Local Metric Learning on Manifolds with Application to Query–based Operations

Karim Abou-Moustafa and Frank Ferrie
{karimt,ferrie}@cim.mcgill.ca

The Artificial Perception Laboratory
Centre for Intelligent Machines, McGill University
3480 University street, Montreal, QC, Canada H3A 2A7

Abstract. We first investigate the combined effect of data complexity, curse of dimensionality and the definition of the Euclidean distance on the distance measure between points. Then, based on the concepts underlying manifold learning algorithms and the minimum volume ellipsoid metric, we design an algorithm that learns a local metric on the lower dimensional manifold on which the data is lying. Experiments in the context of classification on standard benchmark data sets showed very promising results when compared to state of the art algorithms, and consistent improvements over the Euclidean distance in the context of query–based learning.

1 Introduction

The Euclidean distance between two points $\mathbf{x}$ and $\mathbf{y}$ in an Euclidean space $\mathbb{R}^d$ or finding the nearest neighbour(s) or matching points to a query point, are two basic operations for a plethora of algorithms in the literature on pattern recognition, machine learning, computer vision, image retrieval and data mining. Indeed, the solid roots of the Euclidean distance in geometry, its intuitive meaning and its simple computation make it unarguably a prime choice as a similarity measure or simply as the true distance between points. Here, our concern is focused on query–based operations where a typical scenario is to have a set $\mathcal{X}$ of high dimensional vectors (images, image patches, feature vectors, etc) and it is required to find a set of nearest neighbors or matches to a point that is either in the same set or a new incoming one. In such situations, the Euclidean distance is usually the measure of choice for assessing the similarity between points. Despite its success in many applications, and with a sober look at the Euclidean distance, there are few reasons that raise a question on the full validity of this metric when dealing with high dimensional real life data.

A first reason for that is the curse of dimensionality. That is, in high dimensional spaces, the distance between every pair of points is almost the same for a wide variety of data distributions and distance functions [5]. Hence, in such high dimensional spaces, the notion of similarity becomes less accurate. The second reason stems from the complexity of data arising from real life applications. Usually, such data are: (1) High dimensional, highly structured and nonlinear such as images, text documents, proteins, etc. (2) Measurements from various sources at different scales and with various degrees of variability and correlation, and (3) Prone to various sources of noise that may largely deviate measurements and raise outliers in the data. The third reason, which is the definition of the Euclidean distance, combines and builds over the aforementioned ones. By expanding the squared Euclidean norm $\|\mathbf{x}-\mathbf{y}\|_2^2$ to $(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T\mathbf{I}(\mathbf{x}-\mathbf{y})$, where $\mathbf{I}$ is the identity matrix, one directly obtains an instance of the general family of Mahalanobis distances between points $\mathbf{x}$ and $\mathbf{y}$: $D_S(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T\mathbf{S}^{-1}(\mathbf{x}-\mathbf{y})$, where $\mathbf{S}$ is a symmetric and positive definite matrix. Replacing $\mathbf{S}$ by $\mathbf{I}$ implies that the Euclidean distance takes it for granted that all variables are independent, the variance across all dimensions is one and that covariances among all variables are zero, a situation that is hardly attained in real life data. Therefore, the Euclidean distance, by definition, ignores the structure, scale, variance and correlations in the data and consequently it is wise to say that *"in the absence of clear evidence of Euclidean geometry, the metric structure should be inferred from the data"* [13].

Contribution : We are interested in learning a local metric for query–based operations. We combine the concepts underlying manifold learning algorithms and the minimum volume ellipsoid metric (MVEM) [1] in a unified algorithm that tries to overcome the aforementioned problems with using the Euclidean distance as a metric or as similarity measure. That is, given a data set $\mathcal{X}$ of some high dimensional points and a query point $\mathbf{x}_q$ of similar dimensionality, we are interested in learning a similarity measure based on the information in $\mathbf{x}_q$, a small neighbourhood $\mathcal{N}_{\mathbf{x}_q}$ around $\mathbf{x}_q$ and the data set $\mathcal{X}$; i.e. the metric is adaptive for each query point (whether $\mathbf{x}_q \in \mathcal{X}$ or is a new incoming one), such that each $\mathbf{x}_q$ can better define its nearest neighbors from $\mathcal{X}$. To this end, our approach will rely on defining the local metric on the lower dimensional manifold on which $\mathbf{x}_q$ lies on. This is different from previous metric learning algorithms that focused on learning a metric specifically for k–NN (nearest neighbors) classification, exemplified by [19, 12, 22], in that the proposed algorithm is unsupervised, self–adaptive for each new query point, and define the metric on the lower dimensional manifold on which the query point is lying.

2 Related Work

The earliest work on metric learning is due to Short and Fukunaga [19] where they define an optimal distance measure to minimize the difference between the finite sample k-NN error and the asymptotic k-NN error. Recently, there are arguably three main streams for metric learning: global metric learning using labels or similarity constraints (side–information) [23, 2, 22], locally fully supervised metric learning similar to [19, 12], and unsupervised metric learning exemplified by [20, 17, 14]. We briefly review some of these algorithms and the interested reader can see [24] for a comprehensive review on the topic.

Most of the algorithms in the first category learn a global metric through the general family of Mahalanobis distances $\mathrm{D}_A(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T\mathbf{A}(\mathbf{x}-\mathbf{y})$ and the differences between these algorithms are due to the constraints defining each metric. In metric learning using labels, [10] defines a differentiable probability function (softmax) using $\mathrm{D}_A(\mathbf{x},\mathbf{y})$ with $\mathbf{A}$ as its parameter. This function is optimized to maximize the probability of correct classification using the labels in the training set. In an extended work, [9] uses the same objective function to map all points that belong to the same class into a single point. Alternatively, [22] searches for a matrix $\mathbf{A}$ that defines a linear transformation such that k nearest neighbours of the same class are always kept together while samples from other classes are separated by a large margin.

In learning with similarity constraints, [18] uses similarity constraints in the form of triplet relative comparisons; i.e. for samples $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ the relative comparison information is in the form of $\mathrm{D}_A(\mathbf{x},\mathbf{y}) > \mathrm{D}_A(\mathbf{x},\mathbf{z})$. Hence, the objective is to find a matrix $\mathbf{A}$ with minimum rank that respects such constraints. Using a different form of constraints, [23] tries to find a matrix $\mathbf{A}$ that will keep similar points close to each other while keeping dissimilar points far from other, i.e. using positive and negative similarity constraints. RCA [2] in a simpler setting uses only positive constraints and tries to find a matrix $\mathbf{A}$ that keeps similar points close to each other.

3 The Minimum volume ellipsoid metric

In previous work [1], we have introduced the Minimum Volume Ellipsoid Metric (MVEM) as well as the Minimum Volume Ellipsoid of Nearest Neighbors (MiniVenn) algorithm to learn the proposed metric. The MVEM is a similarity measure that tries to mitigate the difficulties of data complexity and curse of dimensionality by using a parametrized Mahalanobis distance instead of the Euclidean distance. The MVEM is defined independently for each point (referred to as a query point) in a data set based on the information in a small neighborhood around it. Hence, the MVEM is locally defined for each query point. First, the MiniVenn is briefly reviewed to see how it can learn this local metric, then its limitations are addressed in the next subsection and finally the modified algorithm is presented with a discussion on its generalization.

Consider a data set $\mathcal{X} = \{\mathbf{x}_i \mid 1 \leq i \leq n\} \subset \mathbb{R}^d$ that is drawn from a probability distribution $p(\mathbf{x})$, and a query point $\mathbf{x}_q \in \mathbb{R}^d$ that is also assumed to be drawn from $p(.)$, where d is the dimensionality of the input space, and n is the number of samples. The first step in MiniVenn is to find the m nearest neighbors to $\mathbf{x}_q$ from the set $\mathcal{X}$ using the Euclidean distance. Under the concept of locality [3], it is assumed that in a small neighborhood $\mathcal{N}_{\mathbf{x}_q}$ around $\mathbf{x}_q$ (ϵ–ball or m nearest neighbors), points will tend to be similar and share some common properties. Such similarities can be characterized by means of the neighborhood's covariance matrix (or local covariance) $\mathbf{S}_q$. The induced Mahalanobis distance using $\mathbf{S}_q$ can measure the similarity between $\mathbf{x}_q$ and its neighbors while taking correlations and variances into considerations. In other words, the distance between $\mathbf{x}_q$ and any other point $\mathbf{x}_i \in \mathcal{X}$ will be based on the information in $\mathbf{x}_q$ and $\mathcal{N}_{\mathbf{x}_q}$ that is encoded in $\mathbf{S}_q$. However, due to the curse of dimensionality effect, the high nonlinearity of real life data and noise, estimating such a local covariance matrix using a Maximum Likelihood estimator (MLE) is hard and not reliable [7]. Therefore, instead of using a MLE, MiniVenn uses a robust estimator, the Minimum Volume Covering Ellipsoid (MVCE) estimator [16], to compute an accurate and a robust estimate for $\mathbf{S}_q$. Finally, the estimated $\mathbf{S}_q$ is used to define a more accurate Mahalanobis distance to the measure similarity between $\mathbf{x}_q$ and any other point in $\mathcal{X}$.

Limitation of the MiniVenn algorithm MiniVenn and consequently the MVEM suffered from three drawbacks. The first drawback arises from using the Euclidean metric in a high dimensional space to find the nearest neighbors of $\mathbf{x}_q$. As mentioned in the introduction, due to the curse of dimensionality effect, the notion of similarity in a small neighborhood around $\mathbf{x}_q$ will be inaccurate. Second, in order to compute the robust estimate using the MVCE estimator, MiniVenn has to solve a convex optimization problem that relies on a difficult formulation for the problem. In addition to using a general convex optimization package [11] that did not consider the special problem structure, this formulation led to a very slow algorithm for computing the MVCE. This, by its turn, hindered the usage of the MVEM in practical situations that require fast query–based operations. Finally, the literature on manifold learning algorithms assumes that the data actually lies on or near a lower dimensional nonlinear manifold that captures most of the data variability and is embedded in the high dimensional input space. MiniVenn in its current design state does not take this assumption into consideration and it is our objective to let MiniVenn defines the metric on the lower dimensional manifold on which $\mathbf{x}_q$ lies on.

Algorithm 1 Regularized Minimum Volume Ellipsoid of Nearest Neighbors : *Learns a local metric for query point $\mathbf{x}_q$ on the manifold on which $\mathbf{x}_q$ is lying.*

```
Require: $\mathcal{X}_{n\times d}$, $\mathbf{x}_q$, $m$, $\tau$ and $\rho$ where $\mathcal{X}_{n\times d}$ is the training set with $n$ $d$-dimensional samples, $\mathbf{x}_q$ is the query point, $m \geq d+1$ is a user input that controls the size of the neighborhood, $\tau > 0$ is the threshold to select the leading (tangent) directions with large eigenvalues along the manifold and $\rho \in [0,1]$ is the MVEM regularization parameter.
Find the set $\mathcal{N}_{\mathbf{x}_q}$ that has the $m$ similar points to $\mathbf{x}_q$ using the similarity measure in Section 3.1.
Compute the robust estimate of $\mathbf{S}_q$ defined by the MVCE estimator for the set $\mathcal{N}_{\mathbf{x}_q}$ and centre $\mathbf{x}_q$ using Titterington algorithm [21].
Compute the eigen decomposition of $\mathbf{S}_q = \mathbf{V}\mathbf{L}\mathbf{V}^T$ where $\mathbf{V} = [V_1 \ \dots \ V_d]$, $\mathbf{L} = \mathrm{diag}(\lambda_1, \dots, \lambda_d)$ are the matrices of eigenvectors and eigenvalues respectively and $\lambda_1 > \lambda_2 > \dots > \lambda_d$.
Select the $d_0$ leading eigenvalues such that $\lambda_{[1\ :\ d_0]} > \tau$ and form the matrix $\tilde{\mathbf{L}} = \mathrm{diag}(\rho, \dots, \rho, \frac{1}{\lambda_{d_0+1}}, \dots, \frac{1}{\lambda_d})$
return $\tilde{\mathbf{S}}_q^{-1} = \mathbf{V}\tilde{\mathbf{L}}\mathbf{V}^T$
```

3.1 The modified MiniVenn algorithm

The modified MiniVenn algorithm, shown in Algorithm (1), tries to overcome the above mentioned limitations by modifying the definition of local neighborhoods, by modifying the computation of the MVCE estimator, and finally by adding two extra steps for manifold detection. The algorithm proceeds as follows. In step 1, the algorithm defines a local neighborhood $\mathcal{N}_{\mathbf{x}_q}$ for $\mathbf{x}_q$ based on a similarity measure, explained in the following subsection, different than the Euclidean distance. In step 2, similar to the original MiniVenn, the algorithm computes the robust estimate of the covariance matrix $\mathbf{S}_q$ using the MVCE of the set $\mathcal{N}_{\mathbf{x}_q}$. The difference here is in the formulation and algorithm used to compute the MVCE of $\mathcal{N}_{\mathbf{x}_q}$. Steps 3 and 4 are new steps in the algorithm concerned with manifold detection, estimation of the local intrinsic dimensionality at $\mathbf{x}_q$, and MVEM regularization. In the following, each modification and addition will be explained in more detail.

Redefining local neighbourhoods The definition of $\mathcal{N}_{\mathbf{x}_q}$ for the query point $\mathbf{x}_q$ in the original MiniVenn used the Euclidean distance as a similarity measure to find the m nearest neighbors to $\mathbf{x}_q$. Since our major objective is to find the most similar points to $\mathbf{x}_q$, one can define a different similarity measure between points and use it to define local neighborhoods. Indeed, selecting appropriate neighbourhoods is a key factor to the success of local learning algorithms [14, 20]. A flexible and easy to compute similarity measure is the dot product between two vectors. That is, let $s_i = \mathrm{Sim}(\mathbf{x}_q, \mathbf{x}_i) = \langle \mathbf{x}_q^{'}, \mathbf{x}_i^{'} \rangle = \mathbf{x}_q^{'T} \mathbf{x}_i^{'}$, be the similarity measure between $\mathbf{x}_q$ and $\mathbf{x}_i$, where $\mathbf{x}_q^{'} = \mathbf{x}_q / \|\mathbf{x}_q\|_2$, $\mathbf{x}_i^{'} = \mathbf{x}_i / \|\mathbf{x}_i\|_2$ and $\mathbf{x}_i \in \mathcal{X}$. This is equivalent to finding the m nearest neighbors for $\mathbf{x}_q^{'}$ after normalizing all the points to lie on the unit sphere in $\mathbb{R}^d$. This similarity measure, formulated as a dot product, can be extended to accommodate different similarity measures using the kernel trick and hence can mitigate the curse of dimensionality effect.

Fast computation of the MVCE Let $\mathcal{N}_{\mathbf{x}_q} = \{\mathbf{x}_j \mid 1 \leq j \leq m,\ \mathbf{x}_j \in \mathcal{X}\}$ be the set of similar neighbors to the point $\mathbf{x}_q$ using the above defined similarity measure. The MVCE of $\mathcal{N}_{\mathbf{x}_q}$ with centre $\mathbf{x}_q$ is denoted by $\mathcal{E}$ and is parameterized by a symmetric and positive definite matrix $\mathbf{S}_q \in \mathbb{R}^{d \times d}$ as follows [4]:

$$\mathcal{E} = \{\mathbf{x}_j \mid \|\mathbf{S}_q^{-\frac{1}{2}} \mathbf{x}_j - \mathbf{b}\|_2^2 \leq 1,\ \forall j\} \tag{1}$$

where $\mathbf{b} = \mathbf{S}_q^{-\frac{1}{2}} \mathbf{x}_q$. Since $V(\mathcal{E}) \propto \det(\mathbf{S}_q)$, where $V(\mathcal{E})$ is the ellipsoid's volume, minimizing this volume can be formulated as follows:

$$\min_{\mathbf{S}_q} \ \log\det \mathbf{S}_q, \quad \text{s.t.} \ \|\mathbf{S}_q^{-\frac{1}{2}} \mathbf{x}_j - \mathbf{b}\|_2^2 \leq 1,\ \forall j \tag{2}$$

The objective and the constraints in (2) are convex in $\mathbf{S}_q$, therefore this optimization problem has a unique global optimal solution. However, as in [1], directly solving this optimization problem using standard convex optimization libraries such as CVX [11] showed to be computationally expensive and not efficient for practical situations. Alternatively, the dual of this optimization problem, thanks to Titterington [21], is easier to optimize and has a very fast and efficient algorithm for its computation (see [21] for algorithm details) :

$$\max_{\mathbf{S}_q,\Phi} \quad \log\det(\mathbf{S}_q) \quad \text{s.t.} \quad \mathbf{\Phi} \in \mathbb{R}^m, \quad \mathbf{\Phi} \geq 0, \quad \mathbf{\Phi}^T\mathbf{e} = 1 \tag{3}$$

$$\mathbf{S}_q = \sum_{j=1}^{m} \phi_j(\mathbf{x}_j - \mathbf{x}_q)(\mathbf{x}_j - \mathbf{x}_q)^T + \gamma\mathbf{I}$$

where Φ is the vector of dual variables ϕ_j, $\gamma \geq 0$ and $\gamma\mathbf{I}$ is an extra constraint that guarantees a minimal diameter of the ellipsoid in all directions. This would prevent the ellipsoid from collapsing to zero volume especially in large dimensional spaces [6].

Manifold detection The literature on manifold learning algorithms assumes that despite the high dimensionality of the data in the input space, most of the data variability can be captured by far fewer dimensions known as the intrinsic dimensionality of the data. Accordingly, it is assumed that the data lies on or near (due to noise) a lower dimensional nonlinear manifold that is embedded in the high dimensional input space. Real life data, however, are usually highly structured, nonlinear and prone to various sources of noise. Consequently, the data might not actually lie on a single nonlinear manifold, but rather on or near several disconnected nonlinear manifolds [13]. It is this last observation that motivates our objective to let MiniVenn define the metric on the lower dimensional manifold on which $\mathbf{x}_q$ is lying. To detect this manifold, MiniVenn performs an eigen decomposition and a regularization step for the robust estimate $\mathbf{S}_q$. The benefit of the eigen decomposition is twofold: (1) It can estimate the intrinsic dimensionality of the data using Fukunaga's algorithm [8] by means of the number of dominating eigenvalues of $\mathbf{S}_q$ (which is the role of parameter τ), and (2) The orthogonal eigenvectors of $\mathbf{S}_q$ decide which vectors are tangent or normal to the underlying manifold. That is, the eigenvector associated with the smallest eigenvalue (or lowest variance in $\mathcal{N}_{\mathbf{x}_q}$) is normal to the manifold, while the eigenvector associated with the largest eigenvalue (or highest variance in $\mathcal{N}_{\mathbf{x}_q}$) is tangent to the manifold and the latter is the main direction of interest since it is the direction that goes along the manifold and contributes the most to the dissimilarity measure in the neighborhood of $\mathbf{x}_q$. Note that in a d–dimensional space and for a d_0–dimensional manifold with $d_0 \ll d$, there will be approximately d_0 tangent vectors associated with the largest eigenvalues.

The Mahalanobis distance, however, measures the similarity using $\mathbf{S}_q^{-1}$, i.e. by taking the inverse of the eigenvalues, thus assigning small weights to high variance components (tangent eigenvectors) and large weights to low variance components (normal eigenvectors). It is at this point that the regularization parameter ρ is needed to emphasize the contribution of the main tangent vectors over the contribution of normal and less significant tangent vectors. More specifically, ρ influences the notion of similarity of the MVEM, however this is task dependent since it can tune the MVEM according to the objective of the task under consideration.

Table 1. The fifteen UCI [15] data sets used in our experiments with the number of classes (C), size (S) and dimensionality (D). m_{opt} is the optimal neighbourhood size for each k–NN classifier ($k = 1$, $k = 3$, $k = 5$) using the MVEM.

ID	Dataset	C	S	D	m_{opt}	ID	Dataset	C	S	D	m_{opt}
bal	Balance	3	625	4	7,8,10	new	NewThyroid	3	215	5	11,6,6
bup	Bupa	2	345	6	15,8,16	pim	Pima	2	768	8	22,12,9
gla	Glass	7	214	9	11,16,20	seg	Segment	7	2086	18	33,21,20
hou	HouseVotes	2	341	16	43,19,19	tic	TicTacToe	2	958	9	10,10,19
ion	Ionosphere	2	350	33	42,35,85	wdb	WDBC	2	569	30	31,43,31
iri	Iris	3	150	4	8,7,9	win	Wine	3	168	13	14,35,31
lym	Lymphography	4	148	18	41,49,33	yea	Yeast	10	1484	6	8,17,7
pag	Pageblocks	5	5473	10	29,23,27						

3.2 Generalization of the MVEM

Generalization of the MVEM is controlled by the MiniVenn's four parameters: m, τ, ρ and implicitly γ to compute the optimization in (3). While m and τ reflect the topological properties of the data, ρ influences the notion of similarity of the obtained metric. Using [8], τ can be fixed for a data set since it is a threshold on the normalized eigenvalues. Similarly, γ can be fixed for each data set separately although it was fixed to either 0 or 0.1 in all our experiments. More attention however, is required to select m and ρ. A large value of m will over smooth the main tangent directions of the patch on which $\mathbf{x}_q$ is lying, while a very small value will lead to crude and rather fragile estimates of these directions. An intuitive approach is to select m and ρ via an optimization procedure. This can be achieved by linking the two parameters to an objective function that can be optimized. The optimal objective function in this case would be the objective function of the task under consideration. Implicitly, this means that the metric (or the MVEM) will be tuned to maximize or minimize this objective function. For instance, in the case of our experiments on query-based learning, m and ρ were optimized by a grid search to minimize the expected zero–one loss $E[L(Y, f(X))] = E[1 - \delta(Y, f(X))]$ (or miss-classification rate) on the available training set, where Y is the true label of the input X, $f(X)$ is the decision obtained from the classifier, and the $\delta(.,.)$ is the Kronecker delta function. Accordingly, since there is a training phase to optimize m and ρ directly on the task's objective function, the MVEM is expected to generalize well on unseen data sets.

4 Experimental results

To assess the validity and generalization of the modified MiniVenn algorithm, we have conducted extensive experiments in the context of classification on a large variety of standard benchmark data sets. In our experiments, fifteen data sets were used from the UCI Machine Learning Repository [15], shown in Table 1, with various sizes, number of classes and dimensionalities. Since there is no explicit training and test sets for these data sets, 10 Folds Double Cross Validation (FDCV) were used to report the errors on all the data sets. For query–based learning, a k–Nearest Neighbour (k–NN) classifier was used with three different values for $k = (1, 3, 5)$. Only raw data was used in the experiments without any kind of preprocessing. Similar to all local learning algorithms [19, 20, 17, 14], the parameter m plays a key role in the MVEM's performance. In all our experiments, the parameter m was set relative to the data dimensionality (d); i.e. $d + 1 \leq m \leq 3d$. The optimal value of m for each k–NN classifier using the MVEM, denoted by (m_{opt}), is shown Table (1).

In terms of comparisons, the k–NN classifier using the MVEM was compared with k–NN classifiers using three different metrics[1]: Euclidean (EUC), large margin nearest neighbour (LMNN) [22] and relevant component analysis (RCA) [2]. Since RCA depends on the availability and the amount of side–information (true labels in the context of classification), all the true labels for a given training set were provided to RCA in order to peel off any doubts about its performance.

Results analysis : Figures 1(a), 1(b) and 1(c) show the error rate and the difference in error for the three classifiers, $k = 1$, $k = 3$ and $k = 5$ respectively, using the four metrics on all data sets (Please see caption of Figure 1(a) for details). It can be seen clearly that for most of the cases, the MVEM is placed first with a large margin in error difference, or placed second with a very small margin in error difference against the competing metric (except for the wine case, see captions of Figure 1(e)). The MVEM is consistently better than the Euclidean metric and very competitive with a more dedicated algorithm like LMNN which was specifically designed to learn a metric that minimizes the error of k–NN classification (i.e. discriminative training). Similar behaviour is observed when comparing between MVEM and RCA. Although, as shown in Figure 1(d), that the MVEM is slightly better as an overall performance, statistical significance tests showed that on average, the MVEM is not significantly different than the more dedicated algorithms LMNN and RCA. This is a very interesting result since MiniVenn had less *a priori* information during training and yet it showed similar performance.

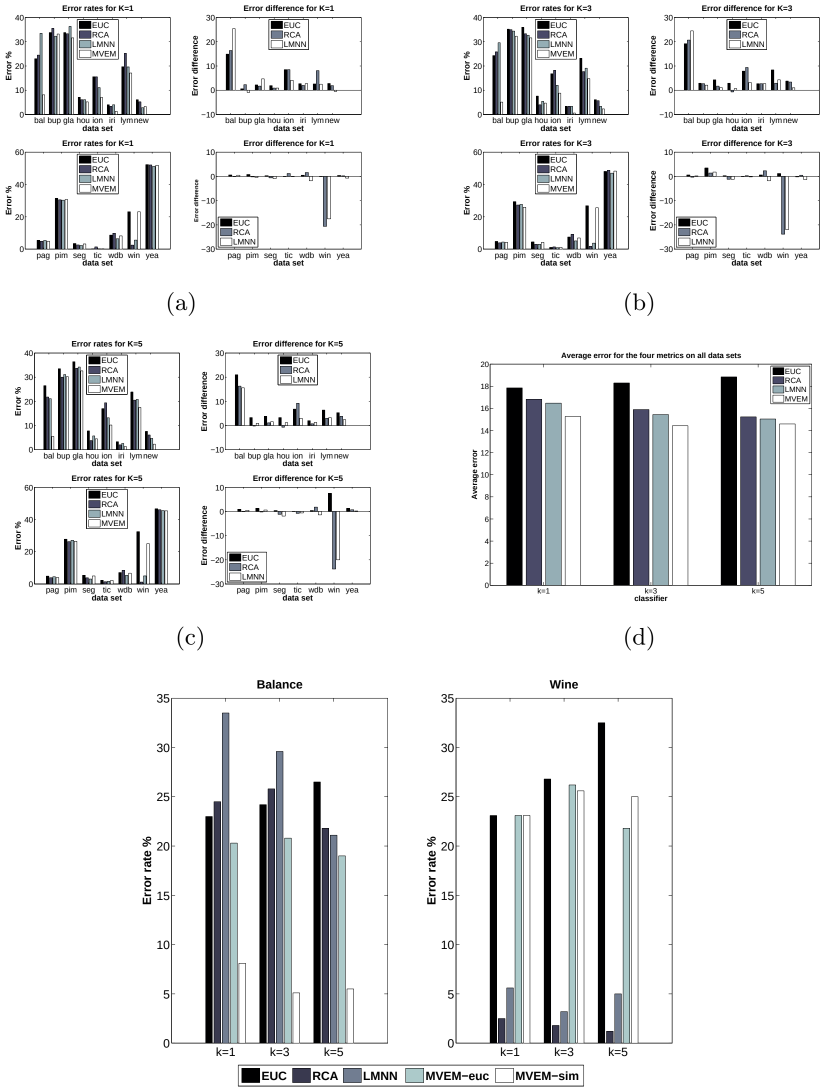
5 Conclusion

We have introduced an algorithm for learning an adaptive local metric for query–based operations. The algorithm combines ideas from the minimum volume ellipsoid metric and manifold learning algorithms to define the local metric on the lower dimensional manifold of the query point. In the context of classification, the metric showed promising results and found to be competitive with other metric learning algorithms in the literature. These results motivate us to extend the proposed algorithm and metric to the domain of clustering and unsupervised learning with complete absence of side–information and labels.

[1] The source code for LMNN and RCA was downloaded from the author's website.

References

1. K. Abou-Moustafa and F. Ferrie. The minimum volume ellipsoid metric. In *LNCS 4713, 29th DAGM Symposium, Heidelberg*, pages 335–344. Springer, 2007.
2. A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *JMLR*, 6:937–965, 2005.
3. L. Bottou and V. Vapnik. Local learning algorithms. *Neural Computation*, 4(6):888–900, 1992.
4. S. Boyd and L. Vandenberghe, editors. *Convex Optimization*. Cambridge Univ. Press, 2004.
5. C. Ding and T. Li. Adaptive dimension reduction using discriminant analysis and k–means clustering. In *Proceedings of the 24th ICML, Corvallis, OR*. 2007.
6. A. Dolia, T. De Bie, C. Harris, J. Shawe-Taylor, and D. Titterington. The minimum volume covering ellipsoid estimation in kernel-defined feature spaces. In *Proceedings of the 17th ECML, Berlin, September*. Springer, 2006.
7. J. Friedman. Regularized discriminant analysis. *J. of the American Statistical Assoc.*, 84(405):165–175, 1989.
8. K. Fukunaga and R. Olsen. An algorithm for finding intrinsic dimensionality of data. *IEEE Trans. on Computers*, 20(2):176–183, 1971.
9. A. Globerson, S. Roweis, G. Hinton, and R. Salakhutdinov. Metric learning by collapsing classes. In NIPS *18*, pages 451–458. MIT Press, 2006.
10. J. Goldberg and S. Roweis. Neighbourhood component analysis. In NIPS *17*, pages 513–520. MIT Press, 2005.
11. M. Grant, S. Boyd, and Y. Yinyu. Matlab software for disciplined convex programming, 2005. `http://www.stanford.edu/˜boyd/cvx`.
12. T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbour classification. *IEEE Trans. PAMI*, 18(6):607–615, 1996.
13. G. Lebanon. Metric learning for text documents. *IEEE Trans. PAMI*, 28(4):497–508, 2006.
14. T. Lin and H. Zha. Riemannian manifold learning. *IEEE. Trans. PAMI*, 30(5):796–809, 2008.
15. D. Newman, S. Hettich, C. Blake, and C. Merz. UCI repository of machine learning databases, 1998.
16. P. Rousseeuw. Multivariate estimation with high breakdown point. *Proc. of the 4th Pannonian Symp. on Mathematical Statistics*, 3:283–297, 1983.
17. S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding (lle). *Science*, 290(5500):2323–2326, 2000.
18. M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. In NIPS *16*. MIT Press, 2004.
19. R. Short and K. Fukunaga. The optimal distance measure for nearest neighbour classification. *IEEE Trans. on Information Theory*, 27(5):622–627, 1981.
20. J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, November 2000.
21. D. Titterington. Estimation of correleation coefficients by ellipsoidal trimming. *J. of Royal Statistical Society*, 27(3):227–234, 1978.
22. K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In NIPS *18*, pages 1473–1480. MIT Press, 2006.
23. E. Xing, A. Ng, M. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In NIPS *15*, pages 505–512. 2003.
24. L. Yang. Distance metric learning: A comprehensive review. Technical report, Dept. of Computer Science and Engineering, Michigan State University, 2006.



(a) (b) (c) (d) (e)

Fig. 1. (a) *Left* The error rates for k–NN classification ($k = 1$) using the four metrics: EUC, RCA [2], LMNN [22] and MVEM on the 15 UCI data sets. *Right* Difference in error for EUC, RCA and LMNN against the MVEM. Positive value implies the MVEM is better and a negative value implies the other metric is better. Ex.: Error(EUC) − Error(MVEM). (b) Error rates and error differences for $k = 3$. (c) Error rates and error differences for $k = 5$. (d) Average error for the three classifiers using the four metrics on all data sets. (e) The error rate using the Five metrics: EUC, RCA, LMNN, MVEM–euc (using Euclidean distance to define neighbourhoods) and MVEM–sim (using the similarity measure in Section 3.1). Note that the *wine* case reflects the power of the true labels exploited by RCA and LMNN. This is not observed in the *balance* case where the EUC error is significantly better than RCA and LMNN. The *balance* case also reflects the impact of well defined neighborhoods of a query point for the MVEM.