The Minimum Volume Ellipsoid Metric

Karim T. Abou-Moustafa and Frank P. Ferrie

The Artificial Perception Laboratory,
Centre for Intelligent Machines, McGill University,
3480 University street, Montreal, QC, Canada H3A 2A7
{karimt,ferrie}@cim.mcgill.ca

Abstract. We propose an unsupervised "local learning" algorithm for learning a metric in the input space. Geometrically, for a given query point, the algorithm finds the minimum volume ellipsoid (MVE) covering its neighborhood which characterizes the correlations and variances of its neighborhood variables. Algebraically, the algorithm maximizes the determinant of the local covariance matrix which amounts to a convex optimization problem. The final matrix parameterizes a Mahalanobis metric yielding the MVE metric (MVEM). The proposed metric was tested in a supervised learning task and showed promising and competitive results when compared with state of the art metrics in the literature.

1 Introduction

The fact that many learning algorithms, supervised, unsupervised, or semi-supervised, depend mainly on a "representative" and a "meaningful" distance metric in the input space, imposes the problem of finding such a metric in the very core problems of machine learning algorithms. The various benefits pointed out in [5,6,9] of having a metric that can better describe similarities in the absence of *a priori* knowledge or side–information [5,9], point to the need for such metrics. This is reflected in the current literature by many new algorithms that tackled the problem directly and indirectly [5,6,7,8,9,11,12], and showed promising results in that regard. The contribution of this paper builds on this research with an algorithm for learning a new distance metric in the input space. The new metric, called the minimum volume ellipsoid metric (MVEM), can be seen as a generalization of existing metrics induced by recent learning algorithms.

Two main objectives and advantages lie behind the MVEM design. First, it is desirable to have a metric that does not depend on *a priori* knowledge, side information as in [5,9], or data labels as in [6,7]. Second, the metric should not depend on the learning paradigm. That is, for any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, labeled or unlabeled, from a training set or test set, it is desirable to replace $\|\mathbf{x} - \mathbf{y}\|_2$ by a distance function $D(\mathbf{x}, \mathbf{y})$ which carries more information on the similarity between $\mathbf{x}$ and $\mathbf{y}$.

Our outlook is statistical, with a motivation rooted in robust statistics, and links to maximum likelihood estimation (MLE) with Gaussian distributions. The MVEM is a parameterized version of the general Mahalanobis distance function with a special structure imposed on the symmetric positive definite matrix

F.A. Hamprecht, C. Schnörr, and B. Jähne (Eds.): DAGM 2007, LNCS 4713, pp. 335–344, 2007.

defining the metric. The special structure stems from combined statistical and geometrical properties, with a useful algebraic interpretation that is used later in the proposed algorithm for learning the MVEM. The proposed algorithm, called MiniVenn (or minimum volume ellipsoid of nearest neighbours), depends primarily on the concept of locality in the input space. For a given query point, with its assigned neighborhood (*k-Nearest-Neighbors*, or *ϵ-ball*), similarities between the query point and its neighbors can be found by means of the neighborhood's covariance matrix (i.e. local covariance). In an ideal setting, if the query point is the mean of a normally distributed neighborhood, the covariance matrix defines an ellipsoid which, in principle, should approximately [1] cover (or enclose) the neighborhood [2]. The induced Mahalanobis distance can measure the similarity between the mean and the neighboring points while taking correlations and variances into consideration. With real life data, however, this is hardly the case. Due to the obvious non-normality of the neighboring points with respect to the query point, the curse of dimensionality effect, nonlinearity of the data, and noise, such an ellipsoid poorly covers the desired neighborhood and the induced metric becomes unreliable.

The first motivation for the proposed MVE approach to define a metric stems from the above observation. If the ellipsoid is reshaped to cover the desired neighborhood, as MLE with a Gaussian component does, one can expect that the covariance matrix will better reflect the local structure. Another primary motivation, stems from the statistics literature [14], where the Mahalanobis distance is well known to expose outliers by assigning them very large distance values. Therefore, should the Mahalanobis distance be well parameterized by an accurate estimate of the covariance matrix, one can expect more accurate distances and similarity measures [14].

The paper is organized as follows: First, the motivation for the MVEM is presented in Section 2. Section 3 presents the algorithm for learning the MVEM, followed by a review of related work and similarities with other metric learning algorithms. Experimental results are illustrated in Section 4, and finally, conclusions are drawn in Section 5.

2 Motivation for the MVEM

The Euclidean distance has been and is still extensively used and embedded in many algorithms of the pattern recognition and machine learning literature. There are many reasons, however, that render the Euclidean metric completely inappropriate. First, if the norm is to deal with very high dimensional structured data, the curse of dimensionality and its consequences are inevitable. Second, effects of the random noise in the data and missing values will be reflected in the Euclidean metric. Third, despite an adequately sized training set, it is very likely that the data set is not balanced, resulting in high and low density areas in the

[1] Due to the infinite support of the Gaussian.

[2] The axes of the ellipsoid lie along the eigenvectors of the covariance matrix, and the squares of the axes' lengths are its eigenvalues.

input space, causing fragile estimation of densities and intrinsic dimensionality. Moreover, the very definition of the Euclidean metric ignores the effect of scale, variance and correlations of and among the variables. Thus the Euclidean metric may not reflect the true geometry of the underlying manifold structure of points under consideration.

The Mahalanobis distance, on the other hand, is well known in the robust statistics literature as an outlier detector [14]. It exposes outliers by assigning them very large Mahalanobis distances. This, however, depends on an accurate estimate of the covariance matrix that parameterizes this distance. An intuitive approach for obtaining such an estimate is via MLE with Gaussian components. This, however, requires a large number of samples, and converges to a local optimum which might result in an unnecessarily large variance. Our proposed approach is to use a robust estimator for the covariance matrix parameterizing the Mahalanobis distance, where robustness is defined in a statistical sense [17,18]. The MVE [15,16] is such a robust estimator with desirable properties such as intuitive geometric meaning, its formulation as a convex optimization problem which has a global unique solution, and its minimum variance.

2.1 Properties of the Mahalanobis Distance

The Euclidean distance between two points, $\mathbf{x} = (x_1, \ldots, x_d)^T$ and $\mathbf{y} = (y_1, \ldots, y_d)^T$, in the d-dimensional space $\mathbb{R}^d$ is defined as: $\mathrm{D}_E(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y})}$. It follows that all non zero points with the same distance from the origin $\mathbf{o}$, satisfy: $x_1^2 + \cdots + x_d^2 = c^2$, $c \in \mathbb{R}^+$, which is the equation of a spheroid. This means that all components of an observation $\mathbf{x}$ contribute equally to the Euclidian distance from $\mathbf{x}$ to the origin or any other reference point. Hence, $\mathrm{D}_E(\mathbf{x}, \mathbf{y})$ is meaningful when the data have an equal variance across all its dimensions.

Real life data, however, are usually measurements from various sources at different scales, and are subject to various noise sources. To account for such variability, each component can be assigned a weight that is proportional to the amount of variation across its values, such that components with high variability should receive less weight than those with low variability. Let $\mathbf{u} = (x_1/s_1, \ldots, x_d/s_d)$, and $\mathbf{v} = (y_1/s_1, \ldots, y_d/s_d)$; then, the distance between $\mathbf{u}$ and $\mathbf{v}$ will be: $\mathrm{D}_E(\mathbf{u}, \mathbf{v}) = \mathrm{D}_\Sigma(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \mathbf{y})}$ where $\mathbf{\Sigma} = diag(s_1^2, \ldots, s_d^2)$, and s_j^2 is the variance of the data across dimension j. Now the distance from $\mathbf{x}$ to the origin equals $\mathrm{D}_\Sigma(\mathbf{x}, \mathbf{o}) = \sqrt{\mathbf{x}^T \mathbf{\Sigma}^{-1} \mathbf{x}}$, and all points with the same distance to the origin satisfy: $(x_1/s_1)^2 + \cdots + (x_d/s_d)^2 = c^2$, which is the equation of an ellipsoid centered at the origin with its principal axes aligned to the coordinate axes.

By considering correlations between components, this will allow the ellipsoid to rotate its axes and to increase/decrease its size, yielding the well known general form of the distance between two points $\mathbf{x}$ and $\mathbf{y}$, the Mahalanobis distance: $\mathrm{D}_\Sigma(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \mathbf{y})}$, where $\mathbf{\Sigma}$ is a symmetric positive definite matrix, $\mathbf{\Sigma} \succ 0$. Consequently, points with the same distance to the origin satisfy: $\mathbf{x}^T \mathbf{\Sigma}^{-1} \mathbf{x} = c^2$, which is the general equation of an ellipsoid centered at the origin.

The general Mahalanobis distance enjoys all the properties of distance functions that are defined on a metric space. That is, for any three points $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ in $\mathbb{R}^d$, the following are satisfied: Symmetry: $\mathrm{D}_{\Sigma}(\mathbf{x},\mathbf{y}) = \mathrm{D}_{\Sigma}(\mathbf{y},\mathbf{x})$, Non negativity: $\mathrm{D}_{\Sigma}(\mathbf{x},\mathbf{y}) > 0$ if $\mathbf{x} \neq \mathbf{y}$, Self reflection: $\mathrm{D}_{\Sigma}(\mathbf{x},\mathbf{y}) = 0$ if $\mathbf{x} = \mathbf{y}$, and Triangle inequality: $\mathrm{D}_{\Sigma}(\mathbf{x},\mathbf{y}) \leq \mathrm{D}_{\Sigma}(\mathbf{x},\mathbf{z}) + \mathrm{D}_{\Sigma}(\mathbf{z},\mathbf{y})$. Also, it is worth noting that the Euclidean distance can be considered as a special case of the general Mahalanobis form by letting $\mathbf{\Sigma} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Alternatively, the general Mahalanobis distance can be seen as a projecting the original $\mathbf{x}$ on the space of $\mathbf{\Sigma}^{-1/2}$ and using the Euclidean metric in that space.

2.2 Robust Statistics and the MVE Estimator

Robust statistics [17,18] is the stability theory of statistical procedures. It systematically investigates the effects of deviations from modeling assumptions on known procedures, and if necessary, develops new better procedures [18]. The primary concern of robust statistics is distributional robustness, i.e. the shape of the true underlying distribution deviates slightly from the assumed model (usually the Gaussian law) [17]. Another concern of paramount importance is the design of estimators that can tolerate a large number of outliers before the estimate is affected. Such estimators are known to have a high breakdown point (BP). Finding robust multivariate location and scatter estimators is crucial to make other multivariate techniques such as principal component analysis and discriminant analysis more robust. In addition, distances based on these estimators are more precise than regular ones, and are better suited to expose outliers [14].

The MVE estimator [15,16] is a robust estimator for location (mean) and scatter (covariance matrix) with the highest possible BP value (50%). Geometrically, the estimator finds the minimum volume ellipsoid covering, or enclosing a given set of points. The MVE estimator is a generalization of the least median of squares (LMS) estimator [15,16] for high dimensional data sets, with the extra property of being equivariant to translation, scaling, orthogonal projection and affine transformations. Formulation of the MVE covering a data set is illustrated in the next section.

3 The Minimum Volume Ellipsoid

We consider the problem of finding the minimum volume ellipsoid (MVE) covering a set. Let $\mathcal{X} = \{\mathbf{x}_i \mid 1 \leq i \leq m, \mathbf{x}_i \in \mathbb{R}^d\}$ be a bounded set, where m is the number of vectors, and d is the dimensionality of the input space. The minimum volume ellipsoid that covers $\mathcal{X}$ is known as the *Löwner − John Ellipsoid* of the set $\mathcal{X}$ and is denoted $\mathcal{E}_{lj}$ [2]. The $\mathcal{E}_{lj}$ can be parametrized as follows:

$$\mathcal{E}_{lj} = \{\mathbf{x} \mid \|\mathbf{\Sigma}\mathbf{x} - \mathbf{b}\|_2 \leq 1\}, \tag{1}$$

where $\mathbf{\Sigma} \in \mathbb{R}^{d \times d}$, $\mathbf{\Sigma} \succ 0$, $\mathbf{x}$ and $\mathbf{b} \in \mathbb{R}^d$, and its center is $\mathbf{\Sigma}^{-1}\mathbf{b}$. The general ellipsoid can be seen as the inverse image of the Euclidean unit ball under an affine transformation. Using the fact that $\mathbf{\Sigma} \succ 0$, it follows that [2]:

$$\mathrm{V}(\mathcal{E}_{lj}) \propto \det(\mathbf{\Sigma^{-1}}) \propto \frac{1}{\det(\mathbf{\Sigma})}, \tag{2}$$

where $\mathrm{V}(\mathcal{E}_{lj})$ is the volume of the ellipsoid $\mathcal{E}_{lj}$. Finding the minimum volume ellipsoid covering $\mathcal{X}$ can be formulated as follows:

$$\min_{\Sigma} \ \log\det(\mathbf{\Sigma^{-1}}) \qquad \textit{or equivalently} \tag{3}$$

$$\max_{\Sigma} \ \log\det(\mathbf{\Sigma}) \tag{4}$$

$$\text{subject to } \ \|\mathbf{\Sigma}\mathbf{x}_i - \mathbf{b}\|_2 \leq 1, \quad i = 1, \ldots, m,$$

where the variables of this minimization are $\mathbf{\Sigma}$ and $\mathbf{b}$, with an implicit constraint that $\mathbf{\Sigma} \succ 0$ which forces the induced distance function to respect all the previously mentioned properties of a metric. The minimization in (3) is a convex optimization problem since the objective and the constraints are convex in the variables $\mathbf{\Sigma}$ and $\mathbf{b}$. This is very useful since, theoretically, it allows a global minimum to be found away from local minima. The details of this convex optimization problem are elaborated in [2].

Computing the minimum volume ellipsoid bounding or enclosing a data set can be done in several ways, and the interested reader can see [7,3,13] for a nice review. At the current stage of our research, all our experiments used the CVX MATLAB toolbox for Disciplined Convex Programming [10]. CVX is a general purpose solver that implements an interior point method algorithm that scales efficiently with small to medium size problems.

3.1 The MVE Metric and the MiniVenn Algorithm

The basic idea of the MVEM is that the metric is learned from the perspective of the point itself, should it be a training or a test point, labeled or unlabeled. This should make the metric independent from the learning paradigm since it does not depend on labels as in [6,7], nor on side–information [5,9]. In other words, the metric tries to answer this question, *How does a point perceive the similarity between itself and other neighboring points?* Based on the concept of locality, the metric tries to find the fine differences between a point and its local neighbors, and major differences between the neighborhood and other points in the space.

To find such a metric, we present the Minimum Volume Ellipsoid of Nearest Neighbors (MiniVenn) algorithm, shown in Algorithm 1. Given a query point $\mathbf{x}_q$, the algorithm finds the MVE with $\mathbf{x}_q$ as its center and covering its m nearest neighbors. Recalling the relation in (2), the MiniVenn actually finds a symmetric positive definite matrix with maximum determinant that can parameterize a Mahalanobis distance function from the perspective of $\mathbf{x}_q$. The MiniVenn starts by finding the m nearest neighbors of $\mathbf{x}_q$ using the Euclidean metric; this is

Algorithm 1. Minimum Volume Ellipsoid of Nearest Neighbors (MiniVenn): *finds a symmetric positive definite matrix* $\mathbf{\Sigma}_q$ *with maximum determinant*

.

```
Require: $\mathbf{X}_{n\times d}$, $\mathbf{x}_q$, and $m$, where $\mathbf{X}$ is the training set with $n$ $d$-dimensional samples,
    $\mathbf{x}_q$ is the query point, and $m \geq d+1$ is a hyper-parameter that controls the size of
    the neighborhood.
Find the set $\mathcal{X}_q$ that has the $m$ nearest neighbours of $\mathbf{x}_q$ using the Euclidean metric.
Find the MVE with center $\mathbf{x}_q$ that covers $\mathcal{X}_q$ using the following convex optimization:

    $\max_{\Sigma} \quad \log\det(\mathbf{\Sigma}_q)$
    $\text{subject to} \quad \|\mathbf{\Sigma}_q\mathbf{x}_j - \mathbf{b}\|_2 \leq 1, \quad 1 \leq j \leq m, \quad \mathbf{x}_j \in \mathcal{X}_q$
    $\qquad \|\mathbf{\Sigma}_q\mathbf{x}_q - \mathbf{b}\|_2 = 0$

    {The second constraint insures that $\mathbf{x}_q$ will be the center of the MVE, since its
    center is defined as $\mathbf{\Sigma}_q^{-1}\mathbf{b}$.}
return $\mathbf{\Sigma}_q$
```

equivalent to considering a spheroid around the query point that covers its m nearest neighbors. Starting from the Euclidean metric is equivalent to setting the initial covariance matrix to the identity matrix which simply reflects our *a priori* assumption that all variables are independent with zero mean and unit variance. This can also be considered as bootstrapping the MVE metric. Next, the convex optimization in (3), reshapes the spheroid into a MVE covering the same set, thereby learning the variances and correlations within and across all variables. The learned $\mathbf{\Sigma}_q$ will be used to measure the Mahalanobis distance from any point $\mathbf{x}$ to $\mathbf{x}_q$. Note that in terms of distances, for any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $D_{\Sigma_\mathbf{x}}(\mathbf{x}, \mathbf{y}) = D_{\Sigma_\mathbf{x}}(\mathbf{y}, \mathbf{x})$ by symmetry. However $D_{\Sigma_\mathbf{x}}(\mathbf{x}, \mathbf{y}) \neq D_{\Sigma_\mathbf{y}}(\mathbf{x}, \mathbf{y})$ since the reference covariance matrix is different.

The advantage of the MVEM stems from its flexibility to be used in any learning paradigm. In an unsupervised setting, and with existence of side–information [5,9], similar samples can be grouped in the same MVE, with the center being their mean. The same applies in the semi–supervised context, when given partially labeled data, which is similar to clustering with side–information. In both cases, m acts as a hyper–parameter that controls clustering affinity.

For supervised learning, two scenarios can take place for learning the metric. In the first, one can learn a full metric $\mathrm{D}_\Sigma : \mathbb{R}^d \times \mathbb{R}^d :\mapsto \mathbb{R}$, where MiniVenn will find a MVE for each training point $\mathbf{x}_i$ (i.e. $\mathbf{\Sigma}_i$). On the one hand, in concept, this makes the MVEM relatively close to the metric found in [5]. On the other hand, as an algorithmic approach, this makes MiniVenn close to the initial step of manifold learning algorithms such as [11,12], albeit without the dimensionality reduction step. The second scenario, on the contrary, learning can be done online using the lazy learning approach [4], where the MVE is computed only on request when a query point $\mathbf{x}_q$ is presented to the training set, and m can be optimized by cross validation. In both scenarios, since there is a training phase to optimize m, the MVEM will generalize well on unseen data sets.

3.2 Links to Other Metric Learning Algorithms

Before proceeding, let us review some basic identities. Let $\mathbf{A} \in \mathbb{R}^{d \times d} \succ 0$ be a symmetric positive definite matrix; then, by eigen decomposition, $\mathbf{A} = V \Lambda V^T$, $\Lambda = diag(\lambda_1, \cdots, \lambda_d)$, where the $\lambda_j s$ are the eigenvalues of $\mathbf{A}$, and the columns of V are its eigenvectors. Then, $\det(\mathbf{A}) = \prod_{j=1}^{d} \lambda_j \succ 0$, and $\det(\mathbf{A}^{-1}) = \prod_{j=1}^{d} 1/\lambda_j \succ 0$. Also, $\|A\|_F^2 = \|V \Lambda V^T\|_F^2 = \|\Lambda\|^2 \succ 0$.

Algorithms found in [5,6,7,8] learn, in general, a parametrized Mahalanobis metric of the form $\mathrm{D}_A(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})\mathbf{A}^{-1}(\mathbf{x} - \mathbf{y})}$, and differences between these algorithms are due to the different structures imposed on $\mathbf{A}$. The work in [5,7,8] directly finds an $\mathbf{A}^{-1}$ with minimum $\|\mathbf{A}^{-1}\|_F^2$, i.e. a MVE, accompanied with another term that stems from the problem context, such as minimization of classification error, minimization of distances between similar points, or learning relative relations between points, which ultimately leads to the different flavors of algorithms. Alternatively, the metric in [6] finds directly a matrix $\mathbf{A}^{-1} \succ 0$ that minimizes the Kullback–Leibler divergence between an observed and a desired distribution that will collapse classes to a single point.

The MVEM is relatively close to [5], with no explicit restriction on $\mathbf{A}$ to encode similar samples differently than all other samples, as this is left to the parameter m in the MiniVenn algorithm (1). This can be considered as letting the data speak for itself, but it will be interesting to apply some local constraints similar to those found in [5,9]. Moreover, it does not group similar points together; rather, it is a locally based algorithm. Unlike the proposed methods in [6,7,8], the MVEM does not have any constraints from the problem context; rather, it is the parameter m that is adjusted according to the problem context, and hence the flexibility of the algorithm.

The concept of parameterized Mahalanobis distances also has interesting links with some recent manifold learning algorithms. Charting a manifold [11] and Manifold Parzen Window [12], initially, fix a Gaussian at each training point and then find a local covariance matrix $\mathbf{A}$ based on the neighboring samples. To overcome the poor representation of local covariance matrices, MPW has an embedded dimensionality reduction step by means of spectral decomposition of $\mathbf{A}$, and flattens those components with very small singular values. This acts as a regularized MLE with a Gaussian component, thereby yielding Gaussian pancakes, i.e. a Mahalanobis distance based on a low dimensional projection. In charting a manifold, however, the charting step includes a maximum likelihood estimation, i.e. directly maximizing $\det(\mathbf{A^{-1}})$ (see [2] p. 355), yielding a rotation and an increase of the ellipsoid size, thereby covering a more representative neighborhood. However, neither of the two approaches guarantees a small variance for their estimate.

4 Experimental Results

The experimental setting was designed to validate the MVEM concept and to show its potential. Since the primary objective is to have a metric that can replace the Euclidean metric in any learning paradigm, it is intuitive to evaluate the "*pure*" impact of the new metric without additional aid or complexities from sophisticated learning algorithms. That is, select a simple algorithm that depends solely on the Euclidean metric and replace this metric with the proposed MVEM. The basic and classical k–Nearest Neighbors classifier ($k = 1$) meets such specifications, where optimization of the hyper–parameter m can be based on the training error. Given a training set and a test set, we find the nearest neighbor of each test point using the MVEM.

Table 1. Error rates(%) for EUC, LM, RCA, XING, and MVEM on eleven data sets from the UCI repository using one–out–of–sample criterion

DataSet	classes	size	dim.	EUC	LM	RCA	XING	MVEM
Liver Disorders (bupa)	2	345	6	37.7	38.5	34.5	37.6	35.6
Glass	7	214	9	26.2	34.1	28.5	26.2	23.8
Ionosphere	2	351	34	11.4	16.2	8.3	12.5	8.8
Iris	3	150	4	4.0	2.7	4.0	2.6	2.6
New–Thyroid	3	215	5	5.1	6.9	4.1	5.1	3.2
Diabetes (pima)	2	768	8	32.0	32.4	30.4	32.0	30.4
satImage	6	4435/2000	36	10.6	12.4	22.4	10.6	10.0
Sonar	2	208	60	17.8	24.5	15.9	17.7	15.4
WDBC	2	569	30	9.1	7.9	8.8	9.1	8.4
Wine	3	168	12	4.5	7.3	2.2	10.1	2.8
Yeast	10	1484	6	48.2	46.9	47.2	no convergence	47.5

The MVEM was compared with four other metrics (or metric learning algorithms). Initially, the MVEM was compared to the regular Euclidean metric (EUC), and the Local–Mahalanobis (LM) metric obtained by the local covariance matrix of each test point and its m neighbors from the training set, where m is also optimized based on the training error. Next, the source codes for XING [5] and RCA [9] were downloaded from the authors' web sites in order to compare their performance with the MVEM. XING [5] and RCA [9] algorithms were specifically designed for unsupervised learning with side–information. Unlike the MVEM, these algorithms not only depend on side–information; it is the amount of available side–information that determines their performance. By providing all the true labels for these two algorithms, the uncertainty in the labels is eliminated, and the algorithms should perform at their best.

All five metrics (or algorithms); EUC, LM, RCA [9], XING [5], and MVEM were run on eleven problems from the UCI Machine Learning Repository [1], shown in Table 1, with various sizes, dimensionalities, and difficulties. Except

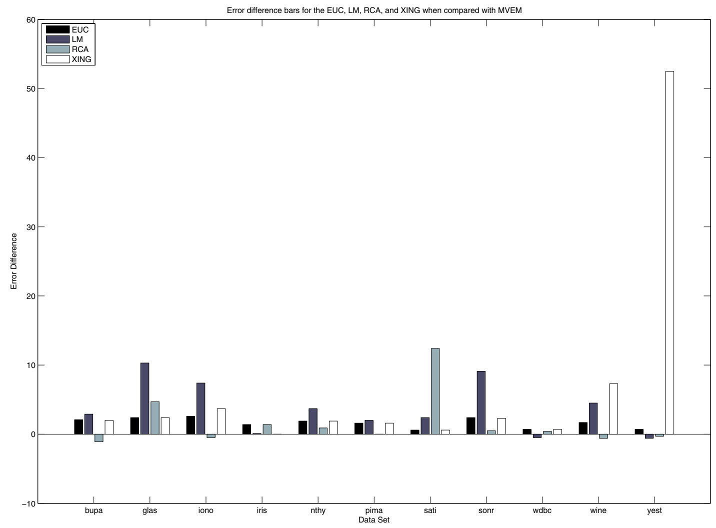

Fig. 1. Error difference bars for EUC, LM, RCA, and XING when compared with MVEM. A positive difference implies that the MVEM is better than the other metric, and negative difference implies the contrary.

for the Sat–Image data set which had explicit training and test sets, the error rates, shown in Table 1, are based on a one–out–of–sample performance using n runs where n is the number of samples in the data set. In order to speed–up the CVX solver, as a preprocessing step, principal component analysis (PCA) was applied to all data sets, except bupa, new–thyroid, pima, and yeast, to keep 99% of their total variance. PCA was obtained from the Sat–Image training set, and from the training set after each split, from all other data sets. The hyper–parameter m (for the case of MVEM and LM) was optimized based on the best training error on the Sat–Image data set, and on the leave–one–out training error after each split for all other data sets.

Discussion: Figure 1 shows error difference bars between all metrics and the MVEM. It can be seen that in overall performance, the MVEM is consistently as good or better than other metrics for most of the cases. In the light of these results, it is worth noting that, in the cases where RCA is slightly better, it is important to recall that RCA was designed for the case when partially labelled data are available, and it achieved this performance when it was provided with the full set of data labels. This is unlike the MVEM which did not use such extra information during its training phase. This makes the MVEM very promising for learning problems where the samples are not labelled, partially labelled, or manually annotated with side–information.

5 Conclusion

We have introduced an unsupervised local–learning algorithm for learning a metric in the input space. The metric has desirable statistical and geometrical properties, the corresponding algorithm does not depend on side–information, and showed promising and competitive results when compared with state of the art metric learning algorithms that depend on side–information.

References

1. Newman, D., Hettich, S., Blake, C., Merz, C.: UCI Repository of Machine Learning Databases University of California, Irvine, Dept. of Information and Computer Sciences (1998), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
2. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge Univ. Press (2004)
3. Sun, P., Freund, R.: Computation of Minimum Volume Covering Ellipsoids. Journal of Operations Research 52, 690–706 (2004)
4. Atkenson, C., Moore, A., Schaal, S.: Local Weighted Learning. AI Review, 11–73 (1997)
5. Xing, E., Ng, A., Jordan, M., Russell, S.: Distance Metric Learning with Application to Clustering with Side-Information. In: NIPS 15, MIT Press, Cambridge (2003)
6. Globerson, A., Roweis, S.: Metric Learning by Collapsing Classes. In: NIPS 18, MIT Press, Cambridge (2006)
7. Weinberger, K., Blitzer, J., Saul, L.: Distance Metric Learning for Large Margin Nearest Neighbor Classification. In: NIPS 18, MIT Press, Cambridge (2006)
8. Schultz, M., Joachims, T.: Learning a Distance Metric from Relative Comparisons. In: NIPS 16, MIT Press, Cambridge (2004)
9. Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D.: Learning a Mahalanobis Metric from Equivalence Constraints. JMLR 6, 937–965 (2005)
10. Grant, M., Boyd, S., Ye, Y.: Matlab Software for Disciplined Convex Programming (2005), `http://www.stanford.edu/~boyd/cvx`
11. Brand, M.: Charting a Manifold. In: NIPS 15, MIT Press, Cambridge (2003)
12. Vincent, P., Bengio, Y.: Manifold Parzen Windows. In: NIPS 15, MIT Press, Cambridge (2003)
13. Dolia, A., De–Bie, T., Harris, C., Shawe–Taylor, J., Titterington, D.: The Minimum Volume Covering Ellipsoid Estimation in Kernel–Defined Feature Spaces. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) ECML 2006. LNCS (LNAI), vol. 4212, Springer, Heidelberg (2006)
14. Rousseeuw, P., Van Driessen, K.: Algorithm for the Minimum Covariance Determinant Estimator. Technometrics 41(3), 212–223 (1999)
15. Rousseeuw, P.: Least Median of Squares Regression. Journal of American Statistical Association 79(388), 871–880 (1984)
16. Rousseeuw, P.: Multivariate Estimation with High Breakdown Point. In: Proceedings of the fourth Pannonian Symposium on Mathematical Statistics, vol. 3, pp. 283–297 (1983)
17. Huber, P.: Robust Statistics. John Wiley Press, Chichester (1981)
18. Hampel, F.: Robust Statistics, ETH–Zurich, Tech. Report No. 94 (2001)