while RCA emphasizes principal dimensions with small variability. One should note, however, that the principal dimensions are computed in different spaces. The dimensions eliminated by PCA have small variability in the original data space (corresponding to $Cov(X)$), while the dimensions emphasized by RCA have low variability in a space where each point is translated according to the centroid of its own chunklet (corresponding to $Cov(X|Z)$). As a result, the method ideally emphasizes those dimensions with large total variance, but small within class variance.

3. Information Maximization with Chunklet Constraints

How can we use chunklets to find a transformation of the data which improves its representation? In Section 3.1 we state the problem for general families of transformations and distances, presenting an information theoretic formulation. In Section 3.2 we restrict the family of transformation to non-singular linear maps, and use the Euclidean metric to measure distances. The optimal solution is then given by RCA. In Section 3.3 we widen the family of permitted transformations to include non-invertible linear transformations. We show that for normally distributed data RCA is the optimal transformation when its dimensionality reduction is obtained with a constraints based Fisher's Linear Discriminant (FLD).

2. Despite the problem's convexity, the proposed gradient based algorithm needs tuning of several parameters, and is not guaranteed to find the optimum without such tuning. See Section 8.1.5 for relevant empirical results.

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 case a set $X = \{x_i\}$ of data points in $\mathcal{R}^D$ is transformed into the set $Y = \{f(x_i)\}$ of points in $\mathcal{R}^K$. We seek a deterministic function $f \in F$ that maximizes $I(X,Y)$, where F is the family of permitted transformation functions (a "hypotheses family").

First, note that since f is deterministic, maximizing $I(X,Y)$ is achieved by maximizing the entropy $H(Y)$ alone. To see this, recall that by definition

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

where $H(Y)$ and $H(Y|X)$ are differential entropies, as X and Y are continuous random variables. Since f is deterministic, the uncertainty concerning Y when X is known is minimal, thus $H(Y|X)$ achieves its lowest possible value at $-\infty$.[3] However, as noted by Bell and Sejnowski (1995), $H(Y|X)$ does not depend on f and is constant for every finite quantization scale. Hence maximizing $I(X,Y)$ 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, if $Y = f(X)$ for an invertible continuous function, we can increase $H(Y)$ simply by choosing $Y = \lambda f(X)$ for any $\lambda > 1$. In order to avoid the trivial solution $\lambda \to \infty$, we can limit the distances between points contained in a single chunklet . This can be done by constraining the average distance between a point in a chunklet and the chunklet's mean. Hence the optimization problem is

$$\max_{f \in F} H(Y_f) \quad s.t. \ \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j} ||y_{ji} - m_j^y|| \leq \kappa \tag{2}$$

where $\{y_{ji}\}_{j=1,i=1}^{n\ ,\ n_j}$ denote the set of points in n chunklets after the transformation, m_j^y denotes the mean of chunklet j after the transformation, and κ is a constant.

3.2 RCA: The Optimal Linear Transformation for the Euclidean Norm

Consider the general problem (2) for the family F of invertible linear transformations, and using the squared Euclidean norm to measure distances. 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 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 (2) is reduced to

$$\max_A log|A| \quad s.t. \ \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j} ||y_{ji} - m_j^y||_2^2 \leq \kappa.$$

3. This non-intuitive divergence is a result of the generalization of information theory to continuous variables, that is, the result of ignoring the discretization constant in the definition of differential entropy.

Multiplying a solution matrix A by $\lambda > 1$ increases both the $log|A|$ argument and the constrained sum of within chunklet distances. Hence the maximum is achieved at the boundary of the feasible region, and the constraint becomes an equality. The constant κ only determines the scale of the solution matrix, and is not important in most clustering and classification tasks, which essentially rely on relative distances. Hence we can set $\kappa = 1$ and solve

$$\max_A log|A| \quad s.t. \ \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j}||y_{ji} - m_j^y||_2^2 = 1. \tag{3}$$

Let $B = A^tA$; since B is positive definite and $\log|A| = \frac{1}{2}\log|B|$, Problem (3) can be rewritten as

$$\max_{B\succ 0} log|B| \quad s.t. \ \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j}||x_{ji} - m_j||_B^2 = 1, \tag{4}$$

where $||.||_B$ denotes the Mahalanobis distance with weight matrix B. The equivalence between the problems is valid since for any $B \succ 0$ there is an A such that $B = A^tA$, and so a solution to (4) gives us a solution to (3) (and vice versa).

The optimization problem (4) can be solved easily, since the constraint is linear in B. The solution is $B = \frac{1}{D}\hat{C}^{-1}$, where $\hat{C}$ is the average chunklet covariance matrix (1) and D is the dimensionality of the data space. This solution is identical to the Mahalanobis matrix compute by RCA up to a global scale factor, or in other words, RCA is a scaled solution of (4).

3.3 Dimensionality Reduction

We now solve the optimization problem (4) for the family of general linear transformations, that is, $Y = AX$ where $A \in \mathcal{M}_{K\times D}$ and $K \leq D$. In order to obtain workable analytic expressions, we assume that the distribution of X is a multivariate Gaussian, from which it follows that Y is also Gaussian with the entropy

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

Following the same reasoning as in Section 3.2 we replace the inequality with equality and let $\kappa = 1$. Hence the optimization problem becomes

$$\max_A \log|A\Sigma_x A^t| \quad s.t. \ \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j}||x_{ji} - m_j||_{A^tA}^2 = 1. \tag{5}$$

For a given target dimensionality K, the solution of the problem is Fisher linear discriminant (FLD),[4] followed by the whitening of the within chunklet covariance in the reduced space. A sketch of the proof is given in Appendix A. The optimal RCA procedure therefore includes dimensionality reduction. Since the FLD transformation is computed based on the estimated within chunklet covariance matrix, it is essentially a semi-supervised technique, as described in Section 6. Note that after the FLD step, the within class covariance matrix in the reduced space is always diagonal, and Step 3 of RCA amounts to the scaling of each dimension separately.

4. Fisher Linear Discriminant is a linear projection A from $\mathcal{R}^D$ to $\mathcal{R}^K$ with $K < D$, which maximizes the determinant ratio $\max_{A\in\mathcal{M}_{K\times D}} \frac{AS_tA^t}{AS_wA^t}$, where S_t and S_w denote the total covariance and the within class covariance respectively.

4. RCA and the Minimization of Within Class Distances

In order to gain some intuition about the solution provided by the information maximization criterion (2), let us look at the optimization problem obtained by reversing the roles of the maximization term and the constraint term in problem (4):

$$\min_B \frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j}||x_{ji}-m_j||_B^2 \quad s.t.\ \ |B|\geq 1. \tag{6}$$

We interpret problem (6) as follows: a Mahalanobis distance B is sought, which minimizes the sum of all within 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 (6) to

$$\min_B \sum_{j=1}^{n}\sum_{i=1}^{n_j}||x_{ji}-m_j||_B^2-\lambda\log|B| \quad s.t.\ \ \lambda\geq 0,\ \ \lambda\log|B|=0. \tag{7}$$

Differentiating this Lagrangian shows that the minimum is given by $B=|\hat{C}|^{\frac{1}{D}}\hat{C}^{-1}$, where $\hat{C}$ 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 consider the related problem of learning a Mahalanobis distance using side information in the form of pairwise constraints (Chunklets of size >2 are not considered). It is assumed that in addition to the set of positive constraints Q_P, one is also given access to a set of negative constraints Q_N–a set of pairs of points known to be dissimilar. Given these sets, they pose the following optimization problem:

$$\min_B \sum_{(x_1,x_2)\in Q_P}||x_1-x_2||_B^2 \qquad s.t.\ \sum_{(x_1,x_2)\in Q_N}||x_1-x_2||_B\geq 1,\quad B\succeq 0. \tag{8}$$

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

In order to allow a clear comparison of RCA with (8), we reformulate the argument of (6) using only within chunklet pairwise distances. For each point x_{ji} in chunklet j we have

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

Problem (6) can now be rewritten as

$$\min_B \frac{1}{N}\sum_{j=1}^{n}\frac{1}{n_j^2}\sum_{i=1}^{n_j}||\sum(x_{ji}-x_{jk})||_B^2 \quad s.t.\ \ |B|\geq 1. \tag{9}$$

When only chunklets of size 2 are given, as in the case studied by Xing et al. (2002), (9) reduces to

$$\min_B \frac{1}{2N}\sum_{j=1}^{n}||x_{j1}-x_{j2}||_B^2 \quad s.t.\ \ |B|\geq 1. \tag{10}$$

Clearly the minimization terms in problems (10) and (8) are identical up to a constant ($\frac{1}{2N}$). The difference between the two problems lies in the constraint term: the constraint proposed by 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. As a result Xing et al. (2002) are faced with a much harder optimization problem, resulting in a slower and less stable algorithm.

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

$$\prod_{j=1}^{n}\prod_{i=1}^{n_j}\frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\tfrac{1}{2}(x_{ji}-m_j)^t\Sigma^{-1}(x_{ji}-m_j)\right).$$

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

$$\sum_{j=1}^{n}\sum_{i=1}^{n_j}||x_{ji}-m_j||_B^2 - N\log|B|, \tag{11}$$

where N is the total number of points in chunklets. Maximizing the log-likelihood is equivalent to minimizing (11), whose minimum is obtained when B equals the RCA Mahalanobis matrix (1). Note, moreover, that (11) is rather similar to the Lagrangian in (7), where the Lagrange multiplier is replaced by the constant N. Hence, under Gaussian assumptions, the solution of Problem (7) is probabilistically justified by a maximum likelihood formulation.

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

$$\hat{C}(n,k)=\frac{1}{n}\sum_{j=1}^{n}\frac{1}{k-1}\sum_{i=1}^{k}(x_{ji}-m_i)(x_{ji}-m_i)^t,$$

where $\hat{C}(n,k)$ denotes the empirical mean of the covariance estimators produced by each chunklet. It is shown in Appendix B that the variance of the elements $\hat{C}_{ij}$ of the estimating matrix is bounded by

$$Var(\hat{C}_{ij}(n,k)) \le (1+\frac{1}{k-1})Var(\hat{C}_{ij}(1,nk)), \tag{12}$$

where $\hat{C}_{ij}(1,nk)$ is the estimator when all the $N=nk$ points are known to belong to the same class, thus forming the best estimate possible from N points. This bound shows that the variance of the RCA estimator rapidly converges to the variance of the best estimator, even for chunklets of small size. For the smallest possible chunklets, of size 2, the variance is only twice as high as the best possible.

6. Dimensionality Reduction

As noted in Section 2, RCA may include dimensionality reduction. We now turn to address this issue in detail. Step 3 of the RCA algorithm decreases the weight of principal directions along which the within class covariance matrix 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 within class covariance matrix, and consider the squared distance between two points from the same class $||x_1-x_2||^2$. We can diagonalize the within

class covariance matrix using an orthonormal transformation which does not change the distance. Therefore, let us assume without loss of generality that the covariance matrix 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 $E[(x_1^i - x_2^i)^2] = 2\lambda^i$. After whitening these values become $2D$ and 2, respectively. Let us define the weight of dimension i, $W(i) \in [0,1]$, as

$$W(i) = \frac{E[(x_1^i - x_2^i)^2]}{E[||x_1 - x_2||^2]}$$

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

$$\frac{W_{before}(i)}{W_{after}(i)} = \frac{\lambda^i}{\frac{1}{D}\sum_{j=1}^{D} \lambda^j}. \tag{13}$$

In Equation (13) we observe that the weight of each principal dimension increases if its initial within class 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 a high dimensional space, the optional dimensionality reduction in RCA (Step 2) becomes mandatory.

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 labelled training data, and the method therefore falls within supervised learning. We now extend FLD, using the same information theoretic criterion, to the case of partial supervision in the form of equivalence constraints. Specifically, denote by S_t and S_w the estimators of the total covariance and the within class covariance respectively. FLD maximizes the determinant ratio

$$\max_{A \in \mathcal{M}_{K \times D}} \frac{AS_tA^t}{AS_wA^t} \tag{14}$$

by solving a generalized eigenvector problem. The row vectors of the optimal matrix A are the first K eigenvectors of $S_w^{-1}S_t$. In our case the optimization problem is of the same form as in (14), with the within chunklet covariance matrix from (1) playing the role of S_w. We compute the projection matrix using SVD in the usual way, and term this FLD variant cFLD (constraints based FLD).

To understand the intuition behind cFLD, note that both PCA and cFLD remove dimensions with small total variance, and hence reduce the risk of RCA amplifying irrelevant dimensions with small variance. However, unsupervised PCA may remove dimensions that are important for the discrimination between classes, if their total variability is low. Intuitively, better dimensionality reduction can be obtained by comparing the total covariance matrix (used by PCA) to the within class covariance matrix (used by RCA), and this is exactly what the partially supervised cFLD is trying to accomplish in (14).

The cFLD dimensionality reduction can only be used if the rank of the within chunklet covariance matrix is higher than the dimensionality of the initial data space. If this condition does not hold, we use PCA to reduce the original data dimensionality as needed. The procedure is summarized below in Algorithm 2.

Algorithm 2 Dimensionality reduction: Step 2 of RCA

Denote by D the original data dimensionality. Given a set of chunklets $\{C_j\}_{j=1}^{n}$ do

1. Compute the rank of the estimated within chunklet covariance matrix $R = \sum_{j=1}^{n}(|C_j| - 1)$, where $|C_j|$ denotes the size of the j'th chunklet.
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. Compute the total covariance matrix estimate S_t, and estimate the within class covariance matrix using $S_w = \hat{C}$ from (1). Solve (14), and use the resulting A 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, and that all the data is sampled from a stationary source. Such conditions are usually not met in the case of biological learning systems, or artificial sensor systems that interact with a gradually changing environment. Consider for example a system that tries to cluster images of different people collected by a surveillance camera in gradually changing illumination conditions, such as those caused by night and day changes. In this case different distance functions should be used during night and day times, and we would like the distance used by the system to gradually adapt to the current illumination conditions. An online algorithm for distance function learning is required to achieve such a gradual adaptation.

Here we briefly present an online implementation of RCA, suitable for a neural-network-like architecture. In this implementation a weight matrix $W \in \mathcal{M}_{D\times D}$, initiated randomly, is gradually developed to become the RCA transformation matrix. In Algorithm 3 we present the procedure for the simple case of chunklets of size 2. The extension of this algorithm to general chunklets is briefly described in Appendix C.

Algorithm 3 Online RCA for point pairs

Input: a stream of pairs of points (x_1^T, x_2^T), where x_1^T x_2^T are known to belong to the same class.
Initialize W to a symmetric random matrix with $||W|| << 1$.
At time step T do:

- receive pair x_1^T, x_2^T;
- let $h = x_1^T - x_2^T$;
- apply W to h, to get $y = Wh$;
- update $W = W + \eta(W - yy^tW)$.

where $\eta > 0$ determines the step size.

Assuming local stationarity, the steady state of this stochastic process can be found by equating the mean update to 0, where the expectation is taken over the next example pair (x_1^{T+1}, x_2^{T+1}). Using

the notations of Algorithm 3, the resulting equation is

$$E[\eta(W - yy^tW)] = 0 \quad \Rightarrow \quad E[I - yy^t] = I - WE[hh^t]W^t = 0 \quad \Rightarrow \quad W = PE[hh^t]^{-\frac{1}{2}},$$

where P is an orthonormal matrix $PP^t = I$. The steady state W is the 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 within chunklet covariance matrix. Thus W converges to the RCA transformation of the input population up to an orthonormal transformation. The resulting transformation is geometrically equivalent to RCA, since the orthonormal transformation P preserves vector norms and angles.

In order to evaluate the stability of the online algorithm we conducted simulations which confirmed that the algorithm converges to the RCA estimator (up to the transformation P), if the gradient steps decrease with time ($\eta = \eta_0/T$). However, the adaptation of the RCA estimator for such a step size policy can be very slow. Keeping η constant avoids this problem, at the cost of producing a noisy RCA estimator, where the noise is proportional to η. Hence η can be used to balance this tradeoff between adaptation, speed and accuracy.

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.

Section 8.2 presents a more realistic surveillance application in which equivalence constraints are gathered automatically from a Markovian process. In Section 8.3 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 and Merz, 1998). The evaluation of different metrics below is presented using *cumulative neighbor purity* graphs, which display the average (over all data points) percentage of correct neighbors among the first k neighbors, as a function of k.

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 data set. 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 more specific issues: In 8.1.4 we look more closely at the two steps of RCA, Step 2 (cFLD dimensionality reduction) and Step 3 (whitening w.r.t. $\hat{C}$), and study their contribution to performance in isolation. In 8.1.5 the retrieval performance of RCA is compared with the

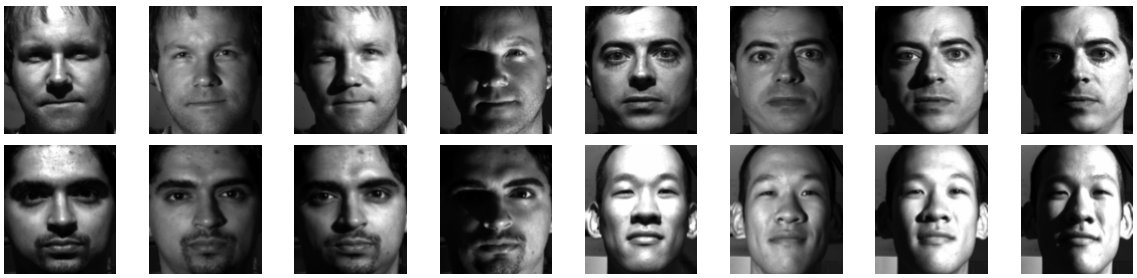

Figure 2: A subset of the YaleB database which contains 1920 frontal face images of 30 individuals taken under different lighting conditions.

algorithm presented by Xing et al. (2002). Finally in 8.1.6 we evaluate the effect of chunklets sizes on retrieval performance, and compare it to the predicted effect of chunklet size on the variance of the RCA estimator.

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 data set 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 60 PCA coefficients. Figure 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 L data points from the data set, and is asked to give his own labels to all the points, effectively partitioning the data set into equivalence classes. Each teacher therefore provides both positive and negative constraints. Note however that RCA only uses the positive constraints thus gathered. The total number of points in chunklets grows linearly with TL, the number of data points seen by all teachers. We control this amount, which provides a loose bound on the number of points in chunklets, by varying the number of teachers T and keeping L constant. We tested a range of values of T for which TL is 10%, 30%, or 75% of the points in the data set.[5]

The parameter L controls the distribution of chunklet sizes. More specifically, we show in Appendix D that this distribution is controlled by the ratio $r = \frac{L}{M}$ 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

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

roughly 2.9 and we typically obtain many small chunklets. Figure 3 shows a histogram of typical chunklet sizes, as obtained in our experiments.[6]

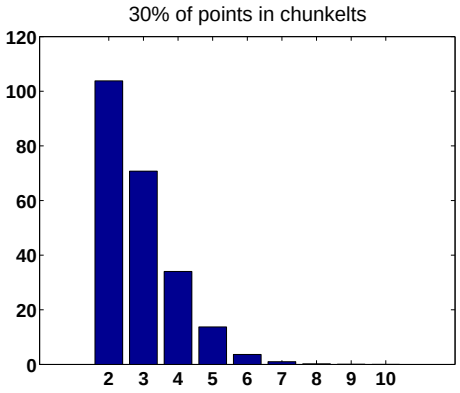

Figure 3: Sample chunklet size distribution obtained using the distributed learning scenario on a subset of the yaleB data set with 1920 images from $M = 30$ classes. L is chosen such that $r = \frac{L}{M} = 2$. The histogram is plotted for distributed learning with 30% of the data points in chunklets.

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 and Pentland, 1991): this unsupervised method reduces the dimensionality of the data using PCA, and compares the images using the Euclidean metric in the reduced space. Images were normalized to have zero mean and unit variance.
- Fisherfaces (Belhumeur et al., 1997): this supervised method starts by applying PCA 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: the RCA algorithm with dimensionality reduction as described in Section 6, that is, PCA followed by cFLD. We varied the amount of data in constraints provided to RCA, using the *distributed learning* paradigm described above.

The left panel in Figure 4 shows the results of the different methods. The graph presents the performance of RCA for low, moderate and high amounts of constrained points. As can be seen, even with low amounts of equivalence constraints the performance of RCA is much closer to the performance of the supervised FLD than to the performance of the unsupervised PCA. With Moderate and high amounts of equivalence constraints RCA achieves neighbor purity rates which are

6. We used a different sampling scheme in the experiments which address the effect of chunklet size, see Section 8.1.6.

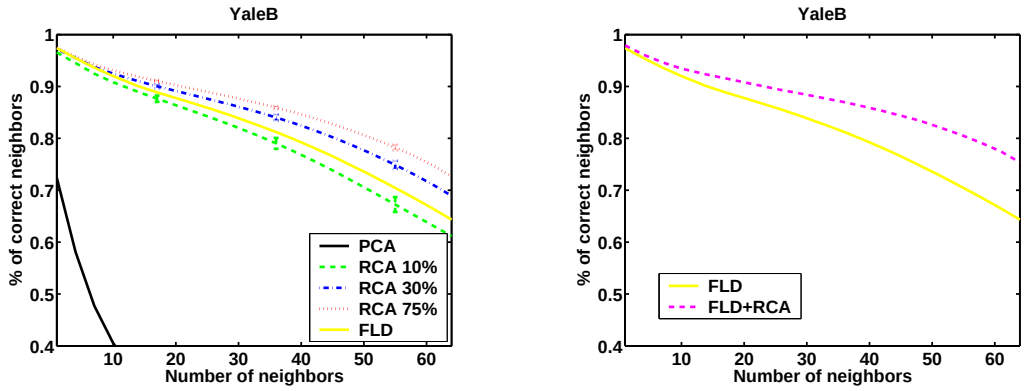

Figure 4: Left: Cumulative purity graphs for the following algorithms and experimental conditions: Eigenface (PCA), RCA 10%, RCA 30%, RCA 75%, and Fisherface (FLD). The percentages stated for RCA are the fractions of data points presented to the 'distributed learning' oracle, as discussed in Section 8.1.2. The data was reduced to dimension 60 using PCA for all the methods. It was then further reduced to dimension 30 using cFLD in the three RCA variants, and using FLD for the Fisherface method. Results were averaged over 50 constraints realizations. The error bars give the Standard Errors of the Mean (SEMs). Right: Cumulative purity graphs for the fully supervised FLD, with and without fully labelled RCA. Here RCA dramatically enhances the performance of FLD.

higher than those achieved by the fully supervised Fisherfaces method, while relying only on fragmentary chunklets with unknown class labels. This somewhat surprising result stems from the fact that the fully supervised FLD in these experiments was not followed by whitening.

In order to clarify this last point, note that RCA can also be used when given a fully labelled 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 the standard, fully supervised FLD, which whitens the output of FLD w.r.t the within class covariance. The right panel in Figure 4 shows comparative results of FLD with and without whitening in the fully labelled case.

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. Figure 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. The Eigenfaces (Turk and Pentland, 1991) method did not produce similar results.

8.1.4 SEPARATING THE CONTRIBUTION OF THE DIMENSIONALITY REDUCTION AND WHITENING STEPS IN RCA

Figure 4 presents the results of RCA including the semi-supervised dimensionality reduction of cFLD. While this procedure yields the best results, it mixes the separate contributions of the two main steps of the RCA algorithm, that is, dimensionality reduction via cFLD (Step 2) and whitening

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.

of the inner chunklet covariance matrix (Step 3). In the left panel of Figure 6 these contributions are isolated.

It can be seen that when cFLD and whitening are used separately, they both provide considerable improvement in performance. These improvements are only partially dependent, since the performance gain when combining both procedures is larger than either one alone. In the right panel of Figure 6 we present learning curves which show the performance of RCA with and without dimensionality reduction, as a function of the amount of supervision provided to the algorithm. For small amounts of constraints, both curves are almost identical. However, as the number of constraints increases, the performance of RCA dramatically improves when using cFLD.

8.1.5 COMPARISON WITH THE METHOD OF XING ET AL.

In another experiment we compared the algorithm of Xing et al. (2002) to RCA on the YaleB data set using code obtained from the author's web site. The experimental setup was the one described in Section 8.1.2, with 30% of the data points presented to the distributed learning oracle. While RCA uses only the positive constraints obtained, the algorithm of Xing et al. (2002) was given both the positive and negative constraints, as it can make use of both. Results are shown in Figure 7, showing that this algorithm failed to converge when given high dimensional data, and was outperformed by RCA in lower dimensions.

8.1.6 THE EFFECT OF DIFFERENT CHUNKLET SIZES

In Section 5 we showed that RCA typically provides an estimator for the within class covariance matrix, which is not very sensitive to the size of the chunklets. This was done by providing a bound on the variance of the elements in the RCA estimator matrix $\hat{C}(n,k)$. We can expect that lower variance of the estimator will go hand in hand with higher purity performance. In order to empirically test the effect of chunklets' size, we fixed the number of equivalence constraints, and

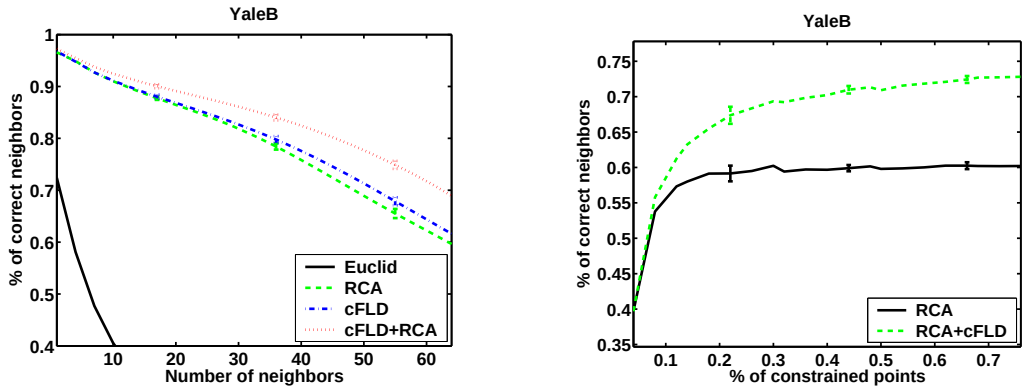

Figure 6: Left: Cumulative purity graphs for 4 experimental conditions: original space, RCA without cFLD, cFLD only, and RCA with cFLD (using the Euclidean norm in all cases). The data was reduced to 60 dimensions using unsupervised PCA. The semi supervised techniques used constraints obtained by distributed learning with 30% of the data points. RCA without cFLD was performed in the space of 60 PCA coefficients, while in the last 2 conditions dimensionality was further reduced to 30 using the constraints. Results were averaged over 50 constraints realizations. Right: Learning curves–neighbor purity performance for 64 neighbors as a function of the amount of constraints. The performance is measured by averaging (over all data points) the percentage of correct neighbors among the first 64 neighbors. The amount of constraints is measured using the percentage of points given to the distributed learning oracle. Results are averaged over 15 constraints realizations. Error bars in both graphs give the standard errors of the mean.

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 $P = 1920$ points) and dividing it into chunklets of size S.[7]

The results can be seen in Figure 8. As expected the performance of RCA improves as the size of the chunklets increases. Qualitatively, this improvement agrees with the predicted improvement in the RCA estimator's variance, as most of the gain in performance is already obtained with chunklets of size $S = 3$. Although the bound presented is not tight, other reasons may account for the difference between the graphs, including the weakness of the Gaussian assumption used to derive the bound (see Section 9), and the lack of linear connection between the estimator's variance and purity performance.

8.2 Using RCA in a Surveillance Application

In this application, a stationary indoor surveillance camera provided short video clips whose beginning and end were automatically detected based on the appearance and disappearance of moving targets. 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.

7. When necessary, the remaining $mod(0.3P, S)$ points were gathered into an additional smaller chunklet.

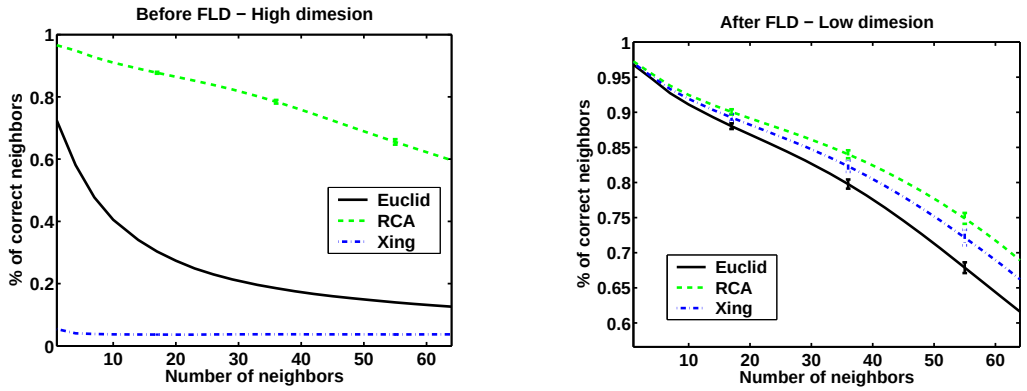

Figure 7: The method of Xing et al. (2002) and RCA on the YaleB facial image data set. Left: Neighbor purity results obtained using 60 PCA coefficients. The algorithm of Xing et al. (2002) failed to converge and returned a metric with chance level performance. Right: Results obtained using a 30 dimensional representation, obtained by applying cFLD to the 60 PCA coefficients. Results are averaged over 50 constraints realizations. The error bars give the standard errors of the mean.

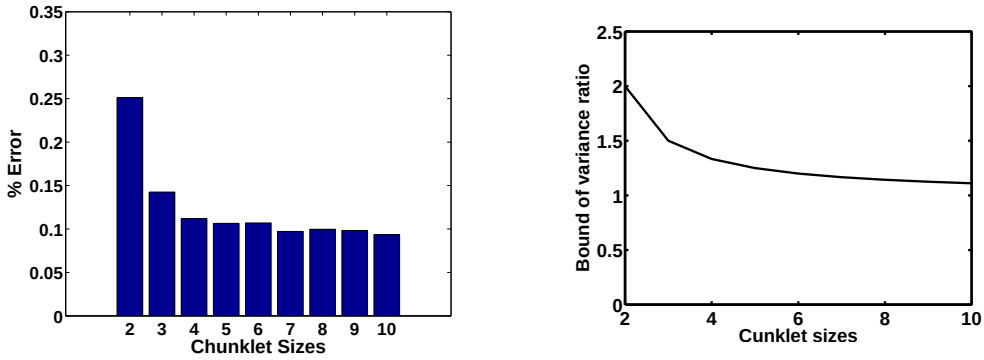

Figure 8: Left: Mean error rate on all 64 neighbors on the yaleB data set when using 30% of the data in chunklets. In this experiment we varied the chunklet sizes while fixing the total amount of points in chunklets. Right: the theoretical bound over the ratio between the variance of the RCA matrix elements and the variance of the best possible estimator using the same number of points (see inequality 12). The qualitative behavior of the graphs is similar, seemingly because a lower estimator variance tends to imply better purity performance.

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

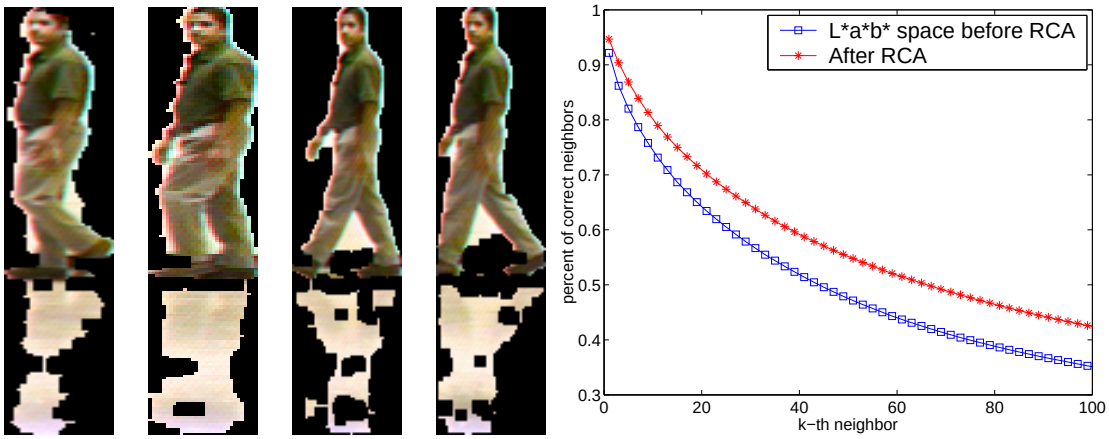

Figure 9: Left: several images from a video clip of one subject. Right: cumulative neighbor purity results before and after RCA.

varying occlusions, reflections and (most importantly from our perspective) highly variable illumination. In fact, the illumination changed dramatically across the scene both in intensity (from brighter to darker regions), and in spectrum (from neon light to natural lighting). Figure 9 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, that is, to overcome the general problem of irrelevant variability due to the varying illumination. This is accomplished by the RCA algorithm.

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). We used over 6000 images from 130 clips (chunklets) of 20 different people. Figure 9 shows the cumulative neighbor purity 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). However, the effect of RCA on retrieval performance here is lower than the effect gained with the YaleB data base. While there may be several reasons for this, an important factor is the difference between the way chunklets were obtained in the two data sets. The automatic gathering of chunklets from a Markovian process tends to provide chunklets with dependent data points, which supply less information regarding the within class covariance matrix.

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 Q_P of pairwise positive equivalence constraints (or chunklets of size 2). We compared the following clustering algorithms:

a. K-means using the default Euclidean metric and no side-information (Fukunaga, 1990).

b. Constrained K-means + Euclidean metric: the K-means version suggested by Wagstaff et al. (2001), in which a pair of points $(x_i, x_j) \in Q_P$ is always assigned to the same cluster.

c. Constrained K-means + the metric proposed by Xing et al. (2002): The metric is learnt from constraints in Q_P. For fairness we replicated the experimental design employed by Xing et al. (2002), and allowed the algorithm to treat all unconstrained pairs of points as negative constraints (the set Q_N).

d. Constrained K-means + RCA: Constrained K-means using the RCA Mahalanobis metric learned from Q_P.

e. EM: Expectation Maximization of a Gaussian Mixture model (using no side-information).

f. Constrained EM: EM using side-information in the form of equivalence constraints (Shental et al., 2004), when using the RCA distance metric as the initial metric.

Clustering algorithms *a* and *e* are unsupervised and provide respective lower bounds for comparison with our algorithms *d* and *f*. Clustering algorithms *b* and *c* compete fairly with our algorithm *d*, 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 Q_P, and (ii) "much" side-information. The set of pairwise similarity constraints Q_P was generated by choosing a random subset of all pairs of points sharing the same class identity c_i. Initially, there are N 'connected components' of unconstrained points, where N is the number of data points. Randomly choosing a pairwise constraint decreases the number of connected components by 1 at most. In the case of "little" ("much") side-information, pairwise constraints are randomly added until the number of different connected components K_c is roughly $0.9N$ ($0.7N$). As in the work of Xing et al. (2002), no negative constraints were sampled.

Following Xing et al. (2002) we used a normalized accuracy score, the "Rand index" (Rand, 1971), 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 the probability that the algorithm's decision regarding the label equivalence of two points agrees with the decision of the "true" assignment c.[8]

Figure 10 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

8. As noted by 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, x_j) such that x_i and x_j are from the same cluster with probability 0.5 and from different clusters with probability 0.5, so that "matches" and "mismatches" are given the same weight.

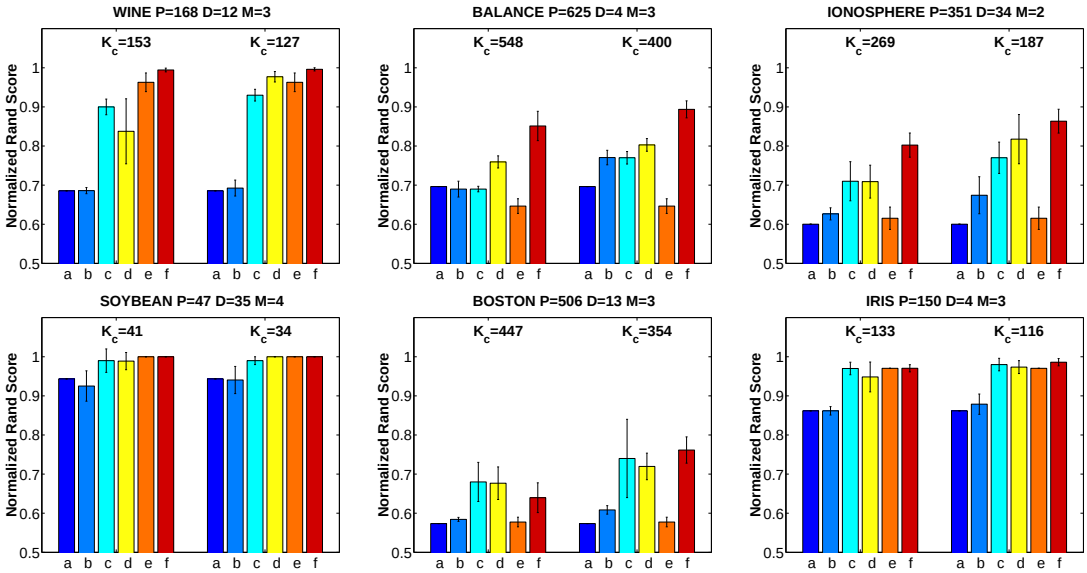

Figure 10: Clustering accuracy on 6 UCI data sets. In each panel, the six bars on the left correspond to an experiment with "little" side-information, and the six bars on the right correspond to "much" side-information. From left to right the six bars correspond respectively to the algorithms described 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., 2004) over the feature space created by RCA. Also shown are P–the number of points, M–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. The error bars give the standard deviations. In all experiments we used K-means with multiple restarts as in done by Xing et al. (2002).

and unconstrained versions). Generally in the context of K-means, we observe that using equivalence constraints to find a better metric improves results much more than using this information to constrain the algorithm. RCA achieves comparable results to those reported by Xing et al. (2002), despite the big difference in computational cost between the two algorithms (see Section 9.1).

The last two algorithms in our comparisons use the EM algorithm to compute a generative Gaussian Mixture Model, and are therefore much more computationally intensive. We have added these comparisons because EM implicitly changes the distance function over the input space in a locally linear way (that is, like a Mahalanobis distance). It may therefore appear that EM can do everything that RCA does and more, without any modification. The histogram bins marked by (e) in Figure 10 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 Figure 10.

9. Discussion

We briefly discuss running times in Section 9.1. The applicability of RCA in general conditions is then discussed in 9.2.

9.1 Runtime Performance

Computationally RCA relies on a few relatively simple matrix operations (inversion and square root) applied to a positive-definite square matrix, whose size is the reduced dimensionality of the data. This can be done fast and efficiently and is a clear advantage of the algorithm over its competitors.

9.2 Using RCA when the Assumptions Underlying the Method are Violated

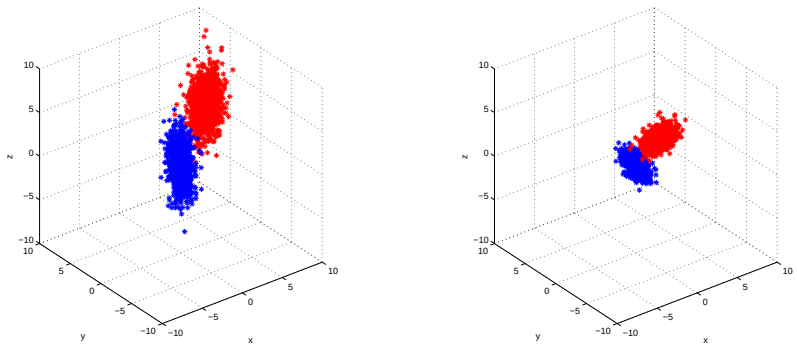

Figure 11: Extracting the shared component of the covariance matrix using RCA: In this example the data originates from 2 Gaussian sources with the following diagonal covariance matrices: $diag(C_1) = (\varepsilon, 1, 2)$ and $diag(C_2) = (1, \varepsilon, 2)$. (a) The original data points (b) The transformed data points when using RCA. In this example we used all of the points from each class as a single chunklet and therefore the chunklet covariance matrix is the average within-class covariance matrix. As can be seen RCA clearly downscales the irrelevant variability in the Z axis, which is the shared component of the 2 classes covariance matrices. Specifically, the eigenvalues of the covariance matrices for the two classes are as follows (for $\varepsilon = 0.1$): class 1–$(3.947, 1.045, 0.009)$ before RCA, and $(1.979, 1.001, 0.017)$ after RCA; class 2–$(3.953, 1.045, 0.010)$ before RCA, and $(1.984, 1.001, 0.022)$ after RCA. In this example, the condition numbers increased by a factor of 3.78 and 4.24 respectively for both classes.

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

1. The classes have multi-variate normal distributions.
2. All the classes share the same covariance matrix.
3. The points in each chunklet 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. Without it, in a distribution-free model, RCA is the best linear transformation optimizing the criteria presented in Sections 3-4: maximal mutual information, and minimal within-chunklet distance. These criteria are reasonable as long as the classes are approximately convex (as assumed by the use of the distance between chunklet's points and chunklet's means). In order to investigate this point empirically, we used Mardia's statistical tests for multi-variate normality (Mardia, 1970). These tests (which are based on skewness and kurtosis) showed that all of the data sets used in our experiments are significantly non-Gaussian (except for the Iris UCI data set). Our experimental results therefore clearly demonstrate that RCA performs well when the distribution of the classes in the data is not multi-variate normal.

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 within class covariance matrix. When this assumption fails, RCA effectively extracts the shared component of all the classes covariance matrices, if such component exists. Figure 11 presents an illustrative example of the use of RCA on data from two classes with different covariance matrices. A quantitative measure of RCA's partial success in such cases can be obtained from the change in the *condition number* (the ratio between the largest and smallest eigenvalues) of the within-class covariance matrices of each of the classes, before and after applying RCA. Since RCA attempts to whiten the within-class covariance, we expect the condition number of the within-class covariance matrices to decrease. This is indeed the case for the various classes in all of the data sets used in our experimental results.

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 (see Section 8.2).

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 improvements in clustering and distance based retrieval.

Appendix A. Information Maximization with Non-Invertible Linear Transformations

Here we sketch the proof of the claim made in Section 3.3. As before, we denote by $\hat{C}$ the average covariance matrix of the chunklets. We can rewrite the constrained expression from Equation 5 as

$$\frac{1}{N}\sum_{j=1}^{n}\sum_{i=1}^{n_j}(x_{ji}-m_j)^t A^t A(x_{ji}-m_j) = tr(A^t A\hat{C}) = tr(A^t\hat{C}A).$$

Hence the Lagrangian can be written as

$$\log|A\Sigma_x A^t| - \lambda(tr(A\hat{C}A^t)-1).$$

Differentiating the Lagrangian with respect to A gives

$$\Sigma_x A^t (A\Sigma_x A^t)^{-1} = \lambda \hat{C} A^t.$$

Multiplying by A and rearranging terms, we get $\frac{I}{\lambda} = A\hat{C}A^t$. Hence as in RCA, A must whiten the data with respect to the chunklet covariance $\hat{C}$ in a yet to be determined subspace. We can now use the equality in (5) to find λ:

$$\begin{aligned} tr(A\hat{C}A^t) = tr(\frac{I}{\lambda}) = \frac{K}{\lambda} = 1 &\Longrightarrow \lambda = K \\ &\Longrightarrow A\hat{C}A^t = \frac{1}{K} I, \end{aligned}$$

where K is the dimension of the projection subspace.

Next, since in our solution space $A\hat{C}A^t = \frac{1}{K} I$, it follows that $\log|A\hat{C}A^t| = K \log\frac{1}{K}$ holds for all points. Hence we can modify the maximization argument as follows:

$$\log|A\Sigma_x A^t| \quad = \quad \log\frac{|A\Sigma_x A^t|}{|A\hat{C}A^t|} + K\log\frac{1}{K}$$

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 K eigenvectors of $\hat{C}^{-1}\Sigma_x$. Denote by G the solution matrix for this unconstrained problem. This matrix orthogonally diagonalizes both $\hat{C}$ and Σ_x, so $G\hat{C}G^t = \Lambda_1$ and $G\Sigma_x G^t = \Lambda_2$ for Λ_1, Λ_2 diagonal matrices. In order to enforce the constraints we define the matrix $A = \sqrt{\frac{1}{K}}\Lambda_1^{-0.5} G$ 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 G since A is a product of G and another full ranked matrix. It can also be shown that A satisfies the constraints and is thus the solution of the Problem (5).

Appendix B. Variance Bound on the RCA Covariance Estimator

In this appendix we prove Inequality 12 from Section 5. Assume we have $N = nk$ data points $X = \{x_{ji}\}_{i=1,j=1}^{n,k}$ 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 m_i the mean of chunklet i, the unbiased RCA estimator of this covariance matrix is

$$\hat{C}(n,k) = \frac{1}{n}\sum_{j=1}^{n}\frac{1}{k-1}\sum_{i=1}^{k}(x_{ji}-m_i)(x_{ji}-m_i)^T.$$

It is more convenient to estimate the convergence of the covariance estimate for data with a diagonal covariance matrix. We hence consider a diagonalized version of the covariance, and return to the original covariance matrix toward the end of the proof. Let U denote the diagonalization transformation of the covariance matrix C of the Gaussian sources, that is, $UCU^t = \Lambda$ where Λ is a diagonal matrix with $\{\lambda_i\}_{i=1}^{D}$ on the diagonal. Let $Z = UX = \{z_{ji}\}_{i=1,j=1}^{n,k}$ denote the transformed data. Denote the transformed within class covariance matrix estimation by $\hat{C}^u(n,k) = U\hat{C}(n,k)U^t$, and denote the chunklet means by $m_i^u = Um_i$. We can analyze the variance of $\hat{C}^u$ as follows:

$$var(\hat{C}^u(n,k)) = var[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{k-1}\sum_{j=1}^{k}(z_{ji}-m_i^u)(z_{ji}-m_i^u)^T]$$

$$= \frac{1}{n} var[\frac{1}{k-1}\sum_{j=1}^{k}(z_{ji}-m_1^u)(z_{ji}-m_1^u)^T]. \quad (15)$$

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

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

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

Hence (15) is simply

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

Replacing $N = nk$, we can write

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

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

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

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

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

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

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

Appendix C. Online RCA with Chunklets of General Size

The online RCA algorithm can be extended to handle a stream of chunklets of varying size. The procedure is presented in Algorithm 4.

The steady state of the weight matrix W can be analyzed in a way similar to the analysis in Section 3. The result is $W = PE[\frac{1}{n}\sum_{i=1}^{n}(x_i^T - m^T)(x_i^T - m^T)^t]^{-\frac{1}{2}}$ where P is an orthonormal matrix, and so W is equivalent to the RCA transformation of the current distribution.

Algorithm 4 Online RCA for chunklets of variable size

Input: a stream of chunklets where the points in a chunklet are known to belong to the same class.
Initialize W to a symmetric random matrix with $||W|| << 1$.
At time step T do:

- receive a chunklet $\{x_1^T,...,x_n^T\}$ and compute its mean $m^T = \frac{1}{n}\sum_{i=1}^n x_i^T$;
- compute n difference vectors $h_i^T = x_i^T - m^T$;
- transform h_i^T using W, to get $y_i^T = Wh_i^T$;
- update $W = W + \eta\sum_{i=1}^n (W - y_i^T (y_i^T)^t W)$.

where $\eta > 0$ determines the step size.

Appendix D. The Expected Chunklet Size in the Distributed Learning Paradigm

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 L data points. Let us assume that the data contains M equiprobable classes, and that the size of the data set is large relative to L. Define the random variables x_i^j as the number of points from class i observed by teacher j. Due to the symmetry among classes and among teachers, the distribution of x_i^j is independent of i and j, thus defined as x. It can be well approximated by a Bernoulli distribution $B(L, \frac{1}{M})$, while considering only $x \geq 2$ (since $x = 0, 1$ do not form chunklets). Specifically,

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

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

$$p(x=0) = (1-\frac{1}{M})^L \approx e^{-\frac{L}{M}} \quad , \quad p(x=1) = \frac{L}{M}(1-\frac{1}{M})^{L-1} \approx \frac{L}{M}e^{-\frac{L}{M}}.$$

Using these approximations, we can derive an approximation for the expected chunklet size as a function of the ratio $r = \frac{L}{M}$:

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

References

Y. Adini, Y. Moses, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. In *proc. of IEEE PAMI*, volume 19(7), pages 721–732, 1997.

A. Bar-Hillel, T. Hertz, N. Shental, and D. weinshall. Learning distance functions using equivalence relations. In T. Fawcett and Nina Mishra, editors, *20th International Conference on Machine Learning*, Wahington DC, 2003. AAAI press.

S. Becker and G. E. Hinton. A self-organising neural network that discovers surfaces in random-dot stereograms. *Nature*, 355:161–163, 1992.

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

A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

M. Bilenko, S. Basu, and R.J Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proc. 21st International Conf. on Machine Learning*, Banff Canada, 2004. AAAI press. URL citeseer.ist.psu.edu/705723.html.

C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998. URL http://www.ics.uci.edu/∼mlearn/MLRepository.html.

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

G. Chechik and N. Tishby. Extracting relevant structures with side information. In S Becker, S. Thrune, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. The MIT Press, 2003.

K. Fukunaga. *Statistical Pattern Recognition*. Academic Press, San Diego, 2nd edition, 1990.

P. Geladi and B. Kowalski. Partial least squares regression: A tutorial. *Analytica Chimica Acta*, 185:1–17, 1986.

T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbor classification and regression. In David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 409–415. The MIT Press, 1996.

T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers, 1998.

D. Klein, S. Kamvar, and C. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *Proc. 19th International Conf. on Machine Learning*, 2002. URL citeseer.nj.nec.com/klein02from.html.

R. Linsker. An application of the principle of maximum information preservation to linear systems. In David S. Touretzky, editor, *Advances in Neural Information Processing Systems*, pages 186–194. Morgan Kaufmann, 1989.

K. V. Mardia. Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 36: 519–530, 1970.

W. M. Rand. Objective criteria for the evaluation of clustering method. *Journal of the American Statistical Association*, 66(366):846–850, 1971.

N. Shental, A. Bar-Hillel, T. Hertz, and D. Weinshall. Computing gaussian mixture models with em using equivalence constraints. In Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

N. Shental, T. Hertz, D. Weinshall, and M. Pavel. Adjustment learning and relevant component analysis. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *7th European Conference on Computer Vision*, volume 4, 2002.

J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural Computation*, 12(6):1247–1283, 2000.

B. Thompson. *Canonical correlation analysis: Uses and interpretation*. Newbury Park CA: SAGE, 1984.

S. Thrun. Is learning the n-th thing any easier than learning the first? In David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 640–646. The MIT Press, 1996.

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

M. A. Turk and A.P Pentland. Face recognition using eigenfaces. In *Proc. of IEEE CVPR*, pages 586–591, 1991.

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

E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems*, volume 15. The MIT Press, 2002.