MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING
DEPARTMENT OF BRAIN AND COGNITIVE SCIENCES

A.I. Memo No. 16XX May 1999
C.B.C.L Paper No. 2XX

A Note on Support Vector Machine Degeneracy

Ryan Rifkin, Massimiliano Pontil and Alessandro Verri

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.
The pathname for this publication is: ai-publications/1500-1999/AIM-16XX.ps

Abstract

When training Support Vector Machines (SVMs) over non-separable data sets, one sets the threshold b using any dual cost coefficient that is strictly between the bounds of 0 and C. We show that there exist SVM training problems with dual optimal solutions with all coefficients at bounds, but that *all* such problems are degenerate in the sense that the "optimal separating hyperplane" is given by $\mathbf{w} = \mathbf{0}$, and the resulting (degenerate) SVM will classify all future points identically (to the class that supplies more training data). We also derive necessary and sufficient conditions on the input data for this to occur. Finally, we show that an SVM training problem can always be made degenerate by the addition of a *single* data point belonging to a certain unbounded polyhedron, which we characterize in terms of its extreme points and rays.

This report describes research done at the Center for Biological & Computational Learning and the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. This research was sponsored by the Office of Naval Research under contract No. N00014-93-1-0385 and contract No. N00014-95-1-0600. Partial support was also provided by Daimler-Benz AG, Eastman Kodak, Siemens Corporate Research, Inc., AT&T, Digital Equipment Corporation, Central Research Institute of Electrical Power Industry, and Honda.

1 Introduction

We are given l examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)$, with $\mathbf{x}_i \in \mathbb{R}^n$ and $y_i \in \{-1, 1\}$ for all i. The SVM training problem is to find a hyperplane and threshold $(\mathbf{w}, b)$ that separates the positive and negative examples with maximum margin, penalizing misclassifications linearly in a user-selected penalty parameter $C > 0$.[1] This formulation was introduced in [2]. For a good introduction to SVMs and the nonlinear programming problems involved in their training, see [3] or [1]. We train an SVM by solving either of the following pair of dual quadratic programs:

$$\text{(P)} \quad \min_{\mathbf{w}, b, \boldsymbol{\Xi}} \quad \tfrac{1}{2}\|\mathbf{w}\|^2 + C(\textstyle\sum_{i=1}^{\ell} \xi_i) \qquad \begin{aligned} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) &\geq 1 - \xi_i \\ \xi_i &\geq 0 \end{aligned}$$

$$\text{(D)} \quad \max_{\boldsymbol{\Lambda}} \quad \boldsymbol{\Lambda} \cdot \mathbf{1} - \tfrac{1}{2}\boldsymbol{\Lambda}\mathbf{D}\boldsymbol{\Lambda} \qquad \begin{aligned} \boldsymbol{\Lambda} \cdot \mathbf{y} &= 0 \\ \lambda_i &\leq C \\ \lambda_i &\geq 0 \end{aligned}$$

$\mathbf{D}$ is the symmetric positive semidefinite matrix defined by $D_{ij} \equiv y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$. Throughout this note, we use the convention that if an equation contains i as an unsummed subscript, the corresponding equation is replicated for all $i \in \{1, \ldots, l\}$.

In practice, the dual program is solved.[2] However, for this pair of primal-dual problems, the KKT conditions are necessary and sufficient to characterize optimal solutions. Therefore, $\mathbf{w}, b, \boldsymbol{\Xi}$, and $\boldsymbol{\Lambda}$ represent a pair of primal and dual optimal solutions if and only if they satisfy the KKT conditions. Additionally, any primal and dual feasible solutions with identical objective values are primal and dual optimal. The KKT conditions (for the primal problem) are as follows:

$$\begin{aligned}
\mathbf{w} - \sum_{i=1}^{\ell} \lambda_i y_i \mathbf{x}_i &= 0 && (1) \\
\sum_{i=1}^{\ell} \lambda_i y_i &= 0 && (2) \\
C - \lambda_i - \mu_i &= 0 && (3) \\
y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i &\geq 0 && (4) \\
\lambda_i\{y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i\} &= 0 && (5) \\
\mu_i \xi_i &= 0 && (6) \\
\xi_i, \lambda_i, \mu_i &\geq 0 && (7)
\end{aligned}$$

The μ_i are Lagrange multipliers associated with the ξ_i; they do not appear explicitly in either (P) or (D). The KKT conditions will be our major tool for investigating the properties of solutions to (P) and (D).

Suppose that we have solved (D) and possess a dual optimal solution $\boldsymbol{\Lambda}$. Equation (1) allows us to determine $\mathbf{w}$ for the associated primal optimal solution. Further suppose that there exists an i such that $0 < \lambda_i < C$. Then, by equation (3), $u_i > 0$, and by equation (6), $\xi_i = 0$. Because $\lambda_i \neq 0$, equation (5) tells us that $y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i = 0$. Using $\xi_i = 0$, we see that we can determine the threshold b using the equation $b = 1 - y_i(\mathbf{x}_i \cdot \mathbf{w})$.

[1] Actually, we penalize linearly points for which $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) < 1$; such points are not actually "misclassifications" unless $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) < 0$.

[2] SVMs in general use a nonlinear kernel mapping. In this note, we explore the linear simplification in order to gain insight into SVM behavior. Our analysis holds identically in the nonlinear case.

Once b is known, we can determine the ξ_i by noting that $\xi_i = 0$ if $\lambda_i \neq C$ (by equations (3) and (6)), and that $\xi_i = 1 - y_i(\mathbf{x}_i \cdot \mathbf{w} + b)$ otherwise (by equation (5)). However, this is not strictly necessary, as it is $\mathbf{w}$ and b that must be known in order to classify future instances.

We note that our ability to determine b and Ξ is crucially dependent on the existence of a λ_i strictly between 0 and C. Additionally, the optimality conditions, and therefore the SVM training algorithm derived in Osuna's thesis [3], depend on the existence of such a λ_i as well. On page 49 of his thesis Osuna states that "We have not found a proof yet of the existence of such λ_i, or conditions under which it does not exist." Other discussions of SVM's ([1], [2]) also implicitly assume the existence of such a λ_i.

In this paper, we show that there need not exist a λ_i strictly between bounds. Such cases are a subset of *degenerate* SVM training problems: those problems where the optimal separating "hyperplane" is $w = 0$, and the optimal solution is to assign all future points to the same class. We derive a strong characterization of SVM degeneracy in terms of conditions on the input data. We go on to show that any SVM training problems can be made degenerate via the addition of a *single* training point, and that, assuming the two classes are of different cardinalities, this new training point can fall anywhere in a certain unbounded polyhedron. We provide a strong characterization of this polyhedron, and give a mild condition which will insure non-degeneracy.

2 Support Vector Machine Degeneracy

In this section, we explore SVM training problems with a dual optimal solution satisfying $\lambda_i \in \{0, C\}$ for all i.

We begin by noting and dismissing the trivial example where all training points belong to the same class, say class 1. In this case, it is easily seen that $\mathbf{\Lambda} = \mathbf{0}$, $\Xi = \mathbf{0}$, $\mathbf{w} = \mathbf{0}$, and $b = 1$ represent primal and dual optimal solutions, both with objective value 0.

Definition 1 *A vector $\mathbf{\Lambda}$ is a $\{0, C\}$-solution for an SVM training problem $\mathcal{P}$ if $\mathbf{\Lambda}$ solves (D), $\lambda_i \in \{0, C\}$ for all i and $\mathbf{\Lambda} \neq \mathbf{0}$ (note that this* includes *cases where $\lambda_i = C$ for all i).*

We demonstrate the existence of problems having $\{0, C\}$-solutions with an example where the data lie in $\mathbb{R}^2$:

$\mathbf{x}$	y
$(2,3)$	1
$(2,2)$	-1
$(1,2)$	1
$(1,3)$	-1

$$\mathbf{D} = \begin{bmatrix} 13 & -10 & 8 & -11 \\ -10 & 8 & -6 & 8 \\ 8 & -6 & 5 & -7 \\ -11 & 8 & -7 & 10 \end{bmatrix}$$

Suppose $C = 10$. The reader may easily verify that $\mathbf{\Lambda} = (10, 10, 10, 10)$, $\mathbf{w} = \mathbf{0}$, $b = -1$, $\Xi = (0, 2, 0, 2)$ are feasible primal and dual solutions, both with objective value 40, and are therefore optimal. Actually, given our choice of $\mathbf{\Lambda}$ and w, we may set b anywhere in the closed interval $[-1, 1]$, and set $\Xi = (1 + b, 1 - b, 1 + b, 1 - b)$.

We have demonstrated the possibility of $\{0, C\}$-solutions, but the above example seems highly abnormal. The data are distributed at the four corners of a unit square centered at $(1.5, 2.5)$, with opposite corners being of the same class. The "optimal separating hyperplane" is $\mathbf{w} = \mathbf{0}$, which is not a hyperplane at all. We now proceed to formally show that *all* SVM training problems which admit $\{0, C\}$-solutions are degenerate in this sense.

The following lemma is obvious from inspection of the KKT conditions:

Lemma 2 *Suppose that* $\mathbf{\Lambda}$ *is a* $\{0, C\}$*-solution to an SVM training problem* $\mathcal{P}_1$ *with* $C = C_1$*. Given a new SVM training problem* $\mathcal{P}_2$ *with identical input data and* $C = C_2$*,* $(C_2/C_1) \cdot \mathbf{\Lambda}$ *is dual optimal for* $\mathcal{P}_2$*. The corresponding primal optimal solution(s) is (are) unchanged.*

We see that $\{0, C\}$-solutions are not dependent on a particular choice of C. This in turn implies the following:

Lemma 3 *If* $\mathbf{\Lambda}$ *is a* $\{0, C\}$*-solution to an SVM training problem* $\mathcal{P}$*,* $\mathbf{D} \cdot \mathbf{\Lambda} = \mathbf{0}$*.*

PROOF:Since $\mathbf{D}$ is symmetric positive semidefinite, we can write $\mathbf{D} = \mathbf{R}\mathbf{\Sigma}\mathbf{R}^T$, where $\mathbf{\Sigma}$ is a diagonal matrix with the (nonnegative) eigenvalues of $\mathbf{D}$ in descending order on the diagonal, $\mathbf{R}$ is an orthogonal basis of corresponding eigenvectors of $\mathbf{D}$, and $\mathbf{R}\mathbf{R}^T = \mathbf{I}$. If $\mathbf{D} \cdot \mathbf{\Lambda} \neq \mathbf{0}$, then for some index k, $\sigma_k \geq 0$ and $\mathbf{R}_k \cdot \mathbf{\Lambda} \neq \mathbf{0}$.
For any value of C, let $\mathbf{\Lambda}_C$ be the $\{0, C\}$-solution obtained by adjusting $\mathbf{\Lambda}$ appropriately. This solution is dual optimal for a problem having input data identical to $\mathcal{P}$, with a new value of C, by Lemma 2.

$$\begin{aligned}
\mathbf{\Lambda}_C \mathbf{D} \mathbf{\Lambda}_C &= \sum_{j=1}^{l} \sigma_j \|\mathbf{R}_j \cdot \mathbf{\Lambda}_C\|^2 \\
&\geq \sigma_k \|\mathbf{R}_k \cdot \mathbf{\Lambda}_C\|^2 \\
&= \sigma_k C^2 \|\mathbf{R}_k \cdot \mathbf{\Lambda}_1\|^2
\end{aligned}$$

Define S to be the number of non-zero elements in $\mathbf{\Lambda}$. As we vary C, the optimal dual objective value of our family of $\{0, C\}$-solutions is given by:

$$\begin{aligned}
f_{\mathbf{\Lambda}}(C) &= \mathbf{\Lambda}_C \cdot \mathbf{1} - \frac{1}{2} \mathbf{\Lambda_C} \mathbf{D} \mathbf{\Lambda}_C \\
&\leq SC - \frac{1}{2} \sigma_k C^2 \|\mathbf{R}_k \cdot \mathbf{\Lambda}_1\|^2
\end{aligned}$$

However, if

$$C^* > \frac{2S}{\sigma_k \|\mathbf{R}_k \cdot \mathbf{\Lambda}_1\|^2}$$

$f_{\mathbf{\Lambda}}(C^*) < 0$. This is a contradiction, for $\mathbf{\Lambda} = \mathbf{0}$ is feasible in $\mathcal{P}$ with objective value zero, and zero is therefore a *lower bound* on the value of any optimal solution to $\mathcal{P}$, regardless of the value of C. $\Box$

Theorem 4 *If* $\mathbf{\Lambda}$ *is a* $\{0, C\}$*-solution to an SVM training problem* $\mathcal{P}$*,* $\mathbf{w} = \mathbf{0}$ *in all primal optimal solutions.*

PROOF:
Any optimal solution must, along with $\mathbf{\Lambda}$, satisfy the KKT conditions. Exploiting this, we see:

$$\begin{aligned}
\mathbf{0} &= \mathbf{D} \cdot \mathbf{\Lambda} \\
\Longrightarrow 0 &= \mathbf{\Lambda} \mathbf{D} \mathbf{\Lambda} \\
&= \sum_{i=1}^{l} \sum_{j=1}^{l} \lambda_i D_{ij} \lambda_j \\
&= \sum_{i=1}^{l} \sum_{j=1}^{l} \lambda_i y_i y_j \mathbf{x}_i \mathbf{x}_j \lambda_j \\
&= (\sum_{i=1}^{l} \lambda_i y_i \mathbf{x}_i) \cdot (\sum_{j=1}^{l} \lambda_j y_j \mathbf{x}_j) \\
&= \mathbf{w} \cdot \mathbf{w} \\
\Longrightarrow \mathbf{w} &= \mathbf{0}
\end{aligned}$$

$\square$

This is a key result. It states that if our dual problem admits a $\{0, C\}$-solution, the "optimal separating hyperplane" is $\mathbf{w} = \mathbf{0}$. In other words, it is of no value to construct a hyperplane at all, no matter how expensive misclassifications are, and the optimal classifier will classify all future data points using only the threshold b. Our data must be arranged in such a way that we may as well "de-metrize" our space by throwing away all information about where our data points are located, and classify all points identically.

The converse of this statement is false: given an SVM training problem $\mathcal{P}$ that admits a primal solution with $\mathbf{w} = \mathbf{0}$, it is *not* necessarily the case that all dual optimal solutions are $\{0, C\}$-solutions, nor even that a $\{0, C\}$-solution necessarily *exists*, as the following example, constructed from the first example by "splitting" a data point into two new points whose *average* is one of the original points, shows:

$\mathbf{x}$	y
$(2, 3)$	1
$(2, 2)$	-1
$(1, 1.5)$	1
$(1, 2.5)$	1
$(1, 3)$	-1

$$\mathbf{D} = \begin{bmatrix} 13 & -10 & 6.5 & 9.5 & -11 \\ -10 & 8 & -5 & -7 & 8 \\ 6.5 & -5 & 3.25 & 4.75 & -5.5 \\ 9.5 & -7 & 4.75 & 7.25 & -8.5 \\ -11 & 8 & -5.5 & -8.5 & 10 \end{bmatrix}$$

Again letting $C = 10$, the reader may verify that setting $\mathbf{\Lambda} = (10, 10, 5, 5, 10)$, $\mathbf{w} = \mathbf{0}$, $b = -1$, $\mathbf{\Xi} = (0, 20, 0, 20, 0, 0)$ are feasible primal and dual solutions, both with objective value 40, and are therefore optimal. With more effort, the reader may verify that $\mathbf{\Lambda} = \{10, 10, 5, 5, 10\}$ is the unique optimal solution to the dual problem, and therefore no $\{0, C\}$-solution exists.

Although our initial motivation was to study problems with optimal solutions having every dual coefficient λ_i at bounds, we gain additional insight by studying the following, broader class of problems.

Definition 5 *An SVM training problem $\mathcal{P}$ is* **degenerate** *if there exists an optimal primal solution to $\mathcal{P}$ in which $\mathbf{w} = \mathbf{0}$.*

By Theorem 4, any problem that admits a $\{0, C\}$-solution is degenerate. As in the $\{0, C\}$-solution case, one can use the KKT conditions to easily show that the degeneracy of an SVM

training problem is independent of the particular choice of the parameter C, and that $\mathbf{w} = \mathbf{0}$ in *all* primal optimal solutions of a degenerate training problem.
For degenerate SVM training problems, even though there is no optimal separating hyperplane in the normal sense, we still call those data points that contribute to the "expansion" $\mathbf{w} = \mathbf{0}$ with $\lambda_i \neq 0$ support vectors. Given an SVM training problem $\mathcal{P}$, define K_i to be the index set of points in class i, $i \in \{1, -1\}$.

Lemma 6 *Given a degenerate SVM training problem $\mathcal{P}$, assume without loss of generality that $|K_{\text{-}1}| \leq |K_1|$. Then all points in class -1 are support vectors; furthermore, $\lambda_i = C$ if $i \in K_{\text{-}1}$. Additionally, if $|K_{\text{-}1}| = |K_1|$, the (unique) dual optimal solution is $\mathbf{\Lambda} = \mathbf{C}$.*

PROOF:Because $\mathbf{w} = \mathbf{0}$, the primal constraints reduce to:

$$y_i b \geq 1 - \xi_i$$

If $|K_{\text{-}1}| < |K_1|$, the optimal value of b is 1, and ξ_i is positive for $i \in |K_{\text{-}1}|$. Therefore, $\lambda_i = C$ for $i \in K_{\text{-}1}$ (by Equations 6 and 3).
Assume $|K_{\text{-}1}| = |K_1|$. We may (optimally) choose b anywhere in the range $[-1, 1]$. If $b \leq 0$, all points in class 1 have $\lambda_i = C$, and if $b \geq 0$, all points in class -1 have $\lambda_i = C$. In either case, there are at least $|K_{\text{-}1}|$ points in a single class satisfying $\lambda_i = C$. But equation (2) says that the sum of the λ_i for each class must be equal, and since no λ_i may be greater then C, we conclude that every λ_i is equal to C in *both* classes. □

Finally, we derive conditions on the input data for a degenerate SVM training problem $\mathcal{P}$.

Theorem 7 *Given an SVM training problem $\mathcal{P}$, assume without loss of generality that $|K_{\text{-}1}| \leq |K_1|$. Then:*
a. $\mathcal{P}$ is degenerate if and only if there exists a set of multipliers $\mathbf{\Omega}$ for the points in K_1 satisfying:

$$0 \leq \omega_i \leq 1$$

$$\sum_{i \in K_{\text{-}1}} \mathbf{x}_i = \sum_{i \in K_1} \omega_i \mathbf{x}_i$$

$$\sum_{i \in K_1} \omega_i = |K_{\text{-}1}|$$

b. $\mathcal{P}$ admits a $\{0, C\}$-solution if and only if $\mathcal{P}$ is degenerate and the ω_i in part (a) may all be chosen to be 0 *or* 1.

PROOF:
(a, $\Rightarrow$) Suppose $\mathcal{P}$ is degenerate. Consider a modification of $\mathcal{P}$ with identical input data, but $C = 1$; this problem is also degenerate. All points in class -1 are support vectors, and their associated λ_i are at 1, by Lemma 6. Letting $\mathbf{\Lambda}$ be any dual optimal solution to $\mathcal{P}$, we see that letting $\omega_i = \lambda_i$ for $i \in K_1$ and applying Equation (2) demonstrates the existence of the ω_i.
(a, $\Leftarrow$) Given ω_i satisfying the condition, we easily see that $\lambda_i = C$ for $i \in K_{\text{-}1}, \lambda_i = \omega_i C$ for $i \in K_1$ induces a pair of optimal primal and dual solutions to $\mathcal{P}$ with $\mathbf{w} = \mathbf{0}$ using the KKT conditions.
(b, $\Rightarrow$) Given a $\{0, C\}$-solution, $\mathbf{w} = \mathbf{0}$ in an associated primal solution by Theorem 4, and setting $\omega_i = \lambda_i / C$ for $i \in K_1$ satisfies the requirements on $\mathbf{\Omega}$.
(b, $\Leftarrow$) Let $\lambda_i = \omega_i C$ for $i \in K_1$, and apply the KKT conditions. □

3 The Degenerating Polyhedron

Theorem 7 indicates that it is *always* possible to make an SVM training problem degenerate by adding a *single* new data point. We now proceed to characterize the set of individual points whose addition will make a given problem degenerate. For the remainder of this section, we assume that $|K_{-1}| \leq |K_1|$, and we denote $\sum_{i \in K_{-1}} \mathbf{x}_i$ by $\mathbf{V}$, and $|K_{-1}|$ by n.
Suppose we choose, for each $i \in K_1$, an $\omega_i \in [0, 1]$, satisfying $n - 1 \leq \sum_{i \in K_1} \omega_i < n$. It is clear from the conditions of Theorem 7 that if we add a *new* data point

$$\mathbf{x}_c = \frac{V - \sum\limits_{i \in K_1} \omega_i \mathbf{x}_i}{n - \sum\limits_{i \in K_1} \omega_i}$$

that the problem becomes degenerate, where the new point has a multiplier given by $\omega_c = n - \sum_{i \in K_1} \omega_i$, and that all single points whose additions would make the problem degenerate can be found in such a manner. We denote the set of points so obtained by $\mathbf{X_D}$.
We introduce the following notation. For $k \leq n$, we let S_k denote the set containing all possible sums of k points in K_1. Given a point $\mathbf{s} \in S_k$, we define an indicator function $\chi_{\mathbf{s}} : K_1 \rightarrow \{0, 1\}$ with the property $\chi_{\mathbf{s}}(\mathbf{x}_i) = 1$ if and only if $\mathbf{x}_i$ is one of the k points of K_1 that were summed to make $\mathbf{x}$.
The region $\mathbf{X_D}$ is in fact a polyhedron whose extreme points and extreme rays are of the form $V - x$ for $x \in S_{n-1}$ and $\S_n$, respectively. More specifically, we have the following theorem; the proof is not difficult, but it is rather technical, and we defer it to Appendix A:

Theorem 8 *Given a non-degenerate problem $\mathcal{P}$, consider the polyhedron*

$$\mathbf{P_D} \equiv \{(\mathbf{V} - \mathbf{s}^p) + (\mathbf{V} - \mathbf{x}^\mathbf{r}) \mid \lambda_{\mathbf{s}^p}, \alpha_{\mathbf{s}^r} \geq 0, = 1\}$$

Then $\mathbf{P_D} = \mathbf{X_D}$.

An example is shown in Figure 1. On the one hand, the idea that the addition of a single data point can make an SVM training problem degenerate seems to bode ill for the usefulness of the method. Indeed, SVMs are in some sense not robust. This is a consequence of the fact that because errors are penalized in the L_1 norm, a *single* outlier can have *arbitrarily large* effects on the separating hyperplane. However, the fact that we are able to precisely characterize the "degenerating" polyhedron allows us to provide a positive result as well. We begin by noting that in the example of Figure 1, the entire polyhedron of points whose addition make the problem degenerate is located well away from the initial data. This is not a coincidence. Indeed, using Theorem 8, we may easily derive the following theorem:

Theorem 9 *Given a non-degnerate problem $\mathcal{P}$ with $|K_{-1}| \leq |K_1|$, suppose there exists a hyperplane w through V/n, the center of mass of K_{-1}, such that all points in K_1 lie on one side of w, and the closest distance between a point in K_1 and w is d. Then all points in the "degenerating" polyhedron P_D lie at least $(|K_{-1}| - 1) * d$ from w on the* other *side of w from K_1.*

Using Theorem 7 we can easily show that if the center of mass of the points in the smaller class (V/n) does not lie in the convex hull of the points in the larger class, our problem is not degenerate, and we may apply Theorem 9 to bound below the distance at which an outlier would

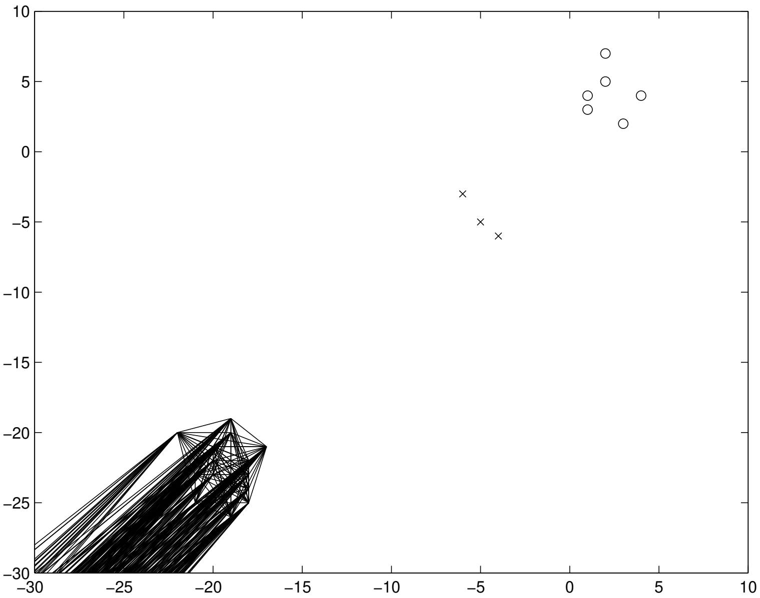

Figure 1: A sample problem, and the "degenerating" polyhedron

have to lie from V/n in order to make the problem degenerate. We conclude that if the class with larger cardinality lies well away from and entirely to one side of a hyperplane through the center of mass of the class of smaller cardinality, our problem is nondegenerate, and any single point we could add to make the problem degenerate would be an extreme outlier, lying on the opposite side of the smaller class from the larger class.

4 Nonlinear SVMs and Further Remarks

The conditions we have derived so far apply to the construction of a linear decision surface. It should be clear that similar arguments apply to nonlinear kernels. In particular, degenerate SVMs will occur if and only if the data satisfy the conditions of Theorem 7 *after* undergoing the nonlinear mapping to the high-dimensional space. It is not necessary that the data be degenerate in the original input space, although examples could be derived where they were degenerate in both spaces, for a particular kernel choice. *The important message of Theorem 7, however, is that while degenerate SVMs are possible, the requirements on the input data are so stringent that one should never expect to encounter them in practice.* On another note, if a degenerate SVM does occur, one simply sets the threshold b to 1 or -1, depending on which class contributes more points to the training set. Thus in all cases, we are able to determine the threshold b. Of course, the wisdom of this approach depends on the data distribution. If our two classes lie largely on top of each other, than classifying according to the larger class may indeed be the best we can do (assuming our examples were drawn randomly from the input distribution). If, instead, our dataset looks more like that of Figure 1, we are better off removing outliers and resolving.

Finally, a brief remark on complexity is in order. The quadratic program (D) can be solved in polynomial time, and solving this program will allow us to determine whether a given SVM training problem $\mathcal{P}$ is degenerate. However, the problem of determining whether or not a $\{0, C\}$-solution exists is not so easy. Certainly, if $\mathcal{P}$ is *not* degenerate, no $\{0, C\}$-solution exists, but the converse is false. Determining the existence of a $\{0, C\}$-solution may be quite difficult: if we require the $\mathbf{x}_i$ to lie in $\mathbb{R}^1$, determining whether a $\{0, C\}$-solution exists is already equivalent to solving the weakly NP-complete problem SUBSET-SUM (see [4] for more information on NP-completeness).[3]

References

[1] C. Burges. A tutorial on support vector machines for pattern recognition. In *Data Mining and Knowledge Discovery*. Kluwer Academic Publishers, Boston, 1998. (Volume 2).

[2] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:1–25, 1995.

[3] E. Osuna. *Support Vector Machines: Training and Applications.* PhD thesis, Massachusetts Institute of Technology, 1998.

[4] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness* W. Freeman and Company, San Francisco, 1979.

A Proof of Theorem 8

Theorem 8 *Given a non-degenerate problem $\mathcal{P}$, consider the polyhedron*

$$\mathbf{P_D} \equiv \{(\mathbf{V} - \mathbf{s}^p) + (\mathbf{V} - \mathbf{x}^\mathbf{r}) \mid \lambda_{\mathbf{s}^p}, \alpha_{\mathbf{s}^r} \geq 0, = 1\}$$

PROOF:
(a, $\mathbf{P_D} \subseteq \mathbf{X_D}$) Given a set of $\lambda_{\mathbf{x}^p}$ and $\alpha_{\mathbf{s}^r}$ satisfying $\lambda_{\mathbf{s}^p}, \alpha_{\mathbf{s}^r} \geq 0, = 1$, we define $A \equiv$, and set

$$\omega_c = \frac{1}{1 + A},$$

and, for $i \in K_1$, we set

$$\omega_i = \omega_c(\chi_{\mathbf{s}^p}(\mathbf{x}_i) + \chi_{\mathbf{s}^r}(\mathbf{x}_i))$$

Then $0 \leq \omega_i \leq 1$ for each $i \in K_1$, and

$$\sum_{i \in K_1} \omega_i = \frac{n - 1 + nA}{1 + A} = n - \frac{1}{1 + A},$$

which is in $[n - 1, n)$, so we conclude that the assigned w_i are valid. Finally, substituting into Equation (8), we find:

$$\begin{aligned} \frac{V - \sum_{i \in K_1} \omega_i \mathbf{x}_i}{n - \sum_{i \in K_1} \omega_i} &= \frac{V - \frac{1}{1+A} \sum_{i \in K_1} (\chi_{\mathbf{s}^p}(\mathbf{x}_i) + \chi_{\mathbf{s}^r}(\mathbf{x}_i))\mathbf{x}_i}{\frac{1}{1+A}} \\ &= (1 + A)V - \mathbf{s}^p - \mathbf{s}^r \\ &= (V - \mathbf{s}^p) + (V - \mathbf{s}^r) \end{aligned}$$

[3] Because the problem is only weakly NP-complete, given a bound on the size of the numbers involved, the problem is polynomially solvable.

We conclude that $\mathbf{P_D} \subseteq \mathbf{X_D}$.
(b, $\mathbf{X_D} \subseteq \mathbf{P_D}$) Our proof is by construction: given a set of ω_i, $i \in K_1$, we show how to choose $\lambda_{\mathbf{s}^p}$ and $\alpha_{\mathbf{s}^r}$ so that:

$$
\begin{array}{rcl}
\lambda_{\mathbf{s}^p} & \geq & 0 \quad \forall \mathbf{s}^p \in S_{n-1} \\
 & = & 1 \\
\alpha_{\mathbf{s}^r} & \geq & 0 \quad \forall \mathbf{x}_r \in S_n \\
\dfrac{V - \sum\limits_{i \in K_1} \omega_i \mathbf{x}_i}{n - \sum\limits_{i \in K_1} \omega_i} & = & (V - \mathbf{s}^p) + (V - \mathbf{s}^r)
\end{array}
$$

If we impose the reasonable "separability" conditions:

$$
\frac{V}{n - \sum\limits_{i \in K_1} \omega_i} = V + V
$$

$$
\frac{\sum\limits_{i \in K_1} \omega_i \mathbf{x}_i}{n - \sum\limits_{i \in K_1} \omega_i} = \mathbf{s}^p + \mathbf{s}^r
$$

we can easily derive the following:

$$
= \frac{(\sum\limits_{i \in K_1} \omega_i + 1) - n}{n - \sum\limits_{i \in K_1} \omega_i} \equiv A
$$

We are now ready to describe the actual construction. We will first assign the $\alpha_{\mathbf{s}^r}$, then the $\lambda_{\mathbf{s}^p}$. We describe in detail the assignment of the $\alpha_{\mathbf{s}^r}$, the assignment of the $\lambda_{\mathbf{s}^p}$ is essentially similar. We begin by initializing each $\alpha_{\mathbf{s}^p}$ to 0. At each step of the algorithm, we consider the "residual":

$$
\frac{V - \sum\limits_{i \in K_1} \omega_i \mathbf{x}_i}{n - \sum\limits_{i \in K_1} \omega_i} - (V - \mathbf{s}^r) \tag{8}
$$

Note that by expanding each $\mathbf{s}^r$ in the n points of K_1 which sum to it, we can represent (8) as a multiple of V minus a linear combination of the points of K_1 — we will maintain the invariant that this linear combination is actually a nonnegative combination. During a step of the algorithm, we select the n points of K_1 that have the largest coefficients in this expansion. If there is a tie, we expand the set to include all points with coefficients equal to the nth largest coefficient. Let j be the number of points in the set that share the nth largest coefficient, and let k ($\geq n$) be the total size of the selected set. We select the $\binom{j}{n-k+j}$ points $\mathbf{s}^r$ containing the remaining $\max(k-j, 0)$ points with the largest coefficients, and $n-k+j$ of the j points which contain the nth largest coefficient. We will then add equal amounts of each of these $\mathbf{s}^r$ to our representation until some pair of coefficients in the residual that were unequal become equal. This can happen in one of two ways: either the smallest of the coefficients in our set can become equal to a new, still smaller coefficient, or the second smallest coefficient in the set can become equal to the smallest (this can only happen in the case where $k > n$.) At each step of the algorithm, the total number of different coefficients in the residual is reduced by at least one, so, within $|K_1|$ steps, we will be able to assign all the $\alpha_{\mathbf{s}^r}$ (note that at each step of our algorithm, we increase $\left(\frac{j}{n-k+j}\right)$ of the $\alpha_{\mathbf{s}^r}$). The only way the algorithm could break down is if, at some step, there were fewer than n points in K_1 with nonzero coefficients in the residual. Trivially, the algorithm does not break down at the first step — there must always be at least n points with non-zero coefficients initially. To show that the algorithm does not break down at a later step, assume that after assigning coefficients to the $\mathbf{s}^r$ totalling k ($< A$), we are left with j ($< n$) non-zero coefficients. Noting that our algorithm requires that each of the j remaining points with non-zero coefficients is part of each $\mathbf{s}^r$ with a non-zero coefficient, we can see that the the residual value of each of these j points is no more then $\frac{1}{n-w} - k$. We derive the following bound on the *initial* sum of the coefficients, which we call I_{sum}:

$$
\begin{aligned}
I_{sum} &\leq j\left(\frac{1}{n - \sum\limits_{i\in K_1}\omega_i} - k\right) + kn \\
&= \frac{j}{n - \sum\limits_{i\in K_1}\omega_i} + k(n-j) \\
&\leq \frac{n-1}{n - \sum\limits_{i\in K_1}\omega_i} + k \\
&< \frac{n-1}{n - \sum\limits_{i\in K_1}\omega_i} + \frac{\sum\limits_{i\in K_1}\omega_i + 1 - n}{n - \sum\limits_{i\in K_1}\omega_i} \\
&= \frac{\sum\limits_{i\in K_1}\omega_i}{n - \sum\limits_{i\in K_1}\omega_i}
\end{aligned}
$$

But this is a contradiction, I_{sum} must be equal to $\frac{\sum_{i\in K_1}\omega_i}{n-\sum_{i\in K_1}\omega_i}$. We conclude that we are able to assign the $\alpha_{\mathbf{s}^r}$ successfully. Extremely similar arguments hold for the $\lambda_{\mathbf{s}^p}$. $\square$