ν-SVMs

In the traditional softmargin classification SVM formulation we have a penalty constant C such that

$$C \propto \frac{1}{\text{size of margin}}.$$

Furthermore, there is no *a priori* guidance as to what C should be set to - the default is a value of 1. However, the precise value needs to be determined experimentally.

ν-SVMs

Schölkopf *et al.* suggest an alternative formulation of ***softmargin SVMs based on the ν parameter***[a] with $\nu \in [0, 1]$.

The advantages of the ν ***parameter*** formulation are that it represents an ***upper bound on the fraction of number of margin errors*** allowed,

$\nu = .1 \quad \rightarrow \quad$ a max. of 10% of training set can be margin errors

$\nu = .8 \quad \rightarrow \quad$ a max. of 80% of training can be margin errors

and that it is proportional to the size of the margin,

$$\nu \propto \text{size of margin}$$

This implies that determining a value for ν is a more intuitive process that finding a value for the penalty constant C.

---

[a] B. Schölkopf, A. Smola, R. C. Williamson, and P. L. Bartlett. *New Support Vector Algorithms.* Neural Computation, 12:12071245, 2000.

ν-SVC

We can formulate the ν-SVC[a] problem in the primal version as follows,

$$\min_{\overline{w},\overline{\xi},\rho,b} \phi(\overline{w},\overline{\xi},\rho) = \frac{1}{2}\overline{w} \bullet \overline{w} - \nu\rho + \frac{1}{l}\sum_{i=1}^{l} \xi_i$$

$$\text{subject to} \quad y_i(\overline{w} \bullet \overline{x}_i - b) \geq \rho - \xi_i$$
$$\xi_i \geq 0$$
$$\rho \geq 0$$

Here $\overline{\xi}$ represents the set of slack variables as before.

Observations:

- We no longer have a constant margin of value 1, instead we consider the size of the margin an explicit optimization variable - ρ.
- Observe that if $\overline{\xi} = \overline{0}$ then the margin is $2\rho/\overline{w} \bullet \overline{w}$.
- We don't directly penalize the size of the margin errors, instead we penalize the size of the margin - term $\nu\rho$.

---

[a] ν-SVC means ν support vector classification.

Dual ν-SVC

The Lagrangian,

$$
\begin{aligned}
L(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \overline{\xi}, \rho, b) = \frac{1}{2}\overline{w} \bullet \overline{w} - \nu\rho + \frac{1}{l}\sum_{i=1}^{l} \xi_i \\
- \sum_{i=1}^{l} \alpha_i (y_i(\overline{w} \bullet \overline{x}_i - b) - \rho + \xi_i) \\
- \sum_{i=1}^{l} \beta_i \xi_i \\
- \delta\rho
\end{aligned}
$$

with $\overline{\alpha}_i, \overline{\beta}_i, \delta \geq 0$.

Where the optimization problem is

$$
\max_{\overline{\alpha}, \overline{\beta}, \delta} \min_{\overline{w}, \overline{\xi}, \rho, b} L(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \overline{\xi}, \rho, b).
$$

KKT Conditions

A solution $\overline{\alpha}^*, \overline{\beta}^*, \delta^*, \overline{w}^*, \overline{\xi}^*, b^*$, and ρ^* has to satisfy the KKT conditions,

$$\begin{aligned}
\frac{\partial L}{\partial \overline{w}}(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}^*, \overline{\xi}, \rho, b) &= 0, \\
\frac{\partial L}{\partial \xi_i}(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \xi_i^*, \rho, b) &= 0, \\
\frac{\partial L}{\partial \rho}(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \overline{\xi}, \rho^*, b) &= 0, \\
\frac{\partial L}{\partial b}(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \overline{\xi}, \rho, b^*) &= 0, \\
\alpha_i^*(y_i(\overline{w}^* \bullet \overline{x}_i - b^*) + \xi_i^* - \rho^*) &= 0, \\
\beta_i^* \xi_i^* &= 0, \\
\delta^* \rho^* &= 0, \\
y_i(\overline{w}^* \bullet \overline{x}_i - b^*) + \xi_i^* - \rho^* &\geq 0, \\
\alpha_i^* &\geq 0, \\
\beta_i^* &\geq 0, \\
\delta^* &\geq 0, \\
\xi_i^* &\geq 0,
\end{aligned}$$

for $i = 1, \ldots, l$.

Dual ν-SVC

Taking the partial derivatives of $L(\overline{\alpha}, \overline{\beta}, \delta, \overline{w}, \overline{\xi}, \rho, b)$ with respect to the primal variables and setting them to 0 we obtain,

$$\overline{w} = \sum_{i=1}^{l} \alpha_i y_i \overline{x}_i$$

$$\alpha_i + \beta_i = \frac{1}{l}$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0$$

$$\sum_{i=1}^{l} \alpha_i = \nu + \delta$$

Plugging these back into the Lagrangian gives us our dual optimization problem.

Dual ν-SVC

This gives us the a training algorithm for softmargin ν-SVC with the kernel $k(\overline{x}_i, \overline{x}_j)$ substituted for the dot product in input space,

$$\max_{\overline{\alpha}} \phi'(\overline{\alpha}) = \max_{\overline{\alpha}} \left(-\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j k(\overline{x}_i, \overline{x}_j) \right)$$

subject to the constraints,

$$\sum_{i=1}^{l} y_i \alpha_i = 0$$

$$\sum_{i=1}^{l} \alpha_i \geq \nu$$

$$1/l \geq \alpha_i \geq 0, i = 1, \ldots, l$$

Compared to the dual optimization problem of C-SVCs we have two differences: (a) we lost the term $\Sigma \alpha_i$ in the objective function and (b) we have an additional constraint due to ρ.

Dual ν-SVC

Turns out that our decision function stays the same as in the C classifiers,

$$\boxed{\hat{f}(\overline{x}) = \mathrm{sign}\left(\sum_{i=1}^{l} \alpha_i^* y_i k(\overline{x}_i, \overline{x}) - b^*\right).}$$

Here, as before, b^* can be computed from support vectors that are not bound, $0 < \alpha_i < 1/l$.

ν-SVC

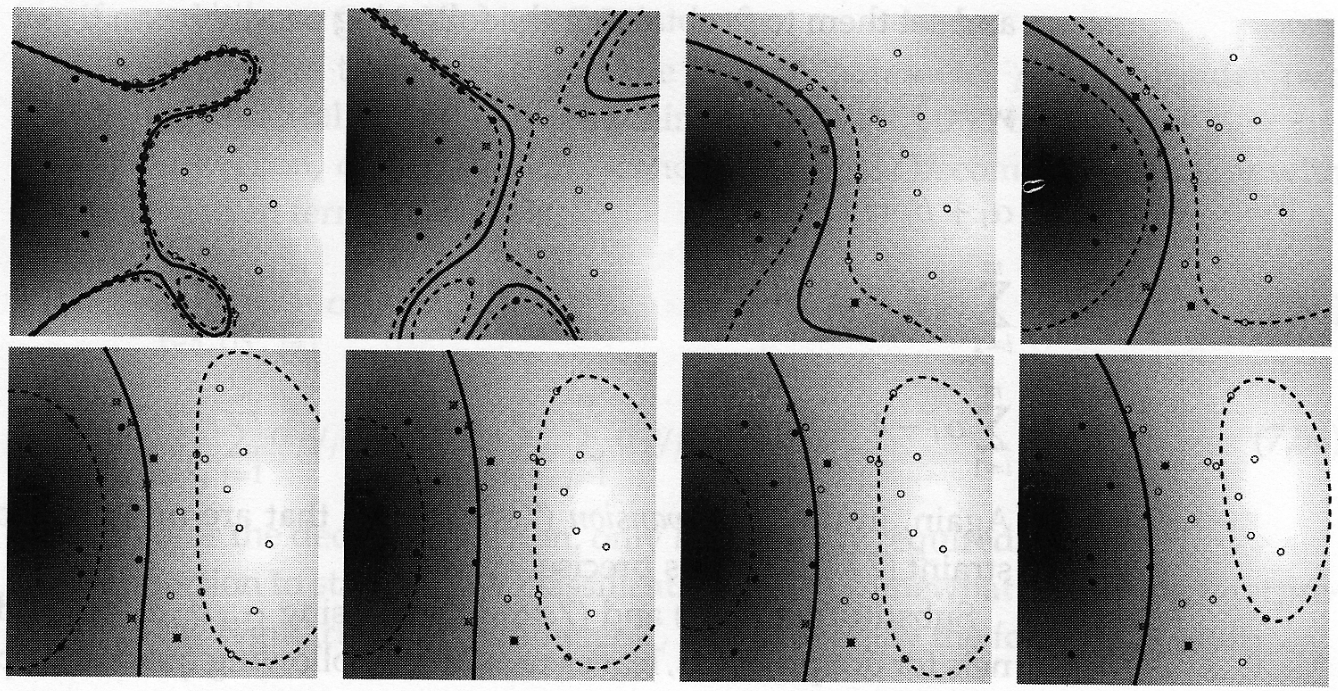

Figure 7.9 Toy problem (task: separate circles from disks) solved using ν-SV classification, with parameter values ranging from $\nu = 0.1$ (top left) to $\nu = 0.8$ (bottom right). The larger we make ν, the more points are allowed to lie inside the margin (depicted by dotted lines). Results are shown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

(source: "Learning with Kernels", Schölkopf and Smola, MIT, 2002)

ν-SVC

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

(source: "Learning with Kernels", Schölkopf and Smola, MIT, 2002)

ν-SVR

In ν-SVR we want to have our ε automatically computed. This gives rise to the following primal optimization problem

$$\min_{\overline{w},\overline{\xi},\overline{\xi}',\varepsilon,b} \phi(\overline{w},\overline{\xi},\overline{\xi}',\varepsilon,b) = \frac{1}{2}\overline{w}\bullet\overline{w} + C\cdot\left(\nu\varepsilon + \frac{1}{n}\sum_{i=1}^{n}(\xi_i+\xi_i')\right)$$

$$\text{subject to} \quad \begin{aligned} &(\overline{w}\bullet\overline{x}_i - b) - y_i \leq \varepsilon + \xi_i' \\ &y_i - (\overline{w}\bullet\overline{x}_i - b) \leq \varepsilon + \xi_i \\ &\xi_i \geq 0 \\ &\xi_i^* \geq 0 \\ &\varepsilon \geq 0 \end{aligned}$$

Notice that here the term $\nu\varepsilon$ determines how much the size of the ε tube contributes to the optimization problem.

Dual ν-SVR

This gives rise to the dual,

$$\max_{\overline{\alpha},\overline{\alpha}'} \phi'(\overline{\alpha},\overline{\alpha}') = \max_{\overline{\alpha},\overline{\alpha}'} \sum_{i=1}^{l}(\alpha_i - \alpha_i')y_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i - \alpha_i')(\alpha_j - \alpha_j')k(\overline{x}_i, \overline{x}_j)$$

subject to the constraints,

$$\sum_{i=1}^{l}(\alpha_i - \alpha_i') = 0$$

$$\sum_{i=1}^{l}(\alpha_i' + \alpha_i) \leq C \cdot \nu$$

$$C/l \geq \alpha_i, \alpha_i' \geq 0, i = 1, \ldots, l$$

Our model is,

$$\hat{f}(\overline{x}) = \sum_{i=1}^{l}(\alpha_i - \alpha_i')k(\overline{x}_i, \overline{x}) - b.$$

ν-SVR

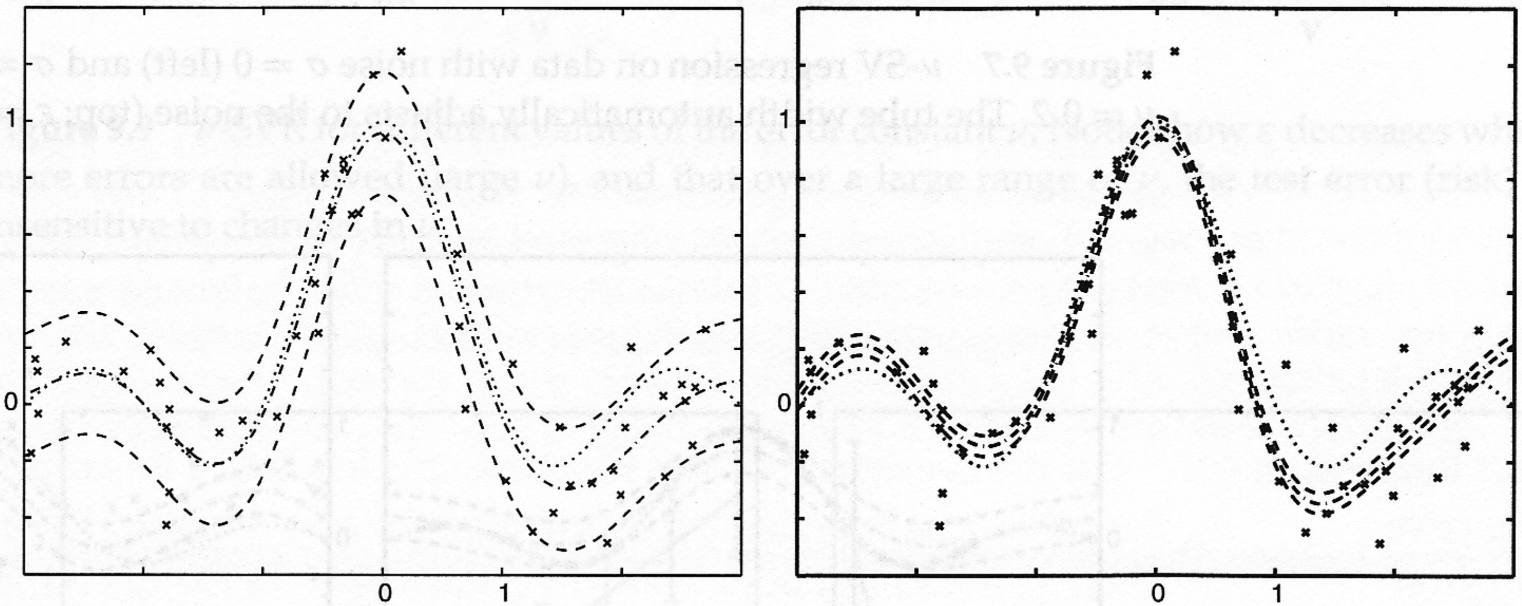

Figure 9.6 ν-SV regression with $\nu = 0.2$ (left) and $\nu = 0.8$ (right). The larger ν allows more points to lie outside the tube (see Section 9.3). The algorithm automatically adjusts ε to 0.22 (left) and 0.04 (right). Shown are the sinc function (dotted), the regression f and the tube $f \pm \varepsilon$.