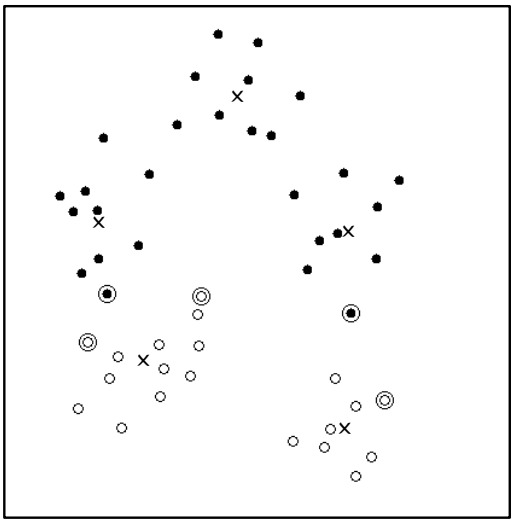

Figure 7.8 RBF centers automatically computed by the Support Vector algorithm (indicated by extra circles), using a Gaussian kernel. The number of SV centers accidentally coincides with the number of identifiable clusters (indicated by crosses found by k-means clustering, with $k = 2$ and $k = 3$ for balls and circles, respectively), but the naive correspondence between clusters and centers is lost; indeed, 3 of the SV centers are circles, and only 2 of them are balls. Note that the SV centers are chosen with respect to the classification task to be solved (from [482]).

algorithm was used to identify the centers (or hidden units) for the RBF network (that is, as a replacement for k-means), exhibited a performance which was in between the previous two. The study concluded that the SVM algorithm yielded two advantages. First, it better identified good expansion patterns, and second, its large margin regularizer led to second-layer weights that generalized better. We should add, however, that using clever engineering, the classical RBF algorithm can be improved to achieve a performance close to the one of SVMs [427].

7.5 Soft Margin Hyperplanes

So far, we have not said much about when the above will actually work. In practice, a separating hyperplane need not exist; and even if it does, it is not always the best solution to the classification problem. After all, an individual outlier in a data set, for instance a pattern which is mislabelled, can crucially affect the hyperplane. We would rather have an algorithm which can tolerate a certain fraction of outliers.

A natural idea might be to ask for the algorithm to return the hyperplane that leads to the *minimal* number of training errors. Unfortunately, it turns out that this is a combinatorial problem. Worse still, the problem is even hard to *approximate*: Ben-David and Simon [34] have recently shown that it is NP-hard to find a hyperplane whose training error is worse by some constant factor than the optimal one. Interestingly, they also show that this can be alleviated by taking into account the concept of the *margin*. By disregarding points that are within some fixed positive margin of the hyperplane, then the problem has polynomial complexity.

Cortes and Vapnik [111] chose a different approach for the SVM, following [40].

Slack Variables

To allow for the possibility of examples violating (7.11), they introduced so-called slack variables,

$$\xi_i \geq 0, \quad \text{where } i = 1, \ldots, m, \tag{7.33}$$

and use relaxed separation constraints (cf. (7.11)),

$$y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b) \geq 1 - \xi_i, \quad i = 1, \ldots, m. \tag{7.34}$$

Clearly, by making ξ_i large enough, the constraint on $(\mathbf{x}_i, y_i)$ can always be met. In order not to obtain the trivial solution where all ξ_i take on large values, we thus need to penalize them in the objective function. To this end, a term $\sum_i \xi_i$ is included in (7.10).

C-SVC

In the simplest case, referred to as the C-SV classifier, this is done by solving, for some $C > 0$,

$$\underset{\mathbf{w} \in \mathcal{H}, \boldsymbol{\xi} \in \mathbb{R}^m}{\text{minimize}} \quad \tau(\mathbf{w}, \boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{m}\sum_{i=1}^{m} \xi_i, \tag{7.35}$$

subject to the constraints (7.33) and (7.34). It is instructive to compare this to Theorem 7.3, considering the case $\rho = 1$. Whenever the constraint (7.34) is met with $\xi_i = 0$, the corresponding point will not be a margin error. All non-zero slacks ξ correspond to margin errors; hence, roughly speaking, the fraction of margin errors in Theorem 7.3 increases with the second term in (7.35). The capacity term, on the other hand, increases with $\|\mathbf{w}\|$. Hence, for a suitable positive constant C, this approach approximately minimizes the right hand side of the bound.

Note, however, that if many of the ξ_i attain large values (in other words, if the classes to be separated strongly overlap, for instance due to noise), then $\sum_{i=1}^{m} \xi_i$ can be significantly larger than the fraction of margin errors. In that case, there is no guarantee that the hyperplane will generalize well.

As in the separable case (7.15), the solution can be shown to have an expansion

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i, \tag{7.36}$$

where non-zero coefficients α_i can only occur if the corresponding example $(\mathbf{x}_i, y_i)$ precisely meets the constraint (7.34). Again, the problem only depends on dot products in $\mathcal{H}$, which can be computed by means of the kernel.

The coefficients α_i are found by solving the following quadratic programming problem:

$$\underset{\boldsymbol{\alpha} \in \mathbb{R}^m}{\text{maximize}} \quad W(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j), \tag{7.37}$$

$$\text{subject to } 0 \leq \alpha_i \leq \frac{C}{m} \text{ for all } i = 1, \ldots, m, \tag{7.38}$$

$$\text{and } \sum_{i=1}^{m} \alpha_i y_i = 0. \tag{7.39}$$

To compute the threshold b, we take into account that due to (7.34), for Support

Vectors x_j for which $\xi_j = 0$, we have (7.31). Thus, the threshold can be obtained by averaging (7.32) over all Support Vectors x_j (recall that they satisfy $\alpha_j > 0$) with $\alpha_j < C$.

In the above formulation, C is a constant determining the trade-off between two conflicting goals: minimizing the training error, and maximizing the margin. Unfortunately, C is a rather unintuitive parameter, and we have no a priori way to select it.[9] Therefore, a modification was proposed in [481], which replaces C by

ν-SVC

a parameter ν; the latter will turn out to control the number of margin errors and Support Vectors.

As a primal problem for this approach, termed the ν-SV classifier, we consider

$$\underset{\mathbf{w}\in\mathcal{H},\boldsymbol{\xi}\in\mathbb{R}^m,\rho,b\in\mathbb{R}}{\text{minimize}} \quad \tau(\mathbf{w},\boldsymbol{\xi},\rho) = \frac{1}{2}\|\mathbf{w}\|^2 - \nu\rho + \frac{1}{m}\sum_{i=1}^{m}\xi_i \tag{7.40}$$

$$\text{subject to} \quad y_i(\langle \mathbf{x}_i, \mathbf{w}\rangle + b) \geq \rho - \xi_i \tag{7.41}$$

$$\text{and} \quad \xi_i \geq 0, \quad \rho \geq 0. \tag{7.42}$$

Note that no constant C appears in this formulation; instead, there is a parameter ν, and also an additional variable ρ to be optimized. To understand the role of ρ, note that for $\boldsymbol{\xi} = 0$, the constraint (7.41) simply states that the two classes are separated by the *margin* $2\rho/\|\mathbf{w}\|$ (cf. Problem 7.4).

Margin Error

To explain the significance of ν, let us first recall the term *margin error*: by this, we denote points with $\xi_i > 0$. These are points which are either errors, or lie within the margin. Formally, the fraction of margin errors is

$$R^{\rho}_{\text{emp}}[g] := \frac{1}{m}\left|\{i|y_i g(x_i) < \rho\}\right|. \tag{7.43}$$

Here, g is used to denote the argument of the sgn in the decision function (7.25): $f = \text{sgn} \circ g$. We are now in a position to state a result that explains the significance

ν-Property

of ν.

Proposition 7.5 ([481]) *Suppose we run ν-SVC with k on some data with the result that $\rho > 0$. Then*

(i) ν is an upper bound on the fraction of margin errors.

(ii) ν is a lower bound on the fraction of SVs.

(iii) Suppose the data $(x_1, y_1), \ldots, (x_m, y_m)$ were generated iid from a distribution $\mathrm{P}(x, y) = \mathrm{P}(x)\mathrm{P}(y|x)$, such that neither $\mathrm{P}(x, y = 1)$ nor $\mathrm{P}(x, y = -1)$ contains any discrete component. Suppose, moreover, that the kernel used is analytic and non-constant. With probability 1, asymptotically, ν equals both the fraction of SVs and the fraction of errors.

The proof can be found in Section A.2.

Before we get into the technical details of the dual derivation, let us take a look

9. As a default value, we use $C/m = 10$ unless stated otherwise.

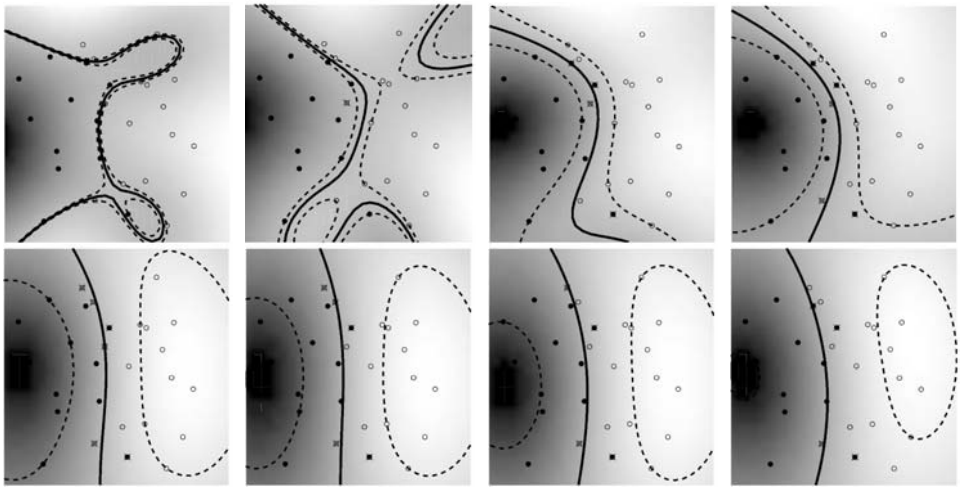

Figure 7.9 Toy problem (task: separate circles from disks) solved using ν-SV classification, with parameter values ranging from $\nu = 0.1$ (top left) to $\nu = 0.8$ (bottom right). The larger we make ν, the more points are allowed to lie inside the margin (depicted by dotted lines). Results are shown for a Gaussian kernel, $k(x, x') = \exp(-\|x - x'\|^2)$.

Table 7.1 Fractions of errors and SVs, along with the margins of class separation, for the toy example in Figure 7.9.
Note that ν upper bounds the fraction of errors and lower bounds the fraction of SVs, and that increasing ν, i.e., allowing more errors, increases the margin.

ν	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
fraction of errors	0.00	0.07	0.25	0.32	0.39	0.50	0.61	0.71
fraction of SVs	0.29	0.36	0.43	0.46	0.57	0.68	0.79	0.86
margin $\rho/\|\mathbf{w}\|$	0.005	0.018	0.115	0.156	0.364	0.419	0.461	0.546

at a toy example illustrating the influence of ν (Figure 7.9). The corresponding fractions of SVs and margin errors are listed in table 7.1.

Derivation of the Dual

The derivation of the ν-SVC dual is similar to the above SVC formulations, only slightly more complicated. We consider the Lagrangian

$$L(\mathbf{w}, \boldsymbol{\xi}, b, \rho, \boldsymbol{\alpha}, \boldsymbol{\beta}, \delta) = \frac{1}{2}\|\mathbf{w}\|^2 - \nu\rho + \frac{1}{m}\sum_{i=1}^{m}\xi_i \\ - \sum_{i=1}^{m}(\alpha_i(y_i(\langle \mathbf{x}_i, \mathbf{w}\rangle + b) - \rho + \xi_i) + \beta_i\xi_i) - \delta\rho, \tag{7.44}$$

using multipliers $\alpha_i, \beta_i, \delta \geq 0$. This function has to be minimized with respect to the primal variables $\mathbf{w}, \boldsymbol{\xi}, b, \rho$, and maximized with respect to the dual variables $\boldsymbol{\alpha}, \boldsymbol{\beta}, \delta$. To eliminate the former, we compute the corresponding partial derivatives

and set them to 0, obtaining the following conditions:

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i, \tag{7.45}$$

$$\alpha_i + \beta_i = 1/m, \tag{7.46}$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0, \tag{7.47}$$

$$\sum_{i=1}^{m} \alpha_i - \delta = \nu. \tag{7.48}$$

Again, in the *SV expansion* (7.45), the α_i that are non-zero correspond to a constraint (7.41) which is precisely met.

Substituting (7.45) and (7.46) into L, using $\alpha_i, \beta_i, \delta \geq 0$, and incorporating kernels for dot products, leaves us with the following quadratic optimization problem for ν-SV classification:

Quadratic Program for ν-SVC

$$\underset{\boldsymbol{\alpha} \in \mathbb{R}^m}{\text{maximize}} \quad W(\boldsymbol{\alpha}) = -\frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j), \tag{7.49}$$

$$\text{subject to} \quad 0 \leq \alpha_i \leq \frac{1}{m}, \tag{7.50}$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0, \tag{7.51}$$

$$\sum_{i=1}^{m} \alpha_i \geq \nu. \tag{7.52}$$

As above, the resulting decision function can be shown to take the form

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i k(x, x_i) + b \right). \tag{7.53}$$

Compared with the C-SVC dual (7.37), there are two differences. First, there is an additional constraint (7.52).[10] Second, the linear term $\sum_{i=1}^{m} \alpha_i$ no longer appears in the objective function (7.49). This has an interesting consequence: (7.49) is now quadratically homogeneous in $\boldsymbol{\alpha}$. It is straightforward to verify that the same decision function is obtained if we start with the primal function

$$\tau(\mathbf{w}, \boldsymbol{\xi}, \rho) = \frac{1}{2}\|\mathbf{w}\|^2 + C\left(-\nu\rho + \frac{1}{m} \sum_{i=1}^{m} \xi_i \right), \tag{7.54}$$

10. The additional constraint makes it more challenging to come up with efficient training algorithms for large datasets. So far, two approaches have been proposed which work well. One of them slightly modifies the primal problem in order to avoid the *other* equality constraint (related to the offset b) [98]. The other one is a direct generalization of a corresponding algorithm for C-SVC, which reduces the problem for each chunk to a linear system, and which does not suffer any disadvantages from the additional constraint [407, 408]. See also Sections 10.3.2, 10.4.3, and 10.6.3 for further details.

i.e., if one does use C, cf. Problem 7.16.

To compute the threshold b and the margin parameter ρ, we consider two sets $S_\pm$, of identical size $s > 0$, containing SVs x_i with $0 < \alpha_i < 1$ and $y_i = \pm 1$, respectively. Then, due to the KKT conditions, (7.41) becomes an equality with $\xi_i = 0$. Hence, in terms of kernels,

$$b = -\frac{1}{2s} \sum_{x \in S_+ \cup S_-} \sum_{j=1}^{m} \alpha_j y_j k(x, x_j), \tag{7.55}$$

$$\rho = \frac{1}{2s} \Big(\sum_{x \in S_+} \sum_{j=1}^{m} \alpha_j y_j k(x, x_j) - \sum_{x \in S_-} \sum_{j=1}^{m} \alpha_j y_j k(x, x_j) \Big). \tag{7.56}$$

Note that for the decision function, only b is actually required.

Connection ν-SVC — C-SVC

A connection to standard SV classification, and a somewhat surprising interpretation of the regularization parameter C, is described by the following result:

Proposition 7.6 (Connection ν-SVC — C-SVC [481]) *If ν-SV classification leads to $\rho > 0$, then C-SV classification, with C set a priori to $1/\rho$, leads to the same decision function.*

Proof If we minimize (7.40), and then fix ρ to minimize only over the remaining variables, nothing will change. Hence the solution $\mathbf{w}_0, b_0, \boldsymbol{\xi}_0$ minimizes (7.35), for $C = 1$, subject to (7.41). To recover the constraint (7.34), we rescale to the set of variables $\mathbf{w}' = \mathbf{w}/\rho, b' = b/\rho, \boldsymbol{\xi}' = \boldsymbol{\xi}/\rho$. This leaves us with the objective function (7.35), up to a constant scaling factor ρ^2, using $C = 1/\rho$. ∎

For further details on the connection between ν-SVMs and C-SVMs, see [122, 38]. A complete account has been given by Chang and Lin [98], who show that for a given problem and kernel, there is an interval $[\nu_{\min}, \nu_{\max}]$ of admissible values for ν, with $0 \leq \nu_{\min} \leq \nu_{\max} \leq 1$. The boundaries of the interval are computed by considering $\sum_i \alpha_i$ as returned by the C-SVM in the limits $C \to \infty$ and $C \to 0$, respectively.

It has been noted that ν-SVMs have an interesting interpretation in terms of *reduced convex hulls* [122, 38] (cf. (7.21)). If a problem is non-separable, the convex hulls will no longer be disjoint. Therefore, it no longer makes sense to search for the shortest line connecting them, and the approach of (7.22) will fail. In this situation, it seems natural to reduce the convex hulls in size, by limiting the size of the coefficients c_i in (7.21) to some value $\nu \in (0, 1)$. Intuitively, this amounts to limiting the influence of individual points — note that in the original problem (7.22), two single points can already determine the solution. It is possible to show that the ν-SVM formulation solves the problem of finding the hyperplane orthogonal to the closest line connecting the *reduced* convex hulls [122].

Robustness and Outliers

We now move on to another aspect of soft margin classification. When we introduced the slack variables, we did not attempt to justify the fact that in the objective function, we used a penalizer $\sum_{i=1}^m \xi_i$. Why not use another penalizer, such as $\sum_{i=1}^m \xi_i^p$, for some $p \geq 0$ [111]? For instance, $p = 0$ would yield a penalizer

that exactly *counts* the number of margin errors. Unfortunately, however, it is also a penalizer that leads to a combinatorial optimization problem. Penalizers yielding optimization problems that are particularly convenient, on the other hand, are obtained for $p = 1$ and $p = 2$. By default, we use the former, as it possesses an additional property which is statistically attractive. As the following proposition shows, linearity of the target function in the slack variables ξ_i leads to a certain "outlier" resistance of the estimator. As above, we use the shorthand $\mathbf{x}_i$ for $\Phi(x_i)$.

Proposition 7.7 (Resistance of SV classification [481]) *Suppose* $\mathbf{w}$ *can be expressed in terms of the SVs which are not at bound,*

$$\mathbf{w} = \sum_{i=1}^{m} \gamma_i \mathbf{x}_i \tag{7.57}$$

with $\gamma_i \neq 0$ *only if* $\alpha_i \in (0, 1/m)$ *(where the* α_i *are the coefficients of the dual solution). Then local movements of any margin error* $\mathbf{x}_m$ *parallel to* $\mathbf{w}$ *do not change the hyperplane.*[11]

The proof can be found in Section A.2. For further results in support of the $p = 1$ case, see [527].

Note that the assumption (7.57) is not as restrictive as it may seem. Even though the SV expansion of the solution, $\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \mathbf{x}_i$, often contains many multipliers α_i which are at bound, it is nevertheless quite conceivable, especially when discarding the requirement that the coefficients be bounded, that we can obtain an expansion (7.57) in terms of a subset of the original vectors.

For instance, if we have a 2-D problem that we solve directly in input space, i.e., with $k(x, x') = \langle x, x' \rangle$, then it suffices to have two linearly independent SVs which are not at bound, in order to express $\mathbf{w}$. This holds true regardless of whether or not the two classes overlap, even if there are many SVs which are at the upper bound. Further information on resistance and robustness of SVMs can be found in Sections 3.4 and 9.3.

We have introduced SVs as those training examples x_i for which $\alpha_i > 0$. In some cases, it is useful to further distinguish different types of SVs. For reference purposes, we give a list of different types of SVs (Table 7.2).

In Section 7.3, we used the KKT conditions to argue that in the hard margin case, the SVs lie exactly on the margin. Using an identical argument for the soft margin case, we see that in this instance, in-bound SVs lie on the margin (Problem 7.9).

Note that in the hard margin case, where $\alpha_{\max} = \infty$, every SV is an in-bound SV. Note, moreover, that for kernels that produce full-rank Gram matrices, such as the Gaussian (Theorem 2.18), in theory every SV is essential (provided there are no duplicate patterns in the training set).[12]

11. Note that the perturbation of the point is carried out in feature space. What it precisely corresponds to in input space therefore depends on the specific kernel chosen.
12. In practice, Gaussian Gram matrices usually have some eigenvalues that are close to 0.

Table 7.2 Overview of different types of SVs. In each case, the condition on the Lagrange multipliers α_i (corresponding to an SV x_i) is given. In the table, $\alpha_{\max}$ stands for the upper bound in the optimization problem; for instance, $\alpha_{\max} = \frac{C}{m}$ in (7.38) and $\alpha_{\max} = \frac{1}{m}$ in (7.50).

Type of SV	Definition	Properties
(standard) SV	$0 < \alpha_i$	lies on or in margin
in-bound SV	$0 < \alpha_i < \alpha_{\max}$	lies on margin
bound SV	$\alpha_i = \alpha_{\max}$	usually lies in margin ("margin error")
essential SV	appears in all possible expansions of solution	becomes margin error when left out (Section 7.3)

7.6 Multi-Class Classification

So far, we have talked about binary classification, where the class labels can only take two values: ± 1. Many real-world problems, however, have more than two classes — an example being the widely studied optical character recognition (OCR) problem. We will now review some methods for dealing with this issue.

7.6.1 One Versus the Rest

To get *M-class classifiers*, it is common to construct a set of binary classifiers $f^1, \ldots, f^M$, each trained to separate one class from the rest, and combine them by doing the multi-class classification according to the maximal output before applying the sgn function; that is, by taking

$$\operatorname*{argmax}_{j=1,\ldots,M} g^j(x), \quad \text{where } g^j(x) = \sum_{i=1}^{m} y_i \alpha_i^j k(x, x_i) + b^j \tag{7.58}$$

(note that $f^j(x) = \operatorname{sgn}(g^j(x))$, cf. (7.25)).

Reject Decisions

The values $g^j(x)$ can also be used for *reject decisions*. To see this, we consider the difference between the two largest $g^j(x)$ as a measure of confidence in the classification of x. If that measure falls short of a threshold θ, the classifier rejects the pattern and does not assign it to a class (it might instead be passed on to a human expert). This has the consequence that on the remaining patterns, a lower error rate can be achieved. Some benchmark comparisons report a quantity referred to as the *punt error*, which denotes the fraction of test patterns that must be rejected in order to achieve a certain accuracy (say 1% error) on the remaining test samples. To compute it, the value of θ is adjusted on the *test* set [64].

The main shortcoming of (7.58), sometimes called the *winner-takes-all* approach, is that it is somewhat heuristic. The binary classifiers used are obtained by training on different binary classification problems, and thus it is unclear whether their