



Large margin classifiers based on affine hulls

Hakan Cevikalp, Bill Triggs, Hasan Serhan Yavuz, Yalçın Küçük, Mahide Küçük, Atalay Barkana

► To cite this version:

Hakan Cevikalp, Bill Triggs, Hasan Serhan Yavuz, Yalçın Küçük, Mahide Küçük, et al.. Large margin classifiers based on affine hulls. *Neurocomputing*, 2010, 73 (16-18), pp.3160-3168. 10.1016/j.neucom.2010.06.018 . hal-00564987

HAL Id: hal-00564987

<https://hal.science/hal-00564987>

Submitted on 10 Feb 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

LARGE MARGIN CLASSIFIERS BASED ON AFFINE HULLS

Hakan Cevikalp^a, Bill Triggs^b, Hasan Serhan Yavuz^a, Yalcin Kucuk^c, Mahide Kucuk^c, Atalay Barkana^d

^a*Electrical and Electronics Engineering Department of Eskisehir Osmangazi University, Meselik
26480 Eskisehir, Turkey*

^b*Laboratoire Jean Kuntzmann, Grenoble, France*

^c*Mathematics Department of Anadolu University, Eskisehir, Turkey*

^d*Electrical and Electronics Engineering Department of Anadolu University, Eskisehir, Turkey*

Abstract

This paper introduces a geometrically inspired large-margin classifier that can be a better alternative to the Support Vector Machines (SVMs) for the classification problems with limited number of training samples. In contrast to the SVM classifier, we approximate classes with affine hulls of their class samples rather than convex hulls. For any pair of classes approximated with affine hulls, we introduce two solutions to find the best separating hyperplane between them. In the first proposed formulation, we compute the closest points on the affine hulls of classes and connect these two points with a line segment. The optimal separating hyperplane between the two classes is chosen to be the hyperplane that is orthogonal to the line segment and bisects the line. The second formulation is derived by modifying the ν -SVM formulation. Both formulations are extended to the nonlinear case by using the kernel trick. Based on our findings, we also develop a geometric

Email addresses: hakan.cevikalp@gmail.com (Hakan Cevikalp),
Bill.Triggs@imag.fr (Bill Triggs), hsyavuz@ogu.edu.tr (Hasan Serhan Yavuz),
ykucuk@anadolu.edu.tr (Yalcin Kucuk), mkucuk@anadolu.edu.tr (Mahide Kucuk),
atalaybarkan@anadolu.edu.tr (Atalay Barkana)

interpretation of the Least Squares SVM classifier and show that it is a special case of the proposed method. Multi-class classification problems are dealt with constructing and combining several binary classifiers as in SVM. The experiments on several databases show that the proposed methods work as good as the SVM classifier if not any better.

Keywords: Affine hull, classification, convex hull, kernel methods, large margin classifier, quadratic programming, support vector machines.

1. Introduction

The Support Vector Machine (SVM) classifier is a successful binary classification method that simultaneously minimizes the empirical classification error and maximizes the geometric margin, which is defined as the distance between the separating hyperplane and closest samples from the classes [2, 8]. To do so, SVM first approximates each class with a convex hull and finds the closest points in these convex hulls [1]. Then, these two points are connected with a line segment. The hyperplane, orthogonal to the line segment that bisects the line, is chosen to be the separating hyperplane. From the geometrical point of view, in the separable case, the two closest points on the convex hulls determine the separating hyperplane, and the SVM margin is merely equivalent to the minimum distance between the convex hulls that represent classes. However, convex hull approximations tend to be unrealistically tight in high-dimensional spaces since the classes typically extend beyond the convex hulls of their training samples. For example, a convex hull constructed by randomly sampled points from a high-dimensional hypersphere can include only a negligible fraction of the volume of the sphere even if the chosen samples are well spaced and close to the surface of the sphere

18 [3]. This situation may also be observed when the low-dimensional data samples
19 are mapped to a higher-dimensional feature space through kernel mapping during
20 estimation of the nonlinear decision boundaries between classes.

21 As opposed to the convex hulls, affine hulls (i.e., spanning linear subspaces
22 that have been shifted to pass through the centroids of the classes) give rather
23 loose approximations to the class regions, because they do not constrain the po-
24 sitions of the training points within the affine subspaces. Therefore, they may
25 be better alternatives to convex hulls for some pattern classification problems es-
26 pecially when the data samples lie in high-dimensional spaces. In the context
27 of classification, affine hulls are first used as global classifiers of isolated word
28 and hand-written digits giving good classification performance [11, 14]. In these
29 methods, each class is approximated with an affine hull constructed from its train-
30 ing samples, and the label of a test sample is determined based on the distance to
31 the nearest affine hull. Vincent and Bengio [26] used affine/convex hulls in a local
32 sense by constructing them using k-nearest neighbors of a test sample for classi-
33 fication problems with complex nonlinear decision boundaries. They report that
34 affine hulls usually give higher classification accuracy than convex hulls and that
35 using both models for classification significantly improves the k-nearest neighbor
36 classification performance [26]. We extended local linear affine/convex hull clas-
37 sifiers to the nonlinear case in [4]. More recently, we compared different convex
38 class models for high-dimensional classification problems, then found that affine
39 hull approximations are typically more accurate than convex hull approximations
40 [3]. These results are not surprising due to the fact that high-dimensional ap-
41 proximations tend to be simple: For a fixed sample size, the amount of geometric
42 details that can be resolved usually decreases rapidly as the dimensionality in-

43 creases. Therefore, affine hulls tend to be better models for high-dimensional data
44 approximations.

45 Besides the classification, approximations based on affine hulls have also been
46 used for dimensionality reduction. Mixtures of Principal Component Analyzers
47 [12] use local affine hulls to estimate nonlinear data manifolds. Similarly, Lo-
48 cally Linear Embedding [18] approximates the nonlinear structure of the high-
49 dimensional data by exploiting local affine hull reconstructions. Verbeek [25]
50 combined several locally valid affine hulls to obtain a global nonlinear map-
51 ping between the high-dimensional sample space and low-dimensional manifold.
52 Many applications of affine hulls in the context of classification and dimension-
53 ality reduction can be attributed in part to their simplicity and computational ef-
54 ficiency. Finding distances from test samples to affine hulls requires only simple
55 linear algebra. On the other hand, computing distances to nonlinear complex mod-
56 els can be problematic. Even if the models are restricted to being convex hulls,
57 distance computations require the solution of a quadratic optimization problem.

58 The classification methods using affine hulls or other convex sets described
59 above are “nearest convex model” classifiers and they are instance-based in na-
60 ture. In other words, decision boundaries are not explicitly created during a train-
61 ing phase. Instead, the decision boundaries remain implicit, and new examples
62 are classified online based on the distances to the nearest convex class models.
63 This paper investigates an alternative “margin between convex model” strategy
64 that is based on explicitly building maximum margin separators between pairs
65 of affine hulls. As a first example of the power of this approach, note that the
66 SVM itself is the maximum margin separator between the convex hulls of the
67 training samples of the two classes. One motivation for replacing nearest-convex-

68 model approaches with margin-based ones is that for all of the above cited nearest-
 69 convex-model classifiers, the decision boundaries (surfaces equidistant from the
 70 two convex models) are generically at least quadratic or piecewise quadratic in
 71 complexity. For example, for affine hulls they are generically hyperboloids. Such
 72 decision boundaries are more flexible than linear ones, but in high dimensions
 73 when the training data is scarce this may lead to overfitting, thus damaging gener-
 74 alization to unseen examples. Linear margin based approaches have fewer degrees
 75 of freedom, so they are typically less sensitive to the precise arrangement of the
 76 training samples. For example, for an SVM classifier, motions of the SVM support
 77 vectors parallel to the SVM decision surface do not alter the margin and hence do
 78 not invalidate the classifier, whereas they do typically change piece-wise quadratic
 79 decision surface of the equivalent nearest convex hull classifier. Another motiva-
 80 tion for studying margin-between-affine hulls approach is their potential flexibility
 81 and compactness. In linear case, affine models allow each class to be fitted indi-
 82 vidually and represented compactly, following which the linear separator between
 83 any two classes can be found quickly by simple linear algebra.

84 In our preliminary work [5] we showed how to construct maximum margin
 85 classifier that separates linear affine hulls. Another study addressing the same
 86 problem was independently given in [29]. Here we extend the method such that
 87 it can be used when the class samples lie on nonlinear manifolds that cannot be
 88 modeled with linear affine hulls. To this end, we map the samples in each class
 89 into a much higher-dimensional feature space through kernel mapping and then
 90 construct the linear affine hulls in the mapped space. Since the problem is cast
 91 in a much higher-dimensional space, it is more likely that class regions can now
 92 be approximated with linear affine hulls. Although the constructed linear affine

hulls in the mapped space corresponds to nonlinear manifolds in the input space, finding the maximum separating hyperplane between these nonlinear manifolds is still straightforward because of their linear nature in the mapped space. In case of outliers, to allow soft margin solutions, we first reduce affine hulls in order to alleviate the effects of those outliers and then search for the best separating hyperplane between these reduced robust models.

The rest of the paper is organized as follows: In Section II, we introduce the proposed method. Section III describes the experimental results. Concluding remarks are given in Section IV.

2. Method

Consider a binary classification problem with the training data given in the form $\{\mathbf{x}_i, y_i\}$, $i = 1, \dots, n$, $y_i \in \{-1, +1\}$, $\mathbf{x}_i \in \mathbb{R}^d$. To separate classes, SVM classifier finds a separating hyperplane that maximizes the margin, which is defined as the distance between the hyperplane and closest samples from the classes. To do so, SVM first approximates each class with a convex hull [1]. A convex hull consists of all points that can be written as a convex combination of the points in the original set, and a convex combination of points is a linear combination of data points where all coefficients are nonnegative and sum up to 1. More formally, the convex hull of samples $\{\mathbf{x}_i\}_{i=1, \dots, n}$ can be written as

$$H^{convex} = \left\{ \mathbf{x} = \sum_{i=1}^n \alpha_i \mathbf{x}_i \mid \sum_{i=1}^n \alpha_i = 1, \alpha_i \geq 0 \right\}. \quad (1)$$

Convex hulls of two classes are illustrated in Fig. 1. Following this approximation, SVM finds the closest points in these convex hulls. Then, these two points are

connected with a line segment. The plane, orthogonal to the line segment that bisects the line, is selected to be the separating hyperplane as shown in Fig. 1.

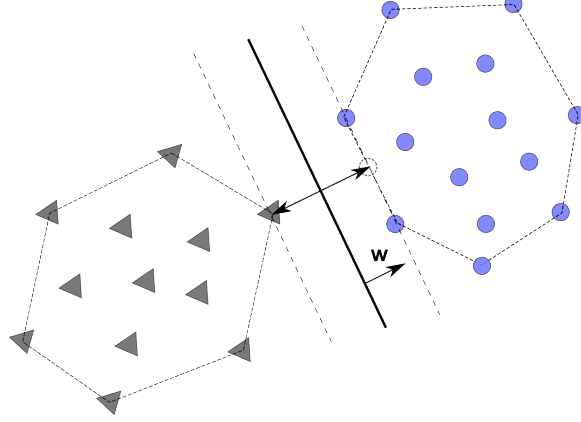


Figure 1: Two closest points on the convex hulls determine the separating hyperplane.

115

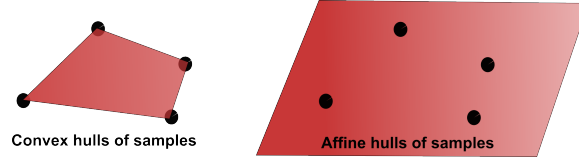


Figure 2: Comparison of convex and affine hulls of samples.

In contrast to the SVM classifier, the proposed method approximates each class (positive and negative classes) with an affine hull of its training samples. An affine hull of a class is the smallest affine subspace containing them. This is an unbounded, and hence typically rather loose model for each class, thus affine hull modeling can be a better choice than convex hull modeling for high-dimensional data. Affine and convex hulls of four samples are illustrated in Fig. 2. The affine hull of samples $\{\mathbf{x}_i\}_{i=1,\dots,n}$ contains all points of the form $\sum_{i=1}^n \alpha_i \mathbf{x}_i$

123 with $\sum_{i=1}^n \alpha_i = 1$. More formally affine hull of a class with samples $\{\mathbf{x}_i\}_{i=1,\dots,n}$
 124 can be written as

$$H^{aff} = \left\{ \mathbf{x} = \sum_{i=1}^n \alpha_i \mathbf{x}_i \mid \sum_{i=1}^n \alpha_i = 1 \right\}. \quad (2)$$

125 Our goal is to find the maximum margin linear separating hyperplane between
 126 affine hulls of classes. The points \mathbf{x} which lie on the separating hyperplane satisfy
 127 $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$, where \mathbf{w} is the normal of the separating hyperplane, $|b|/||\mathbf{w}||$
 128 is the perpendicular distance from the hyperplane to the origin, and $||\mathbf{w}||$ is the
 129 Euclidean norm of \mathbf{w} . For any separating hyperplane, all points \mathbf{x}_i in the posi-
 130 tive class satisfy $\langle \mathbf{w}, \mathbf{x}_i \rangle + b > 0$ and all points \mathbf{x}_i in the negative class satisfy
 131 $\langle \mathbf{w}, \mathbf{x}_i \rangle + b < 0$ so that $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) > 0$ for all training data points. Finding
 132 the best separating hyperplane between affine hulls can be solved by computing
 133 the closest points on them. The optimal separating hyperplane will be the one that
 134 bisects perpendicularly the line segment connecting the closest points as in SVM
 135 classifier. The offset (also called threshold), b , can be chosen as the distance from
 136 the origin to the point halfway between the closest points along the normal \mathbf{w} .
 137 Once the best separating hyperplane is determined, a new sample \mathbf{x} is classified
 138 based on the sign of the decision function, $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$.

139 Next, we will first show how to find the best separating hyperlane for linearly
 140 separable affine hulls and then extend the idea for inseparable case. After that, we
 141 explain kernelization process. This is followed by introducing a second equivalent
 142 formulation based on a variation of ν -SVM [20]. Lastly we show the relation
 143 between the proposed method and the Least Squares SVM (LS-SVM) [24, 23]
 144 and derive a geometric intuition for LS-SVM.

145 2.1. Linearly Separable Case

146 Suppose that affine hulls belonging to the positive and negative classes are
 147 linearly separable. The affine hulls of two classes do not intersect, i.e., they are
 148 linearly separable, if the affine combinations of their samples satisfy the rule

$$\sum_{i:y_i=+1} \alpha_i \mathbf{x}_i \neq \sum_{j:y_j=-1} \alpha_j \mathbf{x}_j \text{ for } \sum_{i:y_i=+1} \alpha_i = \sum_{j:y_j=-1} \alpha_j = 1. \quad (3)$$

149 It should be noted that linear separability of data points does not necessarily guar-
 150 antee the separability of corresponding affine hulls of classes. For linearly sepa-
 151 rable case, it is more convenient to write an affine hull as

$$\{\mathbf{x} = \mathbf{U}\mathbf{v} + \boldsymbol{\mu} \mid \mathbf{v} \in \mathbb{R}^l\}, \quad (4)$$

152 where $\boldsymbol{\mu} = (1/n) \sum_i \mathbf{x}_i$ is the mean of the samples (or any other reference
 153 point in the hull) and \mathbf{U} is an orthonormal basis for the directions spanned by
 154 the affine subspace. The vector \mathbf{v} contains the reduced coordinates of the point
 155 within the subspace, expressed with respect to the basis \mathbf{U} . Numerically, \mathbf{U} can
 156 be found as the U-matrix of the ‘thin’ Singular Value Decomposition (SVD) of
 157 $[\mathbf{x}_1 - \boldsymbol{\mu}, \dots, \mathbf{x}_n - \boldsymbol{\mu}]$. Here, ‘thin’ indicates that we take only the columns of \mathbf{U}
 158 corresponding to “significantly non-zero” singular values λ_k ; l is the number of
 159 such non-zero singular values. This subspace estimation process is essentially or-
 160 thogonal least squares fitting. Discarding near-zero singular values corresponds to
 161 discarding directions that appear to be predominantly “noise”. As an alternative,
 162 samples can be fitted with some other more robust subspace estimation processes
 163 such as L1 norm based subspace fitting procedures described in [10, 13]. But, we
 164 will consider only the least squares fitting (L2 norm) in this study.

Now suppose that we have two affine hulls with point sets $\{\mathbf{U}_+ \mathbf{v}_+ + \boldsymbol{\mu}_+\}$ and
 $\{\mathbf{U}_- \mathbf{v}_- + \boldsymbol{\mu}_-\}$. (These can be estimated with either L2 or L1 fitting and they may

have different numbers of dimensions l). A closest pair of points between the two hulls can be found by solving

$$\min_{\mathbf{v}_+, \mathbf{v}_-} \|(\mathbf{U}_+ \mathbf{v}_+ + \boldsymbol{\mu}_+) - (\mathbf{U}_- \mathbf{v}_- + \boldsymbol{\mu}_-)\|^2. \quad (5)$$

Defining $\mathbf{U} \equiv \begin{pmatrix} \mathbf{U}_+ & -\mathbf{U}_- \end{pmatrix}$ and $\mathbf{v} \equiv \begin{pmatrix} \mathbf{v}_+ \\ \mathbf{v}_- \end{pmatrix}$, this can be written as the standard least squares problem

$$\min_{\mathbf{v}} \|\mathbf{U} \mathbf{v} - (\boldsymbol{\mu}_- - \boldsymbol{\mu}_+)\|^2. \quad (6)$$

165 If we take the derivative of the objective function (6) with respect to \mathbf{v} and equate
166 it to zero, then we obtain

$$\mathbf{U}^\top \mathbf{U} \mathbf{v} - \mathbf{U}^\top (\boldsymbol{\mu}_- - \boldsymbol{\mu}_+) = \mathbf{0} \quad (7)$$

167 Subsequently, we get the solution of the problem as $\mathbf{v} = (\mathbf{U}^\top \mathbf{U})^{-1} \mathbf{U}^\top (\boldsymbol{\mu}_- - \boldsymbol{\mu}_+)$.

168 Taking the decision boundary $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$,

$$\mathbf{w} = \frac{1}{2}(\mathbf{x}_+ - \mathbf{x}_-) = \frac{1}{2}(\mathbf{I} - \mathbf{P})(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-) \quad (8)$$

169 where $\mathbf{P} = \mathbf{U} (\mathbf{U}^\top \mathbf{U})^{-1} \mathbf{U}^\top$ is the orthogonal projection onto the joint span of the
170 directions contained in the two subspaces, $\mathbf{I} - \mathbf{P}$ is the corresponding projection
171 onto the orthogonal complement of this span¹, and \mathbf{x}_+ and \mathbf{x}_- denote the closest
172 points on the positive and negative classes, respectively. Note that \mathbf{w} lies along
173 the line segment joining the two closest points and it is half the line segment's
174 size. The offset b of the separating hyperplane is given by

$$b = -\mathbf{w}^\top (\mathbf{x}_+ + \mathbf{x}_-)/2. \quad (9)$$

¹If the two subspaces share common directions, $\mathbf{U}^\top \mathbf{U}$ is not invertible and the solution for $(\mathbf{v}_+, \mathbf{v}_-)$ and $(\mathbf{x}_+, \mathbf{x}_-)$ is non-unique, but the orthogonal complement remains well defined, giving a unique minimum norm separator \mathbf{w} . Numerically all cases can be handled by finding $\tilde{\mathbf{U}}$, the \mathbf{U} matrix of the thin SVD of \mathbf{U} , and taking $\mathbf{P} = \tilde{\mathbf{U}}^\top \tilde{\mathbf{U}}$.

175 2.2. Inseparable Case

176 A problem arises if the affine hulls of classes intersect, i.e., affine hulls are not
 177 linearly separable. If the affine hulls of classes are close to being linearly separable
 178 and they overlap because of a few outliers, we can restrict the influence of outlying
 179 points by reducing affine hulls. Note that ignoring directions corresponding to the
 180 overly small singular values during affine hulls constructions reduces the effects
 181 of noise and outliers up to some degree. But, we will use a different approach
 182 here in order to cope with the outliers. To this end, we use the initial affine hull
 183 formulation (2) and introduce upper and lower bounds on coefficients α_i to reduce
 184 affine hulls inspired by the idea that is introduced to reduce convex hulls in [1]. It
 185 should be noted that the reduced affine hulls are not uniformly scaled versions of
 186 the initial complete affine hulls. One may go further and choose different lower
 187 and upper bounds, or define a different interval for every sample in the training
 188 set if a-priori information is available. For instance, if the lower bound is set
 189 to zero, then the method will be equivalent to the SVM classifier. Finding the
 190 closest points on the reduced affine hulls can be written as a quadratic optimization
 191 problem

$$\begin{aligned} \min_{\alpha} \quad & \left\| \sum_{i:y_i=+1} \alpha_i \mathbf{x}_i - \sum_{i:y_i=-1} \alpha_i \mathbf{x}_i \right\|^2 \\ \text{s.t.} \quad & \sum_{i:y_i=+1} \alpha_i = 1, \quad \sum_{i:y_i=-1} \alpha_i = 1, \quad -\tau \leq \alpha_i \leq \tau, \end{aligned} \quad (10)$$

192 where τ is the user-chosen bound. This optimization problem (10) can be written
 193 in a more compact form as

$$\begin{aligned} \min_{\alpha} \quad & \sum_{ij} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ \text{s.t.} \quad & \sum_i \alpha_i y_i = 0, \quad \sum_i \alpha_i = 2, \quad -\tau \leq \alpha_i \leq \tau. \end{aligned} \quad (11)$$

194 This is a quadratic programming problem that can be solved using standard opti-
 195 mization techniques. Note that the Hessian matrix, $\mathbf{G} = [G_{ij}] = y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$, is
 196 a positive semi-definite matrix, thus the objective function is convex and a global
 197 minimum exists as in SVM classifier. Moreover, if the Hessian matrix is strictly
 198 positive definite, the solution is unique and it is guaranteed to be the global mini-
 199 mum.

200 Since the coefficients are bounded between $-\tau$ and $+\tau$, the solution is de-
 201 termined by more points and no extreme point or noisy point can excessively
 202 influence the solution for well-chosen τ . Once we compute the optimal values
 203 of coefficients α_i , the normal and the offset of the separating hyperplane can be
 204 computed as in the linearly separable case

$$\mathbf{w} = \frac{1}{2} \left(\sum_{i:y_i=+1} \alpha_i \mathbf{x}_i - \sum_{i:y_i=-1} \alpha_i \mathbf{x}_i \right), \quad (12)$$

$$b = -\frac{1}{2} \mathbf{w}^\top \left(\sum_{i:y_i=+1} \alpha_i \mathbf{x}_i + \sum_{i:y_i=-1} \alpha_i \mathbf{x}_i \right), \quad (13)$$

206 We call this method Large Margin Classifier of Affine Hulls (LMC-AH) since
 207 it uses affine hulls to approximate class regions and finds the optimal separating
 208 hyperplane yielding the largest margin between the affine hulls.

209 If the underlying geometry of the classes is highly complex and nonlinear, and
 210 approximating classes with linear affine hulls is not appropriate, we can map the
 211 data into a higher-dimensional space where the classes can be approximated with
 212 linear affine hulls. Note that the objective function of (11) is written in terms of
 213 the dot products of samples, which allows the use of the kernel trick. Thus, by
 214 using kernel trick, – i.e., replacing $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ with the kernel function $k(\mathbf{x}_i, \mathbf{x}_j) =$
 215 $\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ where $\phi : \mathbb{R}^d \rightarrow \mathfrak{S}$ is the mapping function from the input space
 216 to the mapped space \mathfrak{S} – we can find the best separating hyperplane parameters

217 in the mapped space. As a result, more complex nonlinear decision boundaries
 218 between classes can be approximated by using this trick.

219 2.3. An Equivalent Formulation Based on Variation of ν -SVM Classifier

220 The ν -SVM formulation has been proposed as an alternative to the classical
 221 SVM formulation [20]. A new parameter ρ' is introduced in this formulation, and
 222 error penalty term C that appears in classical SVM formulation is removed. Here,
 223 we introduce an alternative formulation to find the best separating hyperplane be-
 224 tween affine hulls based on a variation of ν -SVM formulation. A major advantage
 225 of new formulation is that one can relate the parameter τ in (11) with the ex-
 226 pected error bounds and this may help us to find a more sophisticated procedure
 227 for choosing unknown parameters that appear in both formulations.

228 The ν -SVM optimization is formulated as

$$\begin{aligned} \min_{\mathbf{w}', b', \xi'_i, \rho'} \quad & \frac{1}{2} \|\mathbf{w}'\|^2 - \nu \rho' + \frac{1}{n} \sum_i \xi'_i \\ \text{s.t.} \quad & y_i(\langle \mathbf{w}', \mathbf{x}_i \rangle + b') \geq \rho' - \xi'_i, \quad \xi'_i \geq 0, \quad \rho' \geq 0, \end{aligned} \quad (14)$$

229 where \mathbf{w}' represents the normal of the separating hyperplane, b' is the offset, ν
 230 is a user-chosen parameter between 0 and 1, and $\xi'_i, i = 1, \dots, n$, are the positive
 231 slack variables. In this formulation, for linearly separable case, there exist two
 232 parallel supporting hyperplanes positioned such that all points in the positive class
 233 satisfy $\langle \mathbf{w}', \mathbf{x} \rangle + b' \geq \rho'$ and all points in the negative class satisfy $\langle \mathbf{w}', \mathbf{x} \rangle +$
 234 $b' \leq -\rho'$ as shown in Fig. 3. Therefore, classes are separated by the margin
 235 $2\rho' / \|\mathbf{w}'\|$ and it is shown that ν acts as an upper bound on the fraction of margin
 236 errors and a lower bound on the fraction of support vectors [20]. Moreover, the
 237 decision function produced by ν -SVM can also be produced by classical SVM for
 238 appropriate choice of error penalty term C .

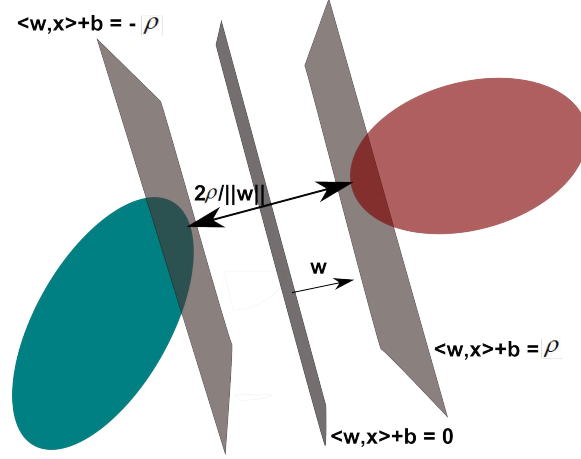


Figure 3: Illustration of the supporting and the best separating hyperplanes in linearly separable case for ν -SVM classifier.

239 The ν -SVM formulation can be interpreted as a maximal separation between
 240 the reduced convex hulls of classes [9]. Since we use affine hulls to model classes,
 241 we need to revise the optimization problem to accommodate this change. To this
 242 end, we first divide the objective function by $\nu^2/2$, the constraints by ν , and make
 243 the following substitutions as in [9]

$$\tau = \frac{2}{\nu n}, \quad \mathbf{w} = \frac{\mathbf{w}'}{\nu}, \quad b = \frac{b'}{\nu}, \quad \rho = \frac{\rho'}{\nu}, \quad \xi_i = \frac{\xi'_i}{\nu} \quad (15)$$

244 These modifications yield the equivalent formulation²

$$\begin{aligned} \min_{\mathbf{w}, b, \xi, \rho} \quad & \|\mathbf{w}\|^2 - 2\rho + \tau \sum_i \xi_i \\ \text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq \rho - \xi_i, \quad \xi_i \geq 0, \end{aligned} \quad (16)$$

245 with the new decision function $f(\mathbf{x}) \equiv f'(\mathbf{x})/\nu$.

²Crisp and Burges [9] showed that the constraint $\rho' \geq 0$ in (14) is redundant and hence it can be removed.

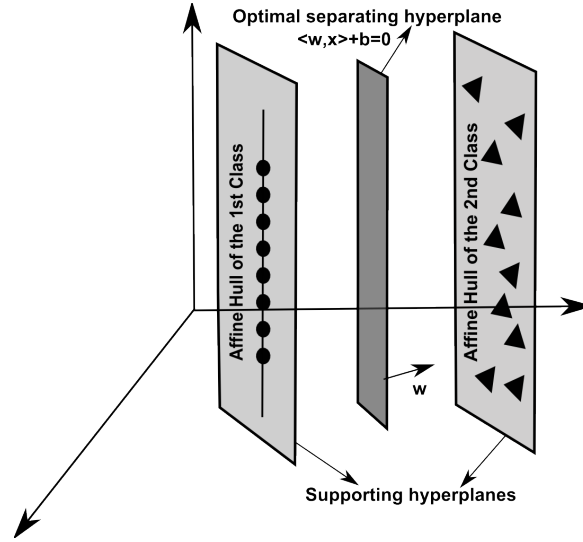


Figure 4: Optimal separating hyperplane between affine hulls of two classes. Note that affine hulls lie on the supporting hyperplanes.

246 Note that two affine hulls are linearly separable if they lie parallel to each other
 247 in the given input space since affine hulls extend to infinity in all directions. In this
 248 case, the supporting hyperplanes yielding the largest margin between affine hulls
 249 will entirely include them so that all affine combinations of samples belonging to
 250 the positive class satisfy $\langle \mathbf{w}, \mathbf{x}_+^{aff} \rangle + b = \rho$ and all affine combinations of samples
 251 belonging to the negative class satisfy $\langle \mathbf{w}, \mathbf{x}_-^{aff} \rangle + b = -\rho$ as illustrated in Fig.
 252 4. Fig. 4 illustrates affine hulls of two classes where the affine hull of the first
 253 class is a line and the affine hull of the second class is a plane. Note that affine
 254 hulls are linearly separable if they lie parallel to each other. Therefore, all samples
 255 of classes and their affine combinations lie on the supporting hyperplanes, which
 256 yield the largest margin between the affine hulls. In case of outliers, we must
 257 construct reduced compact affine hulls that will fit the data robustly. Therefore, we

258 should allow errors for outlier samples from all over the input space, not just the
 259 ones near the decision boundary as illustrated in Fig. 5. To do so, the inequality
 260 constraints in (16) is replaced with equality constraints $y_i(\langle \mathbf{w}, \mathbf{x} \rangle + b) = \rho - \delta_i \xi_i$,
 261 where δ_i is a term which takes values +1 or -1 based on the location of outliers
 262 with respect to the supporting hyperplanes. This leads to the new optimization
 263 problem

$$\begin{aligned} \min_{\mathbf{w}, b, \xi, \rho} \quad & \|\mathbf{w}\|^2 - 2\rho + \tau \sum_i \xi_i \\ \text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) = \rho - \delta_i \xi_i, \quad \xi_i \geq 0. \end{aligned} \quad (17)$$

264 To derive the dual, we consider the Lagrangian

$$L(\mathbf{w}, b, \xi, \rho, \alpha, \beta) = \|\mathbf{w}\|^2 - 2\rho + \tau \sum_i \xi_i - \sum_i \alpha_i [y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \rho + \delta_i \xi_i] - \sum_i \beta_i \xi_i, \quad (18)$$

265 where $\beta_i \geq 0$. The Lagrangian L has to be maximized with respect to α_i, β_i and
 266 minimized with respect to \mathbf{w}, b, ξ , and ρ . The optimality conditions yield

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}} = \mathbf{0} & \rightarrow \mathbf{w} = \frac{1}{2} \sum_i \alpha_i y_i \mathbf{x}_i, \\ \frac{\partial L}{\partial b} = 0 & \rightarrow \sum_i \alpha_i y_i = 0, \\ \frac{\partial L}{\partial \rho} = 0 & \rightarrow \sum_i \alpha_i = 2, \\ \frac{\partial L}{\partial \xi_i} = 0 & \rightarrow \alpha_i = \frac{\tau - \beta_i}{\delta_i} \rightarrow -\tau \leq \alpha_i \leq \tau. \end{aligned} \quad (19)$$

267 Thus, the dual of the optimization problem becomes

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{4} \sum_{ij} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ \text{s.t.} \quad & \sum_i \alpha_i y_i = 0, \quad \sum_i \alpha_i = 2, \quad -\tau \leq \alpha_i \leq \tau. \end{aligned} \quad (20)$$

268 This optimization is equivalent to the one given in (11) - here $1/4$ appears in
 269 the objective function, but rescaling objective function with a positive constant
 270 does not change the solution. Therefore, the new formulation based on modified
 271 ν -SVM is in fact equivalent to finding the best separating hyperplane between
 272 the reduced affine hulls that represent classes. We call this method ν -LMC-AH.
 Due to the Karush-Kuhn-Tucker (KKT) conditions, slack variables can occur only

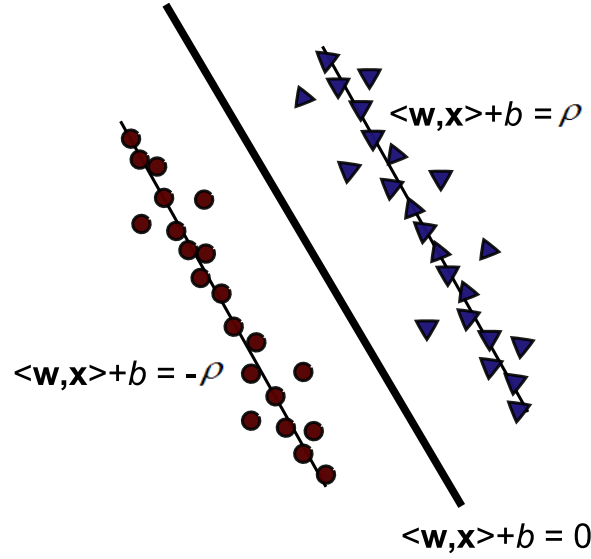


Figure 5: To obtain better separating hyperplanes between affine hulls, we should allow errors for outlier samples from all over the input space.

273
 274 when $\alpha_i = \pm\tau$. To compute offset b , we use the primal constraints and take equal
 275 number of samples with coefficients $\alpha_i \neq \pm\tau$ from positive and negative classes.
 276 Assume that there are l selected samples. By using KKT conditions, we know that
 277 $\xi_i = 0$ for the samples with $\alpha_i \neq \pm\tau$. Thus, the offset will be

$$b = -\frac{1}{2l} \sum_{i=1}^l \sum_{j=1}^n \alpha_j y_j \langle \mathbf{x}_j, \mathbf{x}_i \rangle. \quad (21)$$

278 This offset is not necessarily equivalent to the one given in (13). Therefore, using
 279 geometrically inspired formulation and ν -LMC-AH formulation create separating
 280 hyperplanes with the same normal, but the positions (perpendicular distances from
 281 the origin) of these hyperplanes may be different. It is not a-priori evident that
 282 which offset is the best and one can use other principled methods to determine the
 283 best b for a given problem, e.g., given \mathbf{w} , b can be computed as value yielding the
 284 smallest classification error on a validation set. As in the previous case, extension
 285 to the nonlinear case can be done by using the kernel trick.

286 2.4. Geometric Interpretation of Least Squares SVM Classifier

287 Least Squares Support Vector Machines (LS-SVM) was initially proposed by
 288 Suykens and Vandewalle [24] for classification and nonlinear function estimation
 289 and then new variants of this method have been introduced [7, 21, 22]. The basic
 290 motivation was to simplify the classical SVM formulation without losing much
 291 generalization performance. To this end, inequality constraints in the SVM clas-
 292 sification formulation are heuristically replaced with equality constraints³, and
 293 square of the slack variables are used in the objective function. More formally the
 294 optimization problem is defined as

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + \frac{C}{2} \sum_i \xi_i^2 \\ \text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) = 1 - \xi_i, \end{aligned} \tag{22}$$

295 where C is a user-chosen error penalty term as in classical SVM classifier. It is
 296 shown that the solution is obtained by solving a set of linear equations rather than

³In fact, using equality constraints for nonlinear function estimation was introduced earlier in [19].

297 solving a quadratic programming problem [24]. Note that, in this formulation,
 298 for the linearly separable case, there exist two parallel supporting hyperplanes po-
 299 sitioned such that all points in the positive class satisfy $\langle \mathbf{w}, \mathbf{x} \rangle + b = 1$ and all
 300 points in the negative class satisfy $\langle \mathbf{w}, \mathbf{x} \rangle + b = -1$ where the margin between
 301 these hyperplanes is given by $2/\|\mathbf{w}\|$. As in ν -LMC-AH formulation, this cor-
 302 responds to approximating each class with an affine hull instead of a convex hull
 303 since all samples and their affine combinations are forced to lie on the supporting
 304 hyperplanes. In LS-SVM, L2 norm (squares) of the slack variables are used in
 305 the optimization, but using L1 norm of the slack variables in the objective func-
 306 tion may be more appropriate for some applications since it allows more robust
 307 fitting of data samples. If we use L1 norm of the slack variables, new optimization
 308 problem becomes

$$\begin{aligned}
 \min_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i \\
 \text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) = 1 - \delta_i \xi_i, \quad \xi_i \geq 0,
 \end{aligned} \tag{23}$$

309 where δ_i is a term which takes values +1 or -1. The Lagrangian will be

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i - \sum_i \alpha_i [y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1 + \delta_i \xi_i] - \sum_i \beta_i \xi_i, \tag{24}$$

310 under the constraint $\beta_i \geq 0$. The optimality conditions yield

$$\begin{aligned}
 \frac{\partial L}{\partial \mathbf{w}} = \mathbf{0} & \rightarrow \mathbf{w} = \frac{1}{2} \sum_i \alpha_i y_i \mathbf{x}_i, \\
 \frac{\partial L}{\partial b} = 0 & \rightarrow \sum_i \alpha_i y_i = 0, \\
 \frac{\partial L}{\partial \xi_i} = 0 & \rightarrow \alpha_i = \frac{C - \beta_i}{\delta_i} \rightarrow -C \leq \alpha_i \leq C.
 \end{aligned} \tag{25}$$

311 Thus, the dual of the optimization problem becomes

$$\begin{aligned}
& \min_{\alpha} \quad \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle - \sum_i \alpha_i \\
& s.t. \quad \sum_i \alpha_i y_i = 0, \quad -C \leq \alpha_i \leq C.
\end{aligned} \tag{26}$$

312 Similar to the previous cases, this is a convex quadratic optimization problem
313 with a global minimum. Due to the Karush-Kuhn-Tucker (KKT) conditions, slack
314 variables can occur only when $\alpha_i = \pm C$. To compute offset b , we use the primal
315 constraints and take equal number of samples with coefficients $\alpha_i \neq \pm C$ from
316 positive and negative classes as in ν -LMC-AH. Assume that there are l selected
317 samples. By using KKT conditions, we know that $\xi_i = 0$ for the samples with
318 $\alpha_i \neq \pm C$. Thus, the offset will be

$$b = -\frac{1}{2l} \sum_{i=1}^l \sum_{j=1}^n \alpha_j y_j \langle \mathbf{x}_j, \mathbf{x}_i \rangle. \tag{27}$$

319 A new sample is classified based on the sign of the decision function $f(\mathbf{x}) =$
320 $\langle \mathbf{w}, \mathbf{x} \rangle + b$. Nonlinearization can be done by replacing the dot products $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$
321 with the kernel function $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$. We call this method as C -
322 LMC-AH since it uses error penalty term C . It follows from Proposition 6 of [20]
323 that for appropriate choices of C , the C -LMC-AH algorithm will yield identical
324 results to LMC-AH and ν -LMC-AH classifiers. More precisely, if ν -LMC-AH
325 classification leads to $\rho \geq 0$, then C -LMC-AH method with C set a priori to
326 $1/\rho'n$ (or $1/\rho\nu n$), leads to the same decision function as ν -LMC-AH [6, 20].

327 2.5. Extension to the Multi-Class Classification Problems

328 To use the proposed methods in multi-class classification problems, we can use
329 most of the strategies adopted for extending binary SVM classifiers to the multi-
330 class cases. Here we will discuss the most popular two strategies: one-against-one

(OAO) and one-against-rest (OAR). For a c -class classification problem, the OAR strategy trains c binary classifiers, in which each classifier separates one class from the remaining $c - 1$ classes. All classifiers are needed to be trained on the entire training set, and the class label of a test sample is determined according to the highest output of the classifiers in the ensemble. On the other hand, the OAO strategy constructs all possible $c(c - 1)/2$ binary classifiers out of c classes. The decision of the ensemble is decided by max wins algorithm: Each OAO classifier casts one vote for its preferred class, and the final decision is the class with the most votes. In addition to these we can also use Directed Acyclic Graphs [17] or Binary Decision Trees [28] for multi-class classification.

3. Experiments

We tested⁴ the linear and kernelized versions of the proposed methods LMC-AH, ν -LMC-AH and C -LMC-AH (L1 norm based LS-SVM) on a number of data sets and compared them to the SVM classifier. For the linearly separable case, linear separator is determined by using affine subspace estimation formulation, and subspace dimensions are set by retaining enough leading eigenvectors to account for 95-98% of the total energy in the eigen-decomposition. For the inseparable and nonlinear cases, we used quadratic programming formulations. Both one-against-rest (OAR) and one-against-one (OAO) approaches are used for multi-class classification problems and we report the results of whichever yields the best.

We first tested the linear LMC-AH method on multiple and single shot face

⁴For software see <http://www2.ogu.edu.tr/~mlcv/software.html>.

353 recognition problems to demonstrate that affine hull approximations are more ap-
354 propriate than convex hull approximations when the dimensionality of the input
355 space is high. To assess the generalization performances of kernelized versions of
356 the methods, we tested them on seven low-dimensional databases chosen from the
357 UCI repository.

358 *3.1. Experiments on the Honda/UCSD Database*

359 Honda/UCSD database [15] has been collected for video-based face recogni-
360 tion and it consists of 59 video sequences belonging to 20 individuals. Each video
361 consists of approximately 300-500 frames. It is a fixed database so that 20 of the
362 videos are allocated for training and the remaining 39 for testing. Here, we con-
363 sider face recognition based on multiple images. In this scenario, face recognition
364 problem is defined as taking a set of face images from an unknown person and
finding the most similar set among the database of labeled image sets. We used



Figure 6: Some detected face images from videos belonging to two subjects.

365
366 the cascade face detector of Viola and Jones [27] to detect faces in each video
367 sequence and resized the detected face images to the gray images of size 40×40

Table 1: Classification Rates (%) on the Honda/UCSD Database.

Methods	Clean	Corrup. Training	Corrupted Test	Corrup. Training+Test
LMC-AH	97.44	97.44	92.31	87.18
SVM	94.87	92.31	92.31	82.05

368 followed by histogram equalization. Then, these images are used to construct im-
 369 age sets of individuals. Some of the detected face images are shown in Fig. 6.
 370 We used affine hulls and convex hulls to model image sets, and used the distances
 371 between these models as a similarity measure. In other words, computed margins
 372 between linear affine hulls and convex hulls are used to determine the label of
 373 the tested image sets⁵. We performed several experiments in order to test the ro-
 374 bustness against outliers. In the first experiment, we computed classification rates
 375 based on the clean image sets. Then, we systematically corrupted training and test
 376 sets by adding images from other classes to each set. These images can be seen as
 377 outliers and the changes in classification rates reflect the robustness of the meth-
 378 ods against these outliers. The results are given in Table 3.1. As can be seen from
 379 the table, affine hull approximations yield better results than convex hull approx-
 380 imations in all cases except for the corrupted test where both models give same
 381 results. Thus, affine hulls seem better and more robust models for approximating
 382 image sets.

⁵For the affine hull case we simply used the minimum distance between the estimated affine subspaces using equation (6) whereas soft margin linear SVM algorithm is used to determine the distances between convex hulls.

3.2. Experiments on the AR Face Database

The AR Face data set [16] contains 26 frontal images with different facial expressions, illumination conditions and occlusions for each of 126 subjects, recorded in two 13-image sessions spaced by 14 days. For this experiment, we randomly selected 20 male and 20 female subjects. The images were down-scaled (from 768×576), aligned so that centers of the two eyes fell at fixed coordinates, then cropped to size 105×78 . Some pre-processed images are shown in Fig. 7. Raw pixel values were used as features. For training we randomly selected $n = 7, 13, 20$ samples for each individual, keeping the remaining $26 - n$ for testing. This process was repeated 10 times, with the final classification rate being obtained by averaging the 10 results. The results are shown in Table 3.2. Best re-



Figure 7: Aligned images of one subject from the AR Face database.

sults are obtained by OAR strategy for both tested methods. The proposed method gives better classification rates than soft-margin linear SVM classifier in all cases. The performance difference is more apparent for $n = 7$. These results support our claims, suggesting that affine hulls can be better models for representing classes in high-dimensional spaces when the number of samples is limited.

Table 2: Classification Rates (%) on the AR Face Database.

Methods	$n = 7$	$n = 13$	$n = 20$
LMC-AH	95.19 \pm 0.6	98.95 \pm 0.3	99.62 \pm 0.3
SVM	94.54 \pm 0.6	98.66 \pm 0.2	99.58 \pm 0.3

3.3. Experiments on the UCI Databases

In this group of experiments, we tested the kernelized versions of the methods (quadratic programming formulations) on seven lower-dimensional datasets from the UCI repository: Ionosphere, Iris, Letter Recognition (LR), Multiple Features (MF) - pixel averages, Pima Indian Diabetes (PID), Wine, and Wisconsin Diagnostic Breast Cancer (WDBC). The key parameters of these datasets are summarized in Table 3.3. We used the Gaussian kernels, and all design parameters are set based on random partitions of datasets into training and test sets. OAO strategy was used for multi-class problems. Reported classification rates given in Table 3.3 are computed by 5-fold cross-validation. Although being quite mixed, results indicate that generalization performances of LMC-AH and ν -LMC-AH methods compare favorably with SVM classifier whereas C -LMC-AH generally yields the worst classification accuracy.

4. Summary and Conclusion

We investigated the idea of basing large margin classifiers on affine hulls of classes as an alternative to the SVM (convex hull large margin classifier). Given two affine hull models, their corresponding large margin classifier is easily determined by finding a closest pair of points on these two models and bisecting the

Table 3: Low-Dimensional Databases Selected from UCI Repository

Databases	Number of Classes	Data Set Size	Dimensionality
Ionosphere	2	351	34
Iris	3	150	4
LR	26	20000	16
MF	10	2000	256
PID	2	768	8
Wine	3	178	13
WDBC	2	569	30

Table 4: Classification Rates (%) on the UCI Datasets.

UCI	LMC-AH	ν -LMC-AH	C -LMC-AH	SVM
Ionosphere	93.7 \pm 2.9	93.7 \pm 2.9	93.4 \pm 2.3	92.9 \pm 3.2
Iris	94.7 \pm 2.9	94.7 \pm 2.9	95.3 \pm 3.8	95.3 \pm 3.8
LR	99.98 \pm 0.02	99.98 \pm 0.02	99.89 \pm 0.13	99.64 \pm 0.12
MF	98.4 \pm 0.4	98.4 \pm 0.4	97.8 \pm 0.3	98.0 \pm 0.4
PID	99.9 \pm 0.3	99.9 \pm 0.3	99.9 \pm 0.3	99.9 \pm 0.3
Wine	98.8 \pm 1.6	98.8 \pm 1.6	94.8 \pm 2.6	98.2 \pm 1.6
WDBC	96.0 \pm 2.5	96.0 \pm 2.5	94.9 \pm 3.0	97.6 \pm 0.7

417 displacement between them. We also investigated another formulation obtained
418 by revising the ν -SVM classifier. This formulation yields a separating hyperplane
419 with the same normal as in our first formulation, but the offset is not necessarily
420 the same. This suggests that for a fixed hyperplane normal w in a specific prob-
421 lem, there may be principled procedures to determine the best offset b . To allow
422 soft margin solutions, we first reduce affine hulls to alleviate the effects of outliers
423 and then find the best separating hyperplanes between these reduced models. Such
424 classifiers can also be kernelized, and extension to the multi-class classification is
425 straightforward using any of the standard approaches such as OAO or OAR.

426 The experimental results provided useful insights on the potential application
427 areas of the proposed method. The proposed method is much more efficient than
428 SVM classifier in terms of classification accuracy and real-time performance (test-
429 ing time) when the dimensionality of the sample space is high and affine hulls are
430 linearly separable (in this case solution is easily determined based on subspace es-
431 timation which requires simple linear algebra whereas SVM formulation requires
432 solving a quadratic programming). For the low-dimensional databases generaliza-
433 tion performances of the proposed methods compare favorably with SVM classi-
434 fier but SVM is more efficient in terms of testing time. This is because of the
435 fact that all training data points contribute to the affine hull models (almost all
436 computed α_i coefficients are nonzero), thus the proposed quadratic optimization
437 solutions lack sparseness, and we need more computations to evaluate decision
438 functions. Nevertheless, some pruning techniques can be employed to overcome
439 this problem.

440 **Acknowledgment**

441 This work was supported by the Young Scientists Award Programme (TÜBA-
442 GEBİP/2010) of the Turkish Academy of Sciences.

443 **References**

- 444 [1] K. P. Bennett and E. J. Bredensteiner, 2000. Duality and geometry in SVM
445 classifiers, International Conference on Machine Learning.
- 446 [2] C. J. C. Burges, 1998. tutorial on support vector machines for pattern recog-
447 nition, Data Mining and Knowledge Discovery, vol. 2, pp. 121-167.
- 448 [3] H. Cevikalp and B. Triggs and R. Polikar, 2008. Nearest hyperdisk methods
449 for high-dimensional classification, International Conference on Machine
450 Learning.
- 451 [4] H. Cevikalp and D. Larlus and M. Neamtu and B. Triggs and F. Jurie, 2008.
452 Manifold based local classifiers: linear and nonlinear approaches, Journal of
453 Signal Processing Systems (published online).
- 454 [5] H. Cevikalp and B. Triggs, 2009. Large Margin Classifiers Based on Convex
455 Class Models, International Conference on Computer Vision Workshops.
- 456 [6] C.C. Chang and C.J. Lin, 2001. Training ν -support vector classifiers: Theory
457 and algorithms, Neural Computation, vol. 13, no 9, pp. 2119-2147.
- 458 [7] W. Chu and C. J. Ong and S. Keerthi, 2005. An improved conjugate gradient
459 scheme to the solution of least squares SVM, IEEE Transactions on Neural
460 Networks, vol. 16, pp. 498-501.

- 461 [8] C. Cortes and V. Vapnik, 1995. Support vector networks, Machine Learning,
462 vol.20, pp. 273-297.
- 463 [9] D. J. Crisp and C. J. Burges, 1999. A geometric interpretation of ν -SVM
464 classifiers, Neural Information Processing Systems.
- 465 [10] C. Ding and D. Zhou and X. He and H. Zha, 2006. R1-pca: Rotational invari-
466 ant l1-norm principal component analysis for robust subspace factorization,
467 International Conference on Machine Learning.
- 468 [11] M. B. Gulmezoglu and V. Dzhafarov and A. Barkana, 2001. The com-
469 mon vector approach and its relation to principal component analysis, IEEE
470 Trans. Speech Audio Proc., vol. 9, pp. 655-662.
- 471 [12] G. E. Hinton, P. Dayan, and M. Revow, 1997. Modeling the manifolds of
472 images of handwritten digits, IEEE Trans. on Neural Networks, vol. 18, pp.
473 65-74.
- 474 [13] Q. Ke and T. Kanade, 2005. Robust L1 norm factorization in the presence of
475 outliers and missing data by alternative convex programming, IEEE Com-
476 puter Society Conference on Computer Vision and Pattern Recognition.
- 477 [14] J. Laaksonen, 1997. Subspace classifiers in recognition of handwritten digits,
478 Technical Report.
- 479 [15] K. C. Lee and J. Mo and M. H. Yang and D. Kriegman, 2003. Video-based
480 face recognition using probabilistic appearance manifolds, IEEE Computer
481 Society Conference on Computer Vision and Pattern Recognition.

- 482 [16] A. M. Martinez and R. Benavente, 1998. The AR Face Database, Technical
483 Report, Computer Vision Center, Barcelona, Spain.
- 484 [17] J. C. Platt and N. Cristianini and J. Shawe-taylor, 2000. Large margin dags
485 for multiclass classification, Advances in Neural Information Processing
486 Systems.
- 487 [18] S. T. Roweis and L. K. Saul, 2000. Nonlinear dimensionality reduction by
488 locally linear embedding, Science, vol. 290, pp. 2323-2326.
- 489 [19] C. Saunders and A. Gammerman and V. Vovk, 1998. Ridge regression learn-
490 ing algorithm in dual variables, International Conference on Machine Learn-
491 ing.
- 492 [20] B. Schölkopf and A. J. Smola and R. C. Williamson, and P. L. Bartlett, 2000.
493 New support vector algorithms, Neural Computation, vol. 12, pp. 1207-
494 1245.
- 495 [21] J. A. K. Suykens and L. Lukas and J. Vandewalle, 2000. Sparse least squares
496 support vector machine classifiers, in Proc. of the European Symposium on
497 Artificial Neural Networks (ESANN'2000), Bruges, Belgium, pp. 37-42.
- 498 [22] J. A. K. Suykens and J. De Brabanter and L. Lukas and J. Vandewalle, 2002.
499 Weighted least squares support vector machines: robustness and sparse ap-
500 proximation, Neurocomputing, vol. 48, no. 1-4, pp. 85-105.
- 501 [23] J. A. K. Suykens and T. Van Gestel and J. De Brabanter and B. De Moor
502 and J. Vandewalle, 2002. Least Squares Support Vector Machines, World
503 Scientific Publishing Co. Pte. Ltd.

- 504 [24] J. A. K. Suykens and J. Vandewalle, 1999. Least squares support vector ma-
505 chine classifiers, *Neural Processing Letters*, vol. 9, pp. 293-300.
- 506 [25] J. Verbeek, 2006. Learning non-linear image manifolds by global alignment
507 of local linear models, *IEEE Trans. on Pattern Analysis and Machine Intel-*
508 *ligence*, vol. 28, pp. 1236-1250.
- 509 [26] P. Vincent and Y. Bengio, 2001. K-local hyperplane and convex distance
510 nearest neighbor algorithms, *Advances in Neural Information Processing*
511 *Systems*.
- 512 [27] P. Viola and M. Jones, 2004. Robust real-time face detection, *International*
513 *Journal of Computer Vision*, vol. 57, pp. 137-154.
- 514 [28] V. Vural and J. G. Dy, 2004. A hierarchical method for multi-class support
515 vector machines, *International Conference on Machine Learning*.
- 516 [29] Z. Xiaofei and S. Yong, 2009. Affine Subspace Nearest Points Classification
517 Algorithm for Wavelet Face Recognition, 2009 WRI World Congress on
518 Computer Science and Information Engineering.