Explore Slack Variables

Sometimes there is no separating hyperplane between the two classes. For example, points might be misclassified and could lie amidst points of the other class, in which case the data is not linearly separable. In such cases, there is no solution to the quadratic optimization problem we have discussed so far. If we still want to find a hyperplane that gets most of the points right, we can allow the constraints to be violated ever so slightly. Formally, we introduce slack variables positive scalars ξ_i that we add to the objective and constraints as follows:

$$\min_{\mathbf{w},b} \mathbf{w}^\top \mathbf{w} + C \sum_{i=1}^{n} \xi_i$$
$$\text{such that } \forall i, y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq 1 - \xi_i$$
$$\forall i, \xi_i \geq 0$$

Each slack variable ξ_i allows the input x_i to be closer to the hyperplane (or even be on the wrong side if $\xi_i > 1$). However, there is a penalty in the objective function for such "slack". If the hyperparameter C is very large, SVM becomes very strict and tries to get all points to be on the correct side of the hyperplane (by driving the slack variables ξ_i down to 0). If C is very small, the SVM becomes very loose and may "sacrifice" some points to obtain a simpler (i.e. lower $\mathbf{w}^\top \mathbf{w}$) solution. In other words, the slack variables give SVM some flexibility so that the algorithm still converges, even if there is no "true" optimal hyperplane.