should be constructed such that they maximize the margin, and at the same time separate the training data with as few exceptions as possible. Sections 7.3 and 7.5 respectively will deal with these two issues.

7.3 Optimal Margin Hyperplanes

Let us now derive the optimization problem to be solved for computing the optimal hyperplane. Suppose we are given a set of examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$, $\mathbf{x}_i \in \mathcal{H}$, $y_i \in \{\pm 1\}$. Here and below, the index i runs over $1, \ldots, m$ by default. We assume that there is at least one negative and one positive y_i. We want to find a decision function $f_{\mathbf{w},b}(\mathbf{x}) = \operatorname{sgn}\left(\langle \mathbf{w}, \mathbf{x}\rangle + b\right)$ satisfying

$$f_{\mathbf{w},b}(\mathbf{x}_i) = y_i. \tag{7.8}$$

If such a function exists (the non-separable case will be dealt with later), canonicality (7.2) implies

$$y_i \left(\langle \mathbf{x}_i, \mathbf{w}\rangle + b\right) \geq 1. \tag{7.9}$$

As an aside, note that out of the two canonical forms of the same hyperplane, $(\mathbf{w}, b)$ and $(-\mathbf{w}, -b)$, only one will satisfy equations (7.8) and (7.11). The existence of class labels thus allows to distinguish two orientations of a hyperplane.

Following the previous section, a separating hyperplane which generalizes well can thus be constructed by solving the following problem:

$$\underset{\mathbf{w}\in\mathcal{H}, b\in\mathbb{R}}{\text{minimize}} \quad \tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2, \tag{7.10}$$

$$\text{subject to} \quad y_i \left(\langle \mathbf{x}_i, \mathbf{w}\rangle + b\right) \geq 1 \text{ for all } i = 1, \ldots, m. \tag{7.11}$$

This is called the *primal optimization problem*.

Problems like this one are the subject of optimization theory. For details on how to solve them, see Chapter 6; for a short intuitive explanation, cf. the remarks following (1.26) in the introductory chapter. We will now derive the so-called *dual problem*, which can be shown to have the same solutions as (7.10). In the present case, it will turn out that it is more convenient to deal with the dual. To derive it, we introduce the Lagrangian,

Lagrangian

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_i \left(y_i(\langle \mathbf{x}_i, \mathbf{w}\rangle + b) - 1\right), \tag{7.12}$$

with Lagrange multipliers $\alpha_i \geq 0$. Recall that as in Chapter 1, we use bold face Greek variables to refer to the corresponding vectors of variables, for instance, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$.

The Lagrangian L must be maximized with respect to α_i, and minimized with respect to $\mathbf{w}$ and b (see Theorem 6.26). Consequently, at this saddle point, the

derivatives of L with respect to the primal variables must vanish,

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0, \quad \frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0, \tag{7.13}$$

which leads to

$$\sum_{i=1}^{m} \alpha_i y_i = 0, \tag{7.14}$$

and

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i. \tag{7.15}$$

The solution vector thus has an expansion in terms of training examples. Note that although the solution $\mathbf{w}$ is unique (due to the strict convexity of (7.10), and the convexity of (7.11)), the coefficients α_i need not be.

According to the KKT theorem (Chapter 6), only the Lagrange multipliers α_i that are non-zero at the saddle point, correspond to constraints (7.11) which are precisely met. Formally, for all $i = 1, \ldots, m$, we have

$$\alpha_i [y_i (\langle \mathbf{x}_i, \mathbf{w} \rangle + b) - 1] = 0. \tag{7.16}$$

Support Vectors

The patterns $\mathbf{x}_i$ for which $\alpha_i > 0$ are called *Support Vectors*. This terminology is related to corresponding terms in the theory of convex sets, relevant to convex optimization (e.g., [334, 45]).[3] According to (7.16), they lie exactly on the margin.[4] All remaining examples in the training set are irrelevant: Their constraints (7.11) are satisfied automatically, and they do not appear in the expansion (7.15), since their multipliers satisfy $\alpha_i = 0$.[5]

This leads directly to an upper bound on the generalization ability of optimal margin hyperplanes. To this end, we consider the so-called leave-one-out method (for further details, see Section 12.2) to estimate the expected test error [335, 559]. This procedure is based on the idea that if we leave out one of the training

3. Given any boundary point of a convex set, there always exists a hyperplane separating the point from the interior of the set. This is called a *supporting hyperplane*.

SVs lie on the boundary of the convex hulls of the two classes, thus they possess supporting hyperplanes. The SV optimal hyperplane is the hyperplane which lies in the middle of the two parallel supporting hyperplanes (of the two classes) with maximum distance.

Conversely, from the optimal hyperplane, we can obtain supporting hyperplanes for all SVs of both classes, by shifting it by $1/\|\mathbf{w}\|$ in both directions.

4. Note that this implies the solution $(\mathbf{w}, b)$, where b is computed using $y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) = 1$ for SVs, *is* in canonical form with respect to the training data. (This makes use of the reasonable assumption that the training set contains both positive and negative examples.)

5. In a statistical mechanics framework, Anlauf and Biehl [12] have put forward a similar argument for the *optimal stability perceptron*, also computed using constrained optimization. There is a large body of work in the physics community on optimal margin classification. Some further references of interest are [310, 191, 192, 394, 449, 141]; other early works include [313].

examples, and train on the remaining ones, then the probability of error on the left out example gives us a fair indication of the true test error. Of course, doing this for a single training example leads to an error of either zero or one, so it does not yet give an estimate of the test error. The leave-one-out method *repeats* this procedure for each individual training example in turn, and averages the resulting errors.

Let us return to the present case. If we leave out a pattern $\mathbf{x}_{i^*}$, and construct the solution from the remaining patterns, the following outcomes are possible (cf. (7.11)):

1. $y_{i^*}\left(\langle \mathbf{x}_{i^*}, \mathbf{w}\rangle + b\right) > 1$. In this case, the pattern is classified correctly and does not lie on the margin. These are patterns that would not have become SVs anyway.

2. $y_{i^*}\left(\langle \mathbf{x}_{i^*}, \mathbf{w}\rangle + b\right) = 1$. In other words, $\mathbf{x}_{i^*}$ exactly meets the constraint (7.11). In this case, the solution $\mathbf{w}$ does not change, even though the coefficients α_i would change: Namely, if $\mathbf{x}_{i^*}$ might have become a Support Vector (i.e., $\alpha_{i^*} > 0$) had it been kept in the training set. In that case, the fact that the solution is the same, no matter whether $\mathbf{x}_{i^*}$ is in the training set or not, means that $\mathbf{x}_{i^*}$ can be written as $\sum_{\mathrm{SVs}} \beta_i y_i \mathbf{x}_i$ with, $\beta_i \geq 0$. Note that condition 2 is *not* equivalent to saying that $\mathbf{x}_{i^*}$ may be written as some linear combination of the remaining Support Vectors: Since the sign of the coefficients in the linear combination is determined by the class of the respective pattern, not any linear combination will do. Strictly speaking, $\mathbf{x}_{i^*}$ must lie in the cone spanned by the $y_i \mathbf{x}_i$, where the $\mathbf{x}_i$ are all Support Vectors.[6] For more detail, see [565] and Section 12.2.

3. $0 < y_{i^*}\left(\langle \mathbf{x}_{i^*}, \mathbf{w}\rangle + b\right) < 1$. In this case, $\mathbf{x}_{i^*}$ lies within the margin, but still on the correct side of the decision boundary. Thus, the solution looks different from the one obtained with $\mathbf{x}_{i^*}$ in the training set (in that case, $\mathbf{x}_{i^*}$ would satisfy (7.11) after training); classification is nevertheless correct.

4. $y_{i^*}\left(\langle \mathbf{x}_{i^*}, \mathbf{w}\rangle + b\right) > 0$. This means that $\mathbf{x}_{i^*}$ is classified incorrectly.

Note that cases 3 and 4 necessarily correspond to examples which would have become SVs if kept in the training set; case 2 potentially includes such situations. Only case 4, however, leads to an error in the leave-one-out procedure. Consequently, we have the following result on the generalization error of optimal margin classifiers [570]:[7]

Leave-One-Out Bound

Proposition 7.4 *The expectation of the number of Support Vectors obtained during training on a training set of size m, divided by m, is an* upper bound *on the expected probability of test error of the SVM trained on training sets of size* $m - 1$.[8]

6. Possible non-uniqueness of the solution's expansion in terms of SVs is related to zero Eigenvalues of $(y_i y_j k(x_i, x_j))_{ij}$, cf. Proposition 2.16. Note, however, the above caveat on the distinction between linear combinations, and linear combinations with coefficients of fixed sign.

7. It also holds for the generalized versions of optimal margin classifiers described in the following sections.

8. Note that the leave-one-out procedure performed with m training examples thus yields

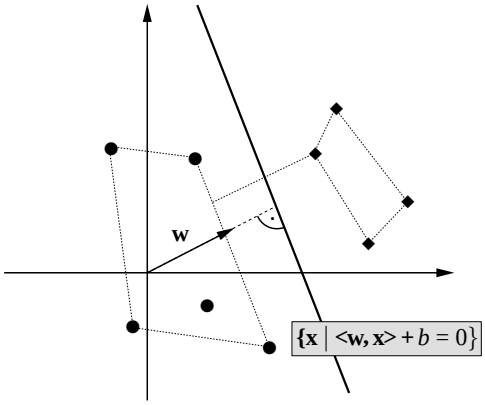

Figure 7.5 The optimal hyperplane (Figure 7.2) is the one bisecting the shortest connection between the convex hulls of the two classes.

A sharper bound can be formulated by making a further distinction in case 2, between SVs that must occur in the solution, and those that can be expressed in terms of the other SVs (see [570, 565, 268, 549] and Section 12.2).

We now return to the optimization problem to be solved. Substituting the conditions for the extremum, (7.14) and (7.15), into the Lagrangian (7.12), we arrive at the dual form of the optimization problem:

Quadratic Program of Optimal Margin Classifier

$$\underset{\boldsymbol{\alpha}\in\mathbb{R}^m}{\text{maximize}} \quad W(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle, \tag{7.17}$$

$$\text{subject to} \quad \alpha_i \geq 0, \quad i = 1, \ldots, m, \tag{7.18}$$

$$\text{and} \quad \sum_{i=1}^{m} \alpha_i y_i = 0. \tag{7.19}$$

On substitution of the expansion (7.15) into the decision function (7.3), we obtain an expression which can be evaluated in terms of dot products, taken between the pattern to be classified and the Support Vectors,

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i \left\langle \mathbf{x}, \mathbf{x}_i \right\rangle + b \right). \tag{7.20}$$

To conclude this section, we note that there is an alternative way to derive the dual optimization problem [38]. To describe it, we first form the convex hulls C_+

a bound valid for training sets of size $m-1$. This difference, however, does not usually mislead us too much. In statistical terms, the leave-one-out error is called *almost unbiased*. Note, moreover, that the statement talks about the *expected probability* of test error — there are thus *two* sources of randomness. One is the expectation over different training sets of size $m-1$, the other is the probability of test error when one of the SVMs is faced with a test example drawn from the underlying distribution generating the data. For a generalization, see Theorem 12.9.

and C_- of both classes of training points,

$$C_\pm := \left\{ \sum_{y_i=\pm 1} c_i \mathbf{x}_i \,\middle|\, \sum_{y_i=\pm 1} c_i = 1, c_i \geq 0 \right\}. \tag{7.21}$$

Convex Hull Separation

It can be shown that the maximum margin hyperplane as described above is the one bisecting the shortest line orthogonally connecting C_+ and C_- (Figure 7.5). Formally, this can be seen by considering the optimization problem

$$\begin{aligned} &\underset{\mathbf{c}\in\mathbb{R}^m}{\text{minimize}} \quad \left\| \sum_{y_i=1} c_i \mathbf{x}_i - \sum_{y_i=-1} c_i \mathbf{x}_i \right\|^2, \\ &\text{subject to} \quad \sum_{y_i=1} c_i = 1, \; \sum_{y_i=-1} c_i = 1, c_i \geq 0, \end{aligned} \tag{7.22}$$

and using the normal vector $\mathbf{w} = \sum_{y_i=1} c_i \mathbf{x}_i - \sum_{y_i=-1} c_i \mathbf{x}_i$, scaled to satisfy the canonicality condition (Definition 7.1). The threshold b is explicitly adjusted such that the hyperplane bisects the shortest connecting line (see also Problem 7.7).

7.4 Nonlinear Support Vector Classifiers

Thus far, we have shown why it is that a large margin hyperplane is good from a statistical point of view, and we have demonstrated how to compute it. Although these two points have worked out nicely, there is still a major drawback to the approach: Everything that we have done so far is linear in the data. To allow for much more general decision surfaces, we now use kernels to nonlinearly transform the input data $x_1, \ldots, x_m \in \mathcal{X}$ into a high-dimensional feature space, using a map $\Phi : x_i \mapsto \mathbf{x}_i$; we then do a linear separation there.

To justify this procedure, Cover's Theorem [113] is sometimes alluded to. This theorem characterizes the number of possible linear separations of m points in general position in an N-dimensional space. If $m \leq N+1$, then all 2^m separations are possible — the VC dimension of the function class is $n+1$ (Section 5.5.6). If $m > N+1$, then Cover's Theorem states that the number of linear separations equals

Cover's Theorem

$$2 \sum_{i=0}^{N} \binom{m-1}{i}. \tag{7.23}$$

The more we increase N, the more terms there are in the sum, and thus the larger is the resulting number. This theorem formalizes the intuition that the number of separations increases with the dimensionality. It requires, however, that the points are in general position — therefore, it does not strictly make a statement about the separability of a given dataset in a given feature space. E.g., the feature map might be such that all points lie on a rather restrictive lower-dimensional manifold, which could prevent us from finding points in general position.

There is another way to intuitively understand why the kernel mapping in-