Large Margin Pursuit for a Conic Section Classifier

Santhosh Kodipaka, Arunava Banerjee, Baba C. Vemuri *
Department of Computer & Information Science & Engineering
University of Florida, Gainesville, FL 32611
{snsk,arunava,vemuri}@cise.ufl.edu

Abstract

Learning a discriminant becomes substantially more difficult when the datasets are high-dimensional and the available samples are few. This is often the case in computer vision and medical diagnosis applications. A novel Conic Section classifier (CSC) was recently introduced in the literature to handle such datasets, wherein each class was represented by a conic section parameterized by its focus, directrix and eccentricity. The discriminant boundary was the locus of all points that are equi-eccentric relative to each class-representative conic section. Simpler boundaries were preferred for the sake of generalizability.

In this paper, we improve the performance of the two-class classifier via a large margin pursuit. When formulated as a non-linear optimization problem, the margin computation is demonstrated to be hard, especially due to the high dimensionality of the data. Instead, we present a geometric algorithm to compute the distance of a point to the non-linear discriminant boundary generated by the CSC in the input space. We then introduce a large margin pursuit in the learning phase so as to enhance the generalization capacity of the classifier. We validate the algorithm on real datasets and show favorable classification rates in comparison to many existing state-of-the-art binary classifiers as well as the CSC without margin pursuit.

1. Introduction

The task of supervised learning becomes remarkably difficult when the number of training samples available is far fewer than the number of features used to represent each sample. We encounter such high dimensional sparse datasets in several computer vision and learning applications, such as in the diagnosis of Epilepsy based on brain MRI scans [7], the diagnosis of various types of Cancer from micro-array gene expression data [1], and speech recognition. Stated formally, the supervised learning problem is severely under constrained when one is given N labeled data points that lie in $\mathbb{R}^M$ where $N \ll M$.

A novel Conic Section classifier (CSC) was recently introduced [3] in the literature, to cater to such learning problem instances. The classifier represents each class by a conic section in the input space, parameterized by its focus, directrix and eccentricity. For data from two-classes lying in $\mathbb{R}^M$, the highly non-linear discriminant boundary generated by the CSC was shown to be fully specifiable using merely $4(M+1)$ parameters. In contrast, the kernel based SVM [11] and the Kernel Fisher Discriminant (KFD) [8] have $O(NM)$ parameters. The classifier assigns labels to input data points based on their distances in the space of eccentricities (*ecc-Space*), to their respective class eccentric descriptors. The performance of the classifier was demonstrated to be comparable to most state-of-the-art classifiers.

1.1. Motivation

The Conic Section classifier pursued simpler discriminants in the learning phase to improve upon its generalization capacity. It was found that as the discriminant boundary became non-linear, the empirical error did reduce. However, this came at the expense of generalizability unless a large margin was achieved. In order to improve upon the generalizability of the CSC, in this paper we pursue a large margin [11] between data points and the discriminant boundary, in the input space. We describe a geometric method to compute the distance of a point (and, as a result, the margin) to the non-linear discriminant boundary generated by the CSC in the input space. The learning algorithm for the CSC is then modified to pursue a large margin.

As is well known, the expected (testing) error of a classifier is bounded from above by the sum of the expected empirical (training) error and a generalization term that depends on its VC dimension [11]. The discriminant boundaries generated by the CSC belong to the family of polynomials and thus have finite VC dimension [11]. Upon introducing the margin constraints, it can be shown that the VC dimension of the CSC becomes quasi-independent of the dimensionality of the input space, as in the case of SVMs [11].

*This research was in part supported by NIH RO1 NS046812 to BCV.

The remainder of the paper is structured as follows. The Conic Section *concept class* is reviewed in Section 2. The learning algorithm involving incremental updates to these conic section descriptors for each class so as to better classify the data, is briefly discussed in Section 2.1 (we refer the reader to [3] for a more in-depth description). A novel algorithm to compute margin to the resultant discriminant boundary is presented in Section 3. This method is used to pursue a large margin discriminant in the input space. Validation results for margin computation are discussed in Section 4. Classification accuracy comparisons from several classifiers are reported for various datasets in Section 5.

2. The Conic Section Classifier

A conic section is defined as the locus of points for which the ratio of the distance to a fixed point (the *focus*) to that to a fixed hyperplane (the *directrix*), is a constant e (the *eccentricity*). When e is set to a value $< 1, = 1$ and > 1, we obtain an ellipse, a parabola and a hyperbola, respectively. The *concept class* for a two-class problem in $\mathbb{R}^M$ is defined as follows. Each member class k for $k = 1, 2$ is associated with a conic section specified by the set of descriptors, $C_k = \{F_k, \{b_k, Q_k\}, e_k\}$, where F_k is the focus, $\{b_k, Q_k\}$ specifies the directrix hyperplane of co-dimension 1 (Q_k is its unit normal and b_k is its offset from the origin), and e_k is the class eccentricity. The *focus* and *directrix* descriptors of each class attribute an eccentricity value to each point X in the input space, as

$$\varepsilon_k(X) = \frac{\|X - F_k\|}{b_k + Q_k^T X}; \text{ where, } \|Q_k\| = 1 \qquad (1)$$

$\|.\|$ denotes the Euclidean $\mathbb{L}_2$ norm. The class descriptors map X into the eccentricity space (*ecc-Space*) as $\xi(X) = \langle \varepsilon_1(X), \varepsilon_2(X) \rangle \in \mathbb{R}^2$. The distance of X from conic section C_k is defined as $dist(X, C_k) = |\varepsilon_k(X) - e_k|$. The classifier assigns X to the class whose conic section is closest to X. The resultant discriminant boundary, $\mathcal{G}$, is the locus of points such that $\{g(X) = 0\}$, where

$$g(X) = |\varepsilon_1(X) - e_1| - |\varepsilon_2(X) - e_2| \qquad (2)$$

With merely $4(M + 1)$ parameters, the classifier can represent a pair of polynomial surfaces whose degree is at most eight. Example boundaries in $\mathbb{R}^3$ are illustrated in Fig. 1.

2.1. The Learning Algorithm

In this section, we briefly describe the incremental Conic Section learning algorithm presented in [3], for the two-class classification problem. Given a set of N labeled samples $\mathcal{Z} = \{\langle X_1, y_1 \rangle, \ldots, \langle X_N, y_N \rangle\}$, where $X_i \in \mathbb{R}^M$ and labels $y_i \in \{-1, +1\}$, the objective is to learn the conic section descriptors, C_1, C_2, that simultaneously minimize the misclassification error and seek a simpler discriminant boundary. The data is assumed to be sparse in a very high dimensional input space, i.e., $N \ll M$.

The process of learning the noted descriptors has two principal stages, played out in the input space $\mathbb{R}^M$ and the *ecc-Space*, respectively. The relationship between these two spaces is given by the map ξ, which depends on the conic section descriptors for each class. In the first stage, given C_1 and C_2, each X_i is mapped into *ecc-Space* as $\langle \varepsilon_1(X_i), \varepsilon_2(X_i) \rangle$. These values are sufficient to compute a pair of class eccentricities $\langle e_1, e_2 \rangle$ that improves the empirical learning accuracy. For each misclassified sample, one can then find a desired pair of attributed eccentricities, denoted as $\langle \varepsilon'_{1i}, \varepsilon'_{2i} \rangle$, that would correctly classify it.

In the second stage, the *focus* and the *directrix* descriptors are alternately modified via a geometric algorithm to learn the map ξ. The map ξ is constrained to achieve the desired attributed eccentricities for the misclassified samples, *without affecting the eccentricities of those samples that are already classified correctly*. The process is repeated until the descriptors converge or there can be no further improvement in classification. To exercise a tight control on learning, the descriptors are updated to learn only one misclassified point at a time. Each misclassified point results in a descriptor update. It is here, that we can improve the generalization capacity of the classifier by picking the update that gives the largest margin.

The learning rate is non-decreasing since the algorithm does not misclassify a previously correctly classified point. A notable feature of the technique is its ability to track the entire feasible set for a descriptor, labeled as *Null Space*, that would map the data points to fixed attributed eccentricities, using simple geometric constructs.

3. The Margin Computation

The generalizability of a classifier can be improved by pursuing larger margins, as discussed in Section 1.1. We define the margin as the minimum over the distances of all the correctly classified data points, to the discriminant boundary. In order to compute the margin, we first have to find the shortest distance of a point, say $P \in \mathbb{R}^M$, to the discriminant boundary $\mathcal{G}$ (Eqn.2). When formulated as an optimization problem (Eqn.3) with the non-linear constraint (Eqn.2), the distance computation is NP-hard in general.

$$\text{dist}(P, \mathcal{G}) = \min \|P - X\|^2, \quad \text{subject to } g(X) = 0 \quad (3)$$

Although the objective function is quadratic in X, the equality constraint is highly non-linear. The numerical solution to Eqn.3 therefore works out to be very expensive. Note that we have to compute the shortest distance from the boundary to all the data points to determine the margin. For each competing descriptor update, we compute its margin

so as to pick an update that yields the largest margin. This task becomes especially difficult due to two reasons. First, the discriminant boundary is a pair of polynomial surfaces of degree at most eight. Second, the data lie in very high dimensional space. We now introduce a novel geometric approach to compute the margin.

3.1. Overview

The margin computation problem can be posed as finding the smallest hypersphere $\mathcal{S}(P,r)$, centered at P with radius r, that intersects with the discriminant boundary $\mathcal{G}$. Assuming we can evaluate the existence/lack of $\mathcal{G}\cap\mathcal{S}(P,r)$, the smallest radius is computed by performing a binary search on $r \in (0, r_p)$. The initial radius r_p is obtained by finding a point Z_0 lying on the discriminant boundary $\mathcal{G}$. Following is a method to find such a point. From the given labeled data set, pick a pair of data points, one from each side of the boundary, (see Eqn.2). Existence of such a point pair is guaranteed by our initial discriminant boundary. A binary search on the line joining this pair of data points gives us a point $Z_0 \in \mathcal{G}$ and hence an upper bound, say r_p, on the shortest distance of a given point P to $\mathcal{G}$.

Consider now sections of the boundary $\mathcal{G}$ for which the distances to either both the directrix planes or both the focal points are held fixed. As a consequence of certain geometric observations, we shall demonstrate that the existence/lack of intersection of the hypersphere $\mathcal{S}$ with any such section of the discriminant boundary can be determined analytically. The shortest distance of a point to these sections of the discriminant boundary can be computed in $O(NM)$ time. We propose an iterative algorithm that alternately updates the shortest distances to these sections so as to find a point on the discriminant boundary, nearest to P. An overview of the technique is presented in Algorithm-1. Next, we present algorithms to evaluate $\mathcal{G} \cap \mathcal{S}(P,r)$.

3.2. Spanning the Discriminant Boundary $\mathcal{G}$

The discriminant boundary (Eqn.2) can also be written as the implicit function:

$$((r_1 - e_1 h_1)h_2) \pm ((r_2 - e_2 h_2)h_1) = 0, \quad (4)$$

$$r_k(X) = \|X - F_k\|, \ \forall k \in 1,2 \quad (5)$$

$$h_k(X) = X^T Q_k + b_k, \ \forall k \in 1,2 \quad (6)$$

where r_k, and h_k are distances to the focus and directrix descriptors respectively. In order to evaluate the intersection, $\mathcal{G} \cap \mathcal{S}(P,r)$, we need to ascertain if $g(X)$ changes sign on the hypersphere, $\mathcal{S}$. The distances r_k, and h_k are bounded now, since $X \in \mathcal{S}(P,r)$. As a first pass, we can search for such a change in the sign of $g(X)$ by evaluating $g(X)$ at discretized distances in a bounded interval, for each valid combination of $\{h_1, h_2, r_1, r_2\}$.

For any point $X \in \mathcal{S}$, the distance $h_k(X)$ is bounded to be within $[h_k(P) - r_p, \ h_k(P) + r_p]$ due to Eqn.6. Similarly, r_k (Eqn.5) is bounded to be within $[|r_k(P) - r_p|, \ r_k(P) + r_p]$. If we discretize each parameter at $O(N)$ locations, the cost of evaluating an intersection is $O(N^4 M)$, which is expensive. In subsequent sections, we introduce a faster $O(M)$ algorithm to compute the intersection of the hypersphere $\mathcal{S}$ with particular sections of the boundary $\mathcal{G}$.

3.3. Finding Nearest Point for fixed $\{h_1(X), h_2(X)\}$

Here, we first determine if the hypersphere $\mathcal{S}$ intersects with the section of the boundary for which the distances to the *directrices* are fixed. Consider a subspace $\mathcal{H}$ in which all the points are at some fixed distances (h_1, h_2) from the two directrices $\{b_k, Q_k\}$ for k=1,2. Such a space $\mathcal{H}$ turns out to be a linear subspace of co-dimension 2 in $\mathbb{R}^M$, *i.e.* the intersection of two hyperplanes. With these constraints, the boundary $\mathcal{G}$ (Eqn.2) in $\mathcal{H}$ can also be written as:

$$\|X - F_2\| = m\|X - F_1\| + c \quad (7)$$

$$where, \ m = \pm\frac{h_2}{h_1}, \quad c = h_2(e_2 \mp e_1),$$

Since $\mathcal{H} \equiv \mathbb{R}^{M-2}$, let us track the section of the discriminant boundary, $\mathcal{G}\cap\mathcal{H}$. Let $X \in \mathcal{H}$ and the distance between a focus point F_k and its orthogonal projection F'_k in $\mathcal{H}$, be $s_k = \|F_k - F'_k\|$. Given (h_1, h_2), the section of the discriminant boundary, $\mathcal{G} \cap \mathcal{H}$ then becomes

$$\sqrt{\|X - F'_2\|^2 + s_2^2} = m\sqrt{\|X - F'_1\|^2 + s_1^2} + c \quad (8)$$

Fig. 1(c) illustrates the points $\{X, F'_1, F'_2\} \in \mathcal{H}$, related to Eqn.8 for $\mathcal{G} \cap \mathcal{H}$. Any point $P \in \mathbb{R}^M$ can be orthogonally projected into $\mathcal{H}$ with the equations below. The coefficients (u, v) in Eqn.9 are obtained by solving the constraints for $\mathcal{H}$ given in Eqn.10.

$$P' = P - (uQ_1 + vQ_2), \quad P' \in \mathcal{H} \quad (9)$$

$$with, \ Q_k^T P' + b_k = h_k, \quad \forall k \in 1,2 \quad (10)$$

Thus, $\mathcal{G}\cap\mathcal{H}$ is fully specified by elements lying in $\mathcal{H}$. We are now interested in finding out if the hypersphere $\mathcal{S}(P,r)$ intersects with $\mathcal{G} \cap \mathcal{H}$. Further, $\mathcal{S} \cap \mathcal{H}$ can be represented as a hypersphere $\mathcal{S}' \in \mathcal{F}$ centered at the projected data point $P' \in \mathcal{H}$ with radius r', derived from Eqn. 9. *Owing to symmetry, the intersection between the section of the boundary, $\mathcal{G} \cap \mathcal{H}$, and the hypersphere $\mathcal{S}'$ needs to be checked only in the plane comprising $\{P', F'_1, F'_2\}$.*

We can deduce from Eqn. 8 that the discriminant boundary in $\mathcal{H}$ is radially symmetric about the line joining F'_1 and F'_2. Let α be the length of the component of X along $(F'_2 - F'_1)$ and β be the length of its orthogonal component in the plane defined by $\{P', F'_1, F'_2\}$, as illustrated in Fig. 1(d).

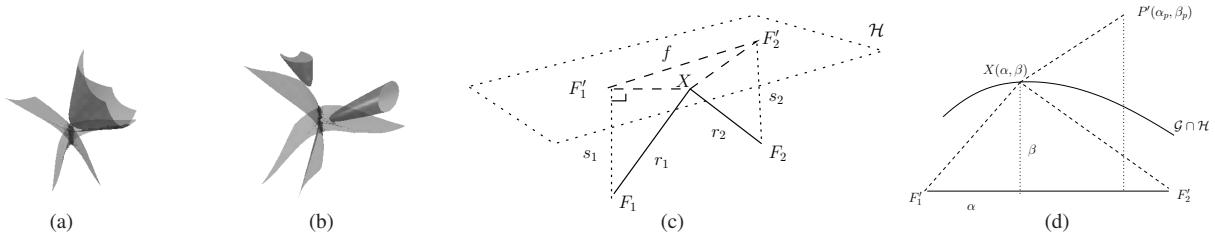

Figure 1. (a,b) Different examples of discriminant boundaries in $\mathbb{R}^3$. (c) $\{X, F_1', F_2'\} \in \mathcal{H}$, the linear subspace in which distances to directrices are constant (see Eqn.8). (d) The discriminant boundary in $\mathcal{H}$ becomes a quartic in (α, β) (see Eqn.11).

Let $f = \|F_2' - F_1'\|$. After translating the projected origin, $O' \in \mathcal{H}$ to F_1', the boundary becomes a quartic polynomial in the two variables (α, β), as in Eqn.11. Also, the equation of the hypersphere $\mathcal{S}'(P', r')$ reduces to the circle, Eqn.12. Here, (α_p, β_p) are the components of P' along and across the line joining F_2' and F_1'.

$$\sqrt{(\alpha - f)^2 + \beta^2 + s_2^2} = m\sqrt{\alpha^2 + \beta^2 + s_1^2} + c \quad (11)$$

$$(\alpha - \alpha_p)^2 + (\beta - \beta_p)^2 = (r')^2 \quad (12)$$

Upon intersecting the two geometric objects due to Eqns. 11, 12, we obtain a quartic equation in α after eliminating β. For any quartic polynomial in one variable, we can check for the existence of real roots [4] and compute them explicitly [10], if necessary. Thus, we determine the intersection, $\mathcal{S} \cap (\mathcal{G} \cap \mathcal{H})$, in $O(M)$ time.

Assume that we begin with a hypersphere $\mathcal{S}$ having an initial radius r_o that is guaranteed to intersect with the section of the discriminant boundary in $\mathcal{H}$, i.e $\mathcal{G} \cap \mathcal{H}$. All that remains to be done is to conduct a binary search on the radius of $\mathcal{S}$ in the interval $(0, r_o]$ to find the shortest distance between P and the discriminant surface in $\mathcal{H}$. Moreover, we can explicitly determine the nearest point on the section $\mathcal{G} \cap \mathcal{H}$, say Z, from the polynomial roots (α, β), and the points $\{P', F_1', F_2'\}$.

3.4. Finding Nearest Point for fixed $\{r_1(X), r_2(X)\}$

In this section, we determine if the hypersphere $\mathcal{S}(P, r)$ intersects with the part of the discriminant boundary in which distances to the *foci* are constant (Eqn.4). The locus of all the points that are at fixed distances $\{r_1, r_2, r\}$ from the points $\{F_1, F_2, P\}$ respectively, can be constructed by computing the intersection:

$$\mathcal{S}(P, r) \cap \mathcal{S}(F_1, r_1) \cap \mathcal{S}(F_2, r_2) \equiv \mathcal{S}'(C, r') \in \mathcal{F} \quad (13)$$

where $\mathcal{S}'$ is a hypersphere in the linear subspace $\mathcal{F}$ of co-dimension 2 in $\mathbb{R}^M$. This can be understood from an analogue in $\mathbb{R}^3$: *The intersection of two spheres in* $\mathbb{R}^3$ *is a circle lying within the plane of intersection.* An $O(K^2M)$ algorithm was presented in [3] to compute the intersection of K hyperspheres. We compute the intersection of three spheres here, hence $K = 3$. The problem is now reduced to determining the intersection $\mathcal{G} \cap \mathcal{S}'$. After translating the origin to $C \in \mathcal{F}$, any point X in $\mathbb{R}^M$ can be projected into $\mathcal{F}$ as:

$$X' = X - (X^T U_1)U_1 - (X^T U_2)U_2 \quad (14)$$

where, $\{U_1, U_2\}$ are two orthonormal vectors perpendicular to $\mathcal{F}$. The section of the discriminant boundary $\mathcal{G}$ in $\mathcal{F}$ for a given pair of fixed focal distances (r_1, r_2), now becomes:

$$\left(\frac{r_1}{h_1(X')} - e_1\right) \pm \left(\frac{r_2}{h_2(X')} - e_2\right) = 0 \quad (15)$$

which upon re-arrangement of terms results in a quadratic implicit surface equation in X'. Eqn.15 represents $\mathcal{G} \cap \mathcal{F}$ lying in $\mathcal{F}$. The intersection of the directrix hyperplane, $X'^T Q_k + b_k = 0$, with $\mathcal{F}$ can be equivalently represented as $X^T Q_k' + b_k = 0$. Here, $X \in \mathcal{F}$ and Q_k' is the component of unit normal Q_k in $\mathcal{F}$, obtained from Eqn.14. Now $\mathcal{G} \cap \mathcal{F}$ is fully specified by elements lying in $\mathcal{F}$.

We make a crucial geometric observation here. *Within* $\mathcal{F}$*, the discriminant function (Eqn.15) is invariant to translations normal to the plane spanned by* $\{Q_1', Q_2'\}$. We now exploit the fact that $\mathcal{G} \cap \mathcal{F}$ is a function only of the distances to the directrix planes in the linear subspace $\mathcal{F}$. Since $\mathcal{S}'(C, r')$ is a hypersphere, *it is sufficient to investigate the intersection of interest,* i.e. $\mathcal{S}' \cap (\mathcal{G} \cap \mathcal{F})$*, in the plane spanned by* $\{Q_1', Q_2'\}$ *and passing through* C*, the center of the hypersphere* S'. Any point X in such a plane can be expressed as :

$$X = C + \alpha Q_1' + \beta Q_2' \quad (16)$$

The section of the boundary $\mathcal{G} \cap \mathcal{F}$ (Eqn.15), in this plane, reduces to a quadratic curve in parameters (α, β). The hypersphere $\mathcal{S}'(C, r')$ in this plane becomes a (quadratic) circle, $\|X(\alpha, \beta) - C\| = r'$, in parameters (α, β). Again, the intersection of these two geometric objects, obtained by eliminating β, yields a quartic polynomial in α. We can analytically find if real roots exist for a given quartic [4] and compute them exactly [10].

We described an $O(M)$ algorithm to find if the intersection, $\mathcal{S} \cap (\mathcal{G} \cap \mathcal{F})$, exists. The shortest distance of a point P to the section of the boundary $\mathcal{G}$, in which the focal distances are constant, is computed via a binary search on the

radius of the initial hypersphere $\mathcal{S}(P, r)$ within the interval $(0, r_0]$. At the end of the binary search, we also compute the nearest point on the section, say $Z'(\alpha, \beta)$, from Eqn.16.

3.5. Large Margin Pursuit

As summarized in Algorithm-1, we alternately find the shortest distance to sections of the discriminant boundary, $\mathcal{G}$, with distances to either the foci or directrices fixed. To begin, the fixed distances, (h_1, h_2), to the directrices are obtained from the initial point Z_0 on $\mathcal{G}$. We then compute the point Z, that is nearest to P in the section $\mathcal{G} \cap \mathcal{H}$. The fixed distances, (r_1, r_2), to the foci are determined by Z for finding the nearest point, Z' in that section of the boundary. In this fashion, the nearest point in each section is used to define the next section.

Data: Data points set $\mathcal{Z}$, Conic Descriptors C_1, C_2
Result: Margin between point set $\mathcal{Z}$ and boundary $\mathcal{G}$
Find a point Z_0 on $\mathcal{G}$
for *each Point* $P \in \mathcal{Z}$ **do**
  Initial Distance $m_i = \|P - Z_0\|$, $Z' = Z_0$
  repeat
    Find closest $Z \in \mathcal{G}$ for fixed $\{h_1(Z'), h_2(Z')\}$
    Find closest $Z' \in \mathcal{G}$ for fixed $\{r_1(Z), r_2(Z)\}$
  until Z *converges*
  Point Distance $m_i = \|P - Z\|$
end
margin $\leftarrow min\{m_i\}$

Algorithm 1: Margin computation

The alternating process is repeated until either the distance to the boundary converges or an $O(N)$ iteration limit is reached. The number of steps in all the binary searches is limited to $O(N)$. The complexity of computing the shortest distance of a point to the boundary $\mathcal{G}$ in this manner, is $O(N^2M)$. The margin for a set of at most N points and a given conic configuration $\{C_1, C_2\}$ is computed in $O(N^3M)$ time. Similar to the numerical optimization technique (Eqn.3), the margin computation could be prone to local minima. However, we observed that the computation times are several orders of magnitude smaller than those for techniques involving either the numerical optimization (Eqn.3) or the discretization approach (Section-3.2). In the learning phase, among a set of competing updates for a conic section descriptor, we pick the one resulting in the largest margin. We also avoid an update if it reduces the current margin without improving the learning accuracy.

4. Evaluating Margin Computation

The margin computation is employed in the learning algorithm to choose a CSC descriptor update that yields larger margin. In this section, we evaluate the accuracy of such a selection and compare margins obtained using our method with the true ones. We considered Colon Tumor data [1] that has 62 samples with 2000 features each, and projected it into $\mathbb{R}^5$ using a whitening transform so that its covariance matrix is identity. To compute the true margins (Section-3.2), we performed brute-force search for change in sign of $g(X)$ on $\mathcal{S}(P, r)$ so as to determine an intersection.

CSC Boundary Types	Selection Accuracy %	Margin Error	
		μ	σ
Linear only	100.0	-.00087	.001413
Linear + non-linear	96.40	.000011	.004877
Non-linear	78.39	.01706	.022756
Highly non-linear	73.96	.03889	.034711

Table 1. Given a pair of CSC descriptor sets, the accuracy of picking the one with larger margin is compared for varying boundary types. The last two columns list errors in margin computation.

We considered discriminant boundaries of successively higher complexity. In the initial configuration for the results in Table-1, the directrices are coincident, the line joining foci is parallel to the directrix normal, say Q, and the class eccentricities are both zeros. This ensures that the discriminant boundary is always linear [3]. Upon making the eccentricities unequal, the boundary turns into a pair of linear and non-linear surfaces. Once the line joining the foci is not parallel to Q, the boundaries become non-linear. In the last case, the directrices are not coincident, resulting in highly non-linear boundaries.

For each boundary type, 20 CSC descriptors were randomly generated and the errors in margin computation are listed in Table-1. It can be seen that our approximations are reliable for simpler boundaries. Given each possible pair of competing descriptors, we verify if our method picked the configuration with larger margins. We observed that for near-linear and simpler boundaries, we selected the desired update in more than 95% of the instances. Since the discriminant turns out to be a collection of non-linear surfaces in general, our method is prone to local minima.

5. Classification Results

We compared the classification accuracies of CSC with the large margin pursuit (CSC-M) to CSC [3], as well as other state-of-the-art classifiers such as linear and kernel SVMs [11], and Fisher [5],[8] discriminants. We used polynomial kernels for the kernel based classifiers. The degree of the polynomial was empirically determined. Unless otherwise noted, all the classification experiments were performed using a leave-one-out cross validation protocol. The discriminant for CSC-M was initialized with that generated by either the linear SVM [11] or LFD [5]. In the learning phase, the competing conic-section descriptor updates were allowed to increase the margin without losing training

Dataset	Samples	Features	CSC-M	CSC	LFD	Lin-SVM	KFD PLY	SVM PLY
Epilepsy	44	216	**93.18**	88.64	77.27	56.18	86.36 (1)	86.36 (6)
Isolet-BC	100	617	**92.00**	84.00	81.00	91.00	91.00 (2)	91.00 (1)
CNS	60	7129	70.00	**73.33**	51.67	68.33	65.00 (3)	68.33 (1)
Colon Tumor	62	2000	**87.10**	**87.10**	85.48	80.56	75.81 (1)	82.26 (2)
Leukemia	72	7129	97.22	**98.61**	97.22	97.22	**98.61** (1)	97.22 (1)
HWdigits35	400	649	96.00	96.25	85.25	95.75	**97.75** (3)	95.75 (1)

Table 2. Classification accuracies for the Conic Section Classifier with margin pursuit (CSC-M), CSC, (Linear & Kernel) Fisher Discriminants and SVMs with polynomial (PLY) kernels. The degree of polynomial kernel is in parentheses.

classification accuracy. The CSC introduced in [3], pursued simpler boundaries in the learning phase to enhance generalizability. The classification results for six high dimensional sparse datasets pertaining to applications in computer vision and medical diagnosis, are listed in Table-2.

The **Epilepsy** dataset [7] consists of 3D Histograms of displacement vector fields representing the non-rigid registration between the left and right hippocampi in 3D. The task is to distinguish between Left and Right Anterior Temporal Lobe (RATL) epileptics. The dataset includes features for 19 LATL and 25 RATL epileptic subjects. The feature vector length, which depended on the number of 3D bins used, was empirically determined. All the classifiers performed well with a $6 \times 6 \times 6$ binning of the displacement field, except KFD for which we used a $16 \times 16 \times 16$ binning. As listed in Table-2, CSC with margin pursuit (CSC-M) achieved an impressive 93% testing accuracy, clearly outperforming all the other classifiers.

The **Isolet-BC** dataset is a part of the Isolet Spoken Letter Recognition Database [2]. We considered 50 speech samples each, for alphabets B and C. CSC-M classifier outperformed all the other classifiers, with a large improvement over CSC without margin pursuit. The **HWdigits35** data is a part of the Hand-written Digits dataset [2]. It has 200 samples of each digit, and around 649 features for each sample. We classified the digits *three* and *five*, as they turned out to be a difficult pair. We report 10-fold cross-validation experiments for these two datasets. The next three datasets involve gene-expression features. The **CNS** dataset [9] contains treatment outcomes for central nervous system embryonal tumor on 60 patients, which includes 21 survivors and 39 failures. The **Colon Tumor** data [1] consists of 22 normal and 40 tumor colon tissue features. In the **Leukemia** cancer class discovery [6], the task is to discriminate between 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL) samples.

Overall, the performance of CSC-M improved significantly over that without the margin pursuit, for the *Epilepsy* and *Isolet-BC* datasets. CSC-M almost matched CSC in the remaining four experiments in Table-2, as both the methods are susceptible to achieving local optima w.r.t. learning the optimal descriptors. The performance of linear-SVM is comparable to that of CSC-M for Isolet-BC, HWdigits35 and Leukemia datasets, since the large margin discriminant can turn out to be near-linear.

6. Summary

In this paper, we proposed a novel geometric algorithm to compute margin to discriminant boundaries resulting from the Conic Section classifier. This technique was then used to maximize margin while learning conic section descriptors for each class, so as to improve upon the generalizability of the classifier. The classification performance of the classifier did improve on real datasets, due to enhancement of the learning method with a large margin pursuit.

References

[1] U. Alon et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *PNAS (96)*, 1999.

[2] A. Asuncion and D. Newman. UCI ML Repository, 2007.

[3] A. Banerjee, S. Kodipaka, and B. C. Vemuri. A conic section classifier and its application to image datasets. In *CVPR (1):103-108*, 2006.

[4] I. Z. Emiris and E. P. Tsigaridas. Comparing real algebraic numbers of small degree. In *ESA*, pages 652–663, 2004.

[5] R. Fisher. The use of multiple measurements in taxonomic problems. In *Annals of Eugenics (7)*, pages 111–132, 1936.

[6] T. Golub et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[7] S. Kodipaka et al. Kernel fisher discriminant for shape-based classification in epilepsy. *Medical Image Analysis*, 2006.

[8] S. Mika et al. Fisher discriminant analysis with kernels. *NNSP(IX):41–48*, 1999.

[9] S. L. Pomeroy et al. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870):436–442, 2002.

[10] S. Rao. *Mechanical vibrations*. Addison-Wesley, 4/ed, 2004.

[11] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, 1999.