SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

01.112 Machine Learning, Fall 2019
Lecture Notes for Week 4

8. Support Vector Machines (II)

Thursday, 10 October, 2019

1 Support Vector Machines (with Offset)

As we had discussed before, maximum margin linear separators can be found by solving the following primal quadratic programming problem

$$\text{(primal)} \qquad \min \frac{1}{2}\|\theta\|^2 \text{ subject to } y^{(t)}(\theta \cdot x^{(t)} + \theta_0) \geq 1 \tag{1}$$

If we wish to solve the problem in the dual, i.e., represent the parameters θ in terms of examples as $\theta = \sum_{t=1}^{n} \alpha_t y^{(t)} x^{(t)}$, we can solve the dual

$$\text{(dual)} \qquad \max \sum_{t=1}^{n} \alpha_t - \frac{1}{2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \alpha_t \alpha_{t'} y^{(t)} y^{(t')} (x^{(t)} \cdot x^{(t')})$$
$$\text{subject to } \alpha_t \geq 0, t = 1, \ldots, n \tag{2}$$

where the additional constraint $\sum_{t=1}^{n} \alpha_t y^{(t)} = 0$ pertains to including the offset parameter θ_0 in the primal. But θ_0 does not appear anywhere in the dual. How do we set it?

After we solve the dual, we can see that $\hat{\alpha}_t > 0$ for some examples (support vectors) and $\hat{\alpha}_t = 0$ for others (non-support vectors). Since for support vectors, the classification constraints must be satisfied with equality, we have that when $\hat{\alpha}_t > 0$,

$$y^{(t)}(\hat{\theta} \cdot x^{(t)} + \hat{\theta}_0) = y^{(t)}(\sum_{t'=1}^{n} \hat{\alpha}_{t'} y^{(t')} (x^{(t')} \cdot x^{(t)}) + \theta_0) = 1 \tag{3}$$

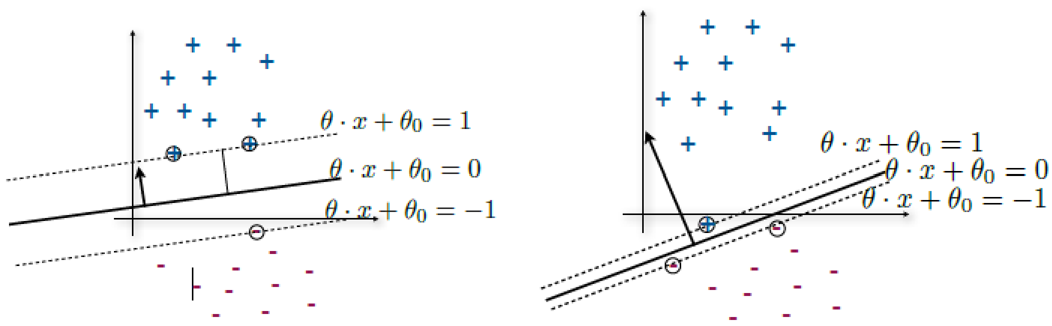
As $y^{(t)} \in \{-1, 1\}$ or $(y^{(t)})^2 = 1$ we can multiply both sides by $y^{(t)}$ and get

$$\hat{\theta}_0 = y^{(t)} - (\sum_{t'=1}^{n} \hat{\alpha}_{t'} y^{(t')} (x^{(t')} \cdot x^{(t)})) \tag{4}$$

This should hold for all support vectors. However, solving the dual quadratic programming problem numerically does introduce errors and relying on any particular constraint may be unwise. Instead, we can simply take the median of all $\hat{\theta}_0$ estimates from support vector constraints.

2 Support Vector Machines (with Errors)

If the labels for training examples contain errors, finding the maximum margin linear classifier may be problematic. The resulting decision boundary is potentially drastically affected by a single mislabeled point. Consider, for example, how the decision boundary changes by the single positive (perhaps mislabeled) point included in Figure 1b).



a) b)

Figure 1: The maximum margin separator is strongly affected by individual points

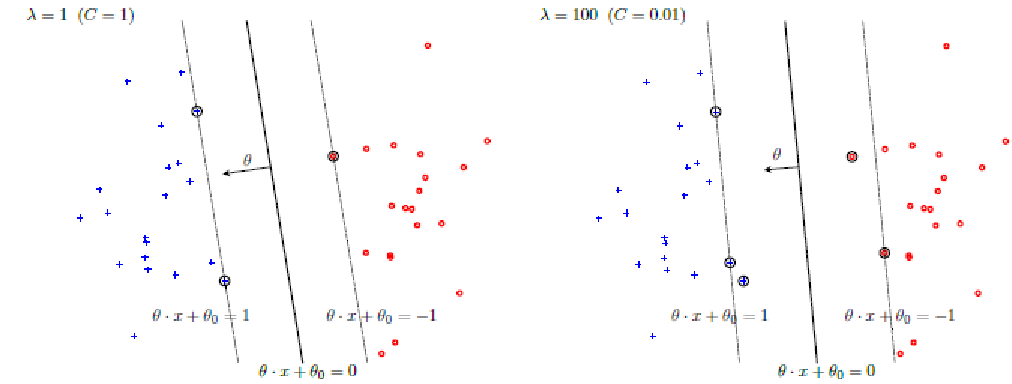
In order to remedy the situation, we should allow for misclassified points, yet maximize the margin. How do we achieve this? The simplest way is to treat the classification constraints as "soft" constraints rather than hard constraints. In other words, we permit them to be violated but specify the cost of such discrepancies in relation to increasing the margin (decreasing the norm $\|\theta\|$). Specifically, we will add "slack" variables $\xi_t \geq 0$ to the primal optimization problem as follows

$$\text{(primal)} \qquad \min \frac{\lambda}{2}\|\theta\|^2 + \sum_{t=1}^{n} \xi_t \tag{5}$$

$$\text{subject to } y^{(t)}(\theta \cdot x^{(t)} + \theta_0) \geq 1 - \xi_t, \xi_t \geq 0, t = 1, \ldots, n \tag{6}$$

We now minimize with respect to the parameters θ, θ_0 as well as the slack variables $\xi_t \geq 0$. Note that we had to introduce a parameter [1] λ (regularization parameter) to balance how much we favour increasing the margin over satisfying the classification constraints. Larger values of λ will push the margin boundaries and potentially the decision boundary past the examples. The margin boundaries are still defined as points satisfying $\theta \cdot x + \theta_0 = 1$ or $\theta \cdot x + \theta_0 = -1$. Figure 2 illustrates the choice of λ and the resulting maximum margin solutions when the examples are still linearly separable.

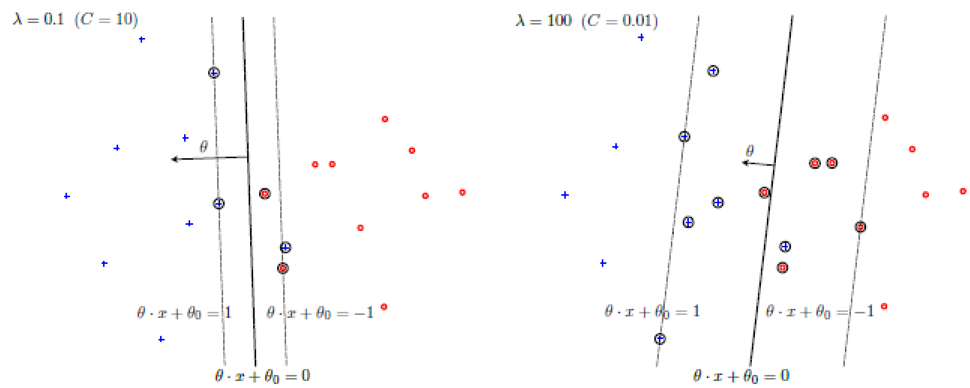
[1] In the literature you will often see a parameter C multiplying the sum of slack variables instead. This is the same formulation so long as $C = 1/\lambda$



a) b)

Figure 2: The effect of slack when examples are still linearly separable

The other advantage of the slack variables is that we can now solve problems that are no longer linearly separable. This is illustrated in Figure 3 with different values of the regularization parameter λ.



a) b)

Figure 3: The effect of slack when examples are no longer linearly separable

Our current formulation of the support vector machines can be written in a familiar form. Indeed, the primal version is exactly the same as minimizing

$$\frac{\lambda}{2}\,\|\theta\|^2 + \sum_{t=1}^{n} \text{Loss}_h(y^{(t)}(\theta \cdot x^{(t)} + \theta_0)) \tag{7}$$

with respect to θ and θ_0. Here $\text{Loss}_h(z) = \max\{1 - z, 0\}$ is the hinge loss. To see that these are equivalent, let us define

$$\xi_t = \text{Loss}_h(y^{(t)}(\theta \cdot x^{(t)} + \theta_0)) = \max\{1 - y^{(t)}(\theta \cdot x^{(t)} + \theta_0), 0\} \tag{8}$$

Clearly, $\xi_t \geq 0$ and

$$\xi_t \geq 1 - y^{(t)}(\theta \cdot x^{(t)} + \theta_0) \quad \text{or} \quad y^{(t)}(\theta \cdot x^{(t)} + \theta_0) \geq 1 - \xi_t \tag{9}$$

In other words, the slack variables are simply encoding the hinge loss in the primal formulation.

How will the dual formulation change in light of the slack variables in the primal? We will simply limit how large the Lagrange multipliers α_t can become. In other words, the adversary "gives up" on trying to further satisfy some of the classification constraints. More formally,

$$\begin{aligned} \text{(dual)} \qquad & \max \sum_{t=1}^{n} \alpha_t - \frac{1}{2} \sum_{t=1}^{n} \sum_{t'=1}^{n} \alpha_t \alpha_{t'} y^{(t)} y^{(t')} (x^{(t)} \cdot x^{(t')}) \\ & \text{subject to } 0 \leq \alpha_t \leq 1/\lambda, \sum_{t=1}^{n} \alpha_t y^{(t)} = 0 \end{aligned} \tag{10}$$

Note that the larger the value of λ is, i.e., the more we wish to increase the margin at the expense of the constraints, the smaller the resulting α_t must be.

We have to be a bit more careful now in reconstructing the offset parameter θ_0. Not all the support vectors, i.e., points with non-zero $\alpha's$, will lie exactly on the margin boundaries. Indeed, there will be points that lie on the wrong side of the margin boundaries or points that are misclassified altogether. The *complementary slackness* constraints that characterize the dual solution are now given by

$$\hat{\alpha}_t = 0 \Rightarrow y^{(t)}(\sum_{t'=1}^{n} \hat{\alpha}_{t'} y^{(t')} (x^{(t')} \cdot x^{(t)}) + \hat{\theta}_0) \geq 1 \qquad \text{(non-support vectors)} \tag{11}$$

$$\hat{\alpha}_t \in (0, 1/\lambda) \Rightarrow y^{(t)}(\sum_{t'=1}^{n} \hat{\alpha}_{t'} y^{(t')} (x^{(t')} \cdot x^{(t)}) + \hat{\theta}_0) = 1 \qquad \text{(SVs, on the margin)} \tag{12}$$

$$\hat{\alpha}_t = 1/\lambda \Rightarrow y^{(t)}(\sum_{t'=1}^{n} \hat{\alpha}_{t'} y^{(t')} (x^{(t')} \cdot x^{(t)}) + \hat{\theta}_0) \leq 1 \qquad \text{(SVs, margin violations)} \tag{13}$$

We can use the points for which α_t lies in the interior of the possible values, i.e., $\alpha_t \in (0, 1/\lambda)$, to reconstruct $\hat{\theta}_0$.

Learning Objective

You need to know:

1. What is the primal and dual form for SVM when we consider mis-classification errors
2. What is the definition of support vectors and how to check if a training instance is a support vector
3. What is the connection between SVM's training objective and hinge loss