Support Vector Machines

Max Welling
Department of Computer Science
University of Toronto
10 King's College Road
Toronto, M5S 3G5 Canada
welling@cs.toronto.edu

Abstract

This is a note to explain support vector machines.

1 Preliminaries

Our task is to predict whether a test sample belongs to one of two classes. We receive training examples of the form: $\{\mathbf{x}_i, y_i\}$, $i = 1, ..., n$ and $\mathbf{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, +1\}$. We call $\{\mathbf{x}_i\}$ the co-variates or input vectors and $\{y_i\}$ the response variables or labels.

We consider a very simple example where the data are in fact linearly separable: i.e. I can draw a straight line $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} - b$ such that all cases with $y_i = -1$ fall on one side and have $f(\mathbf{x}_i) < 0$ and cases with $y_i = +1$ fall on the other and have $f(\mathbf{x}_i) > 0$. Given that we have achieved that, we could classify new test cases according to the rule $y_{\text{test}} = \text{sign}(\mathbf{x}_{\text{test}})$.

However, typically there are infinitely many such hyper-planes obtained by small perturbations of a given solution. How do we choose between all these hyper-planes which the solve the separation problem for our training data, but may have different performance on the newly arriving test cases. For instance, we could choose to put the line very close to members of one particular class, say $y = -1$. Intuitively, when test cases arrive we will not make many mistakes on cases that should be classified with $y = +1$, but we will make very easily mistakes on the cases with $y = -1$ (for instance, imagine that a new batch of test cases arrives which are small perturbations of the training data). A sensible thing thus seems to choose the separation line as far away from both $y = -1$ and $y = +1$ training cases as we can, i.e. right in the middle.

Geometrically, the vector $\mathbf{w}$ is directed orthogonal to the line defined by $\mathbf{w}^T\mathbf{x} = b$. This can be understood as follows. First take $b = 0$. Now it is clear that all vectors, $\mathbf{x}$, with vanishing inner product with $\mathbf{w}$ satisfy this equation, i.e. all vectors orthogonal to $\mathbf{w}$ satisfy this equation. Now translate the hyperplane away from the origin over a vector $\mathbf{a}$. The equation for the plane now becomes: $(\mathbf{x} - \mathbf{a})^T\mathbf{w} = 0$, i.e. we find that for the offset $b = \mathbf{a}^T\mathbf{w}$, which is the projection of $\mathbf{a}$ onto to the vector $\mathbf{w}$. Without loss of generality we may thus choose $\mathbf{a}$ perpendicular to the plane, in which case the length $||\mathbf{a}|| = |b|/||\mathbf{w}||$ represents the shortest, orthogonal distance between the origin and the hyperplane.

We now define 2 more hyperplanes parallel to the separating hyperplane. They represent that planes that cut through the closest training examples on either side. We will call them

"support hyper-planes" in the following, because the data-vectors they contain support the plane.

We define the distance between the these hyperplanes and the separating hyperplane to be d_+ and d_- respectively. The *margin*, γ, is defined to be $d_+ + d_-$. Our goal is now to find a the separating hyperplane so that the margin is largest, while the separating hyperplane is equidistant from both.

We can write the following equations for the support hyperplanes:

$$\mathbf{w}^T\mathbf{x} = b + \delta \quad (1)$$

$$\mathbf{w}^T\mathbf{x} = b - \delta \quad (2)$$

We now note that we have over-parameterized the problem: if we scale $\mathbf{w}, b$ and δ by a constant factor α, the equations for $\mathbf{x}$ are still satisfied. To remove this ambiguity we will require that $\delta = 1$, this sets the scale of the problem, i.e. if we measure distance in millimeters or meters.

We can now also compute the values for $d_+ = (||b+1| - |b||)/||\mathbf{w}|| = 1/||\mathbf{w}||$ (this is only true if $b \notin (-1, 0)$ since the origin doesn't fall in between the hyperplanes in that case. If $b \in (-1, 0)$ you should use $d_+ = (||b + 1| + |b||)/||\mathbf{w}|| = 1/||\mathbf{w}||$). Hence the margin is equal to twice that value: $\gamma = 2/||\mathbf{w}||$.

With the above definition of the support planes we can write down the following constraint that any solution must satisfy,

$$\mathbf{w}^T\mathbf{x}_i - b \leq -1 \quad \forall\ y_i = -1 \quad (3)$$

$$\mathbf{w}^T\mathbf{x}_i - b \geq +1 \quad \forall\ y_i = +1 \quad (4)$$

or in one equation,

$$y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 \geq 0 \quad (5)$$

We now formulate the primal problem of the SVM:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}||\mathbf{w}||^2 \\ &\text{subject to} \quad y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 \geq 0\ \forall i \end{aligned} \quad (6)$$

Thus, we maximize the margin, subject to the constraints that all training cases fall on either side of the support hyper-planes. The data-cases that lie on the hyperplane are called support vectors, since they support the hyper-planes and hence determine the solution to the problem.

The primal problem can be solved by a quadratic program. However, it is not ready to be kernelised, because its dependence is not only on inner products between data-vectors. Hence, we transform to the dual formulation by first writing the problem using a Lagrangian,

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{N} \alpha_i \left[y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1\right] \quad (7)$$

The solution that minimizes the primal problem subject to the constraints is given by $\min_{\mathbf{w}} \max_{\boldsymbol{\alpha}} \mathcal{L}(\mathbf{w}, \boldsymbol{\alpha})$, i.e. a saddle point problem. When the original objective-function is convex, (and only then), we can interchange the minimization and maximization. Doing that, we find that we can find the condition on $\mathbf{w}$ that must hold at the saddle point we are solving for. This is done by taking derivatives wrt $\mathbf{w}$ and b and solving,

$$\mathbf{w} - \sum_i \alpha_i y_i \mathbf{x}_i = 0 \quad \Rightarrow \quad \mathbf{w}^* = \sum_i \alpha_i y_i \mathbf{x}_i \quad (8)$$

$$\sum_i \alpha_i y_i = 0 \quad (9)$$

Inserting this back into the Lagrangian we obtain what is known as the dual problem,

$$\begin{aligned} \text{maximize} \quad & \mathcal{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{subject to} \quad & \sum_i \alpha_i y_i = 0 \qquad (10) \\ & \alpha_i \geq 0 \ \ \forall i \qquad (11) \end{aligned}$$

The dual formulation of the problem is also a quadratic program, but note that the number of variables, α_i in this problem is equal to the number of data-cases, N.

The crucial point is however, that this problem *only depends on* $\mathbf{x}_i$ *through the inner product* $\mathbf{x}_i^T \mathbf{x}_j$. This is readily kernelised through the substitution $\mathbf{x}_i^T \mathbf{x}_j \rightarrow k(x_i, x_j)$. This is a recurrent theme: the dual problem lends itself to kernelisation, while the primal problem did not.

The theory of duality guarantees that for convex problems, the dual problem will be concave, and moreover, that the unique solution of the primal problem corresponds tot the unique solution of the dual problem. In fact, we have: $\mathcal{L}_P(\mathbf{w}^*) = \mathcal{L}_D(\alpha^*)$, i.e. the "duality-gap" is zero.

Next we turn to the conditions that must necessarily hold at the saddle point and thus the solution of the problem. These are called the KKT conditions (which stands for Karush-Kuhn-Tucker). These conditions are necessary in general, and sufficient for convex optimization problems. They can be derived from the primal problem by setting the derivatives wrt to $\mathbf{w}$ to zero. Also, the constraints themselves are part of these conditions and we need that for *inequality* constraints the Lagrange multipliers are non-negative. Finally, an important constraint called "complementary slackness" needs to be satisfied,

$$\begin{aligned} \partial_{\mathbf{w}} \mathcal{L}_P = 0 \ \ \rightarrow \qquad & \mathbf{w} - \sum_i \alpha_i y_i \mathbf{x}_i = 0 \qquad (12) \\ \partial_b \mathcal{L}_P = 0 \ \ \rightarrow \qquad & \sum_i \alpha_i y_i = 0 \qquad (13) \\ \text{constraint - 1} \qquad & y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 \geq 0 \qquad (14) \\ \text{multiplier condition} \qquad & \alpha_i \geq 0 \qquad (15) \\ \text{complementary slackness} \qquad & \alpha_i \left[y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 \right] = 0 \qquad (16) \end{aligned}$$

It is the last equation which may be somewhat surprising. It states that either the inequality constraint is satisfied, but not saturated: $y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 > 0$ in which case α_i for that data-case must be zero, or the inequality constraint is saturated $y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 = 0$, in which case α_i can be any value $\alpha_i \geq 0$. Inequality constraints which are saturated are said to be "active", while unsaturated constraints are inactive. One could imagine the process of searching for a solution as a ball which runs down the primary objective function using gradient descent. At some point, it will hit a wall which is the constraint and although the derivative is still pointing partially towards the wall, the constraints prohibits the ball to go on. This is an active constraint because the ball is glued to that wall. When a final solution is reached, we could remove some constraints, without changing the solution, these are inactive constraints. One could think of the term $\partial_{\mathbf{w}} \mathcal{L}_P$ as the force acting on the ball. We see from the first equation above that only the forces with $\alpha_i \neq 0$ exsert a force on the ball that balances with the force from the curved quadratic surface $\mathbf{w}$.

The training cases with $\alpha_i > 0$, representing active constraints on the position of the support hyperplane are called support vectors. These are the vectors that are situated in the support hyperplane and they determine the solution. Typically, there are only few of them, which people call a "sparse" solution (most α's vanish).

What we are really interested in is the function $f(\cdot)$ which can be used to classify future test cases,

$$f(\mathbf{x}) = \mathbf{w}^{*T}\mathbf{x} - b^* = \sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x} - b^* \tag{17}$$

As an application of the KKT conditions we derive a solution for b^* by using the complementary slackness condition,

$$b^* = \left(\sum_j \alpha_j y_j \mathbf{x}_j^T \mathbf{x}_i - y_i \right) \quad i \text{ a support vector} \tag{18}$$

where we used $y_i^2 = 1$. So, using any support vector one can determine b, but for numerical stability it is better to average over all of them (although they should obviously be consistent).

The most important conclusion is again that this function $f(\cdot)$ can thus be expressed solely in terms of inner products $\mathbf{x}_i^T \mathbf{x}_i$ which we can replace with kernel matrices $k(\mathbf{x}_i, \mathbf{x}_j)$ to move to high dimensional non-linear spaces. Moreover, since $\boldsymbol{\alpha}$ is typically very sparse, we don't need to evaluate many kernel entries in order to predict the class of the new input $\mathbf{x}$.

2 The Non-Separable case

Obviously, not all datasets are linearly separable, and so we need to change the formalism to account for that. Clearly, the problem lies in the constraints, which cannot always be satisfied. So, let's relax those constraints by introducing "slack variables", ξ_i,

$$\mathbf{w}^T \mathbf{x}_i - b \leq -1 + \xi_i \quad \forall \; y_i = -1 \tag{19}$$
$$\mathbf{w}^T \mathbf{x}_i - b \geq +1 - \xi_i \quad \forall \; y_i = +1 \tag{20}$$
$$\xi_i \geq 0 \;\; \forall i \tag{21}$$

The variables, ξ_i allow for violations of the constraint. We should penalize the objective function for these violations, otherwise the above constraints become void (simply always pick ξ_i very large). Penalty functions of the form $C(\sum_i \xi_i)^k$ will lead to convex optimization problems for positive integers k. For $k = 1, 2$ it is still a quadratic program (QP). In the following we will choose $k = 1$. C controls the tradeoff between the penalty and margin.

To be on the wrong side of the separating hyperplane, a data-case would need $\xi_i > 1$. Hence, the sum $\sum_i \xi_i$ could be interpreted as measure of how "bad" the violations are and is an upper bound on the number of violations.

The new primal problem thus becomes,

$$\text{minimize} \quad \mathcal{L}_P = \frac{1}{2}||\mathbf{w}||^2 + C \sum_i \xi_i$$
$$\text{subject to} \quad y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i \geq 0 \;\; \forall i \tag{22}$$
$$\xi_i \geq 0 \;\; \forall i \tag{23}$$

leading to the Lagrangian,

$$\mathcal{L}(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{1}{2}||\mathbf{w}||^2 + C \sum_i \xi_i - \sum_{i=1}^N \alpha_i \left[y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1 + \xi_i \right] - \sum_{i=1}^N \mu_i \xi_i \tag{24}$$

from which we derive the KKT conditions,

$$1.\partial_{\mathbf{w}}\mathcal{L}_P = 0 \;\rightarrow\; \mathbf{w} - \sum_i \alpha_i y_i \mathbf{x}_i = 0 \tag{25}$$

$$2.\partial_b \mathcal{L}_P = 0 \;\rightarrow\; \sum_i \alpha_i y_i = 0 \tag{26}$$

$$3.\partial_\xi \mathcal{L}_P = 0 \;\rightarrow\; C - \alpha_i - \mu_i = 0 \tag{27}$$

$$4.\text{constraint-1} \quad y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 + \xi_i \geq 0 \tag{28}$$

$$5.\text{constraint-2} \quad \xi_i \geq 0 \tag{29}$$

$$6.\text{multiplier condition-1} \quad \alpha_i \geq 0 \tag{30}$$

$$7.\text{multiplier condition-2} \quad \mu_i \geq 0 \tag{31}$$

$$8.\text{complementary slackness-1} \quad \alpha_i \left[y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 + \xi_i\right] = 0 \tag{32}$$

$$9.\text{complementary slackness-1} \quad \mu_i \xi_i = 0 \tag{33}$$

$$\tag{34}$$

From here we can deduce the following facts. If we assume that $\xi_i > 0$, then $\mu_i = 0$ (9), hence $\alpha_i = C$ (1) and thus $\xi_i = 1 - y_i(\mathbf{x}_i^T\mathbf{w} - b)$ (8). Also, when $\xi_i = 0$ we have $\mu_i > 0$ (9) and hence $\alpha_i < C$. If in addition to $\xi_i = 0$ we also have that $y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 = 0$, then $\alpha_i > 0$ (8). Otherwise, if $y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 > 0$ then $\alpha_i = 0$. In summary, as before for points not on the support plane and on the correct side we have $\xi_i = \alpha_i = 0$ (all constraints inactive). On the support plane, we still have $\xi_i = 0$, but now $\alpha_i > 0$. Finally, for data-cases on the wrong side of the support hyperplane the α_i max-out to $\alpha_i = C$ and the ξ_i balance the violation of the constraint such that $y_i(\mathbf{w}^T\mathbf{x}_i - b) - 1 + \xi_i = 0$.

Geometrically, we can calculate the gap between support hyperplane and the violating data-case to be $\xi_i/||\mathbf{w}||$. This can be seen because the plane defined by $y_i(\mathbf{w}^T\mathbf{x} - b) - 1 + \xi_i = 0$ is parallel to the support plane at a distance $|1 + y_i b - \xi_i|/||\mathbf{w}||$ from the origin. Since the support plane is at a distance $|1 + y_i b|/||\mathbf{w}||$ the result follows.

Finally, we need to convert to the dual problem to solve it efficiently and to kernelise it. Again, we use the KKT equations to get rid of $\mathbf{w}, b$ and ξ,

$$\text{maximize} \quad \mathcal{L}_D = \sum_{i=1}^N \alpha_i - \frac{1}{2}\sum_{ij} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\text{subject to} \quad \sum_i \alpha_i y_i = 0 \tag{35}$$

$$0 \leq \alpha_i \leq C \;\; \forall i \tag{36}$$

Surprisingly, this is almost the same QP is before, but with an extra constraint on the multipliers α_i which now live in a box. This constraint is derived from the fact that $\alpha_i = C - \mu_i$ and $\mu_i \geq 0$. We also note that it only depends on inner products $\mathbf{x}_i^T\mathbf{x}_j$ which are ready to be kernelised.