CS269: Machine Learning Theory
Lecture 15: Support Vector Machines
November 15, 2010

Lecturer: Jennifer Wortman Vaughan
Scribes: Nikhilesh Tadiparthi, Sritulasi Edpuganti, Karthika Mohan

In today's lecture we will begin discussing Support Vector Machines (aka SVMs).

1 Introduction

In the previous lectures, we saw that large margin classifiers can lead to low generalization error. One might ask if we can design an algorithm that explicitly maximizes the margin on our training data. This is the basic idea behind Support Vector Machines. We will see how to maximize the margin directly using techniques from convex optimization

Before we dive into SVMs, let's briefly review the idea behind convex optimization.

2 Standard form optimization

We say that an optimization problem is in standard form if we can write it as

$$\begin{aligned} \min_{\vec{w}} \quad & f(\vec{w}) \\ \text{s.t.} \quad & g_i(\vec{w}) \leq 0, \quad i = 1, ..., k \\ & h_i(\vec{w}) = 0, \quad i = 1, ..., \ell \end{aligned}$$

where f is convex, each of the functions g_i is convex, and each of the functions h_i is affine.[1] The constraints $g_i(\vec{w}) \leq 0, i = 1, ..., k$ are referred to as *inequality constraints*, while the constraints $h_i(\vec{w}) = 0, i = 1, ..., \ell$ are *equality constraints*.

Standard form optimization problems are nice because they can be solved efficiently using a variety of different techniques including the subgradient method, the interior point method, and the ellipsoid method. While we will not have time to discuss these techniques in detail in this class, anyone who is curious can refer to the Boyd and Vandenberghe book, which is available for free online.[2]

3 Maximizing the Margin via Optimization

We know that the linear threshold function is represented by $y = \text{sign}(\vec{w} \cdot \vec{x} + b)$, where $\vec{w}$ is the weight vector and b is the intercept (signifying that the threshold function may not pass through the origin).

[1] The function $h_i(\vec{w})$ is said to be affine if it can be written as $h_i(\vec{w}) = \vec{z} \cdot \vec{w} + b$

[2] Refer book: `http://www.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf`

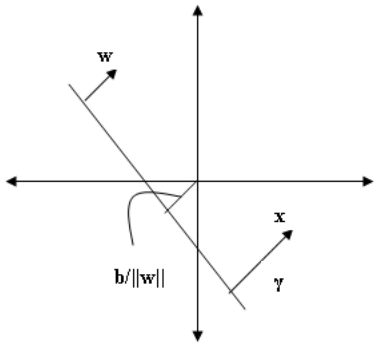

Figure 1: Depiction of a typical 2 dimensional threshold function and margin

Recall from Lecture 6 that the margin for the point x when $\| \vec{w} \| = 1$ is given by

$$\gamma_i = y_i(\vec{w} \cdot \vec{x} + b)$$

It is simple to show that when $\| \vec{w} \| \neq 1$ the margin for point x is given by

$$\gamma_i = y_i(\frac{\vec{w}}{\| \vec{w} \|} \cdot \vec{x} + \frac{b}{\| \vec{w} \|})$$

This derivation of this expression is very similar to derivation for the normalized version. From the previous lecture on Adaboost, we know that large margin in linear threshold functions implies small generalization error. Intuitively, the larger the value of margin of a point, the higher is the confidence that the point belongs to a certain class.

Assume we have m separable points $(\vec{x}_1, y_1), (\vec{x}_2, y_2), \ldots (\vec{x}_m, y_m)$. Given these training samples, we would like to find a classifier that maximizes the value of the margin. We discuss several attempts to formulate this problem as a standard form convex optimization problem below.

3.1 Attempt 1

First, note that for any constant $k > 0$, sign$(\vec{w} \cdot \vec{x} + b) =$ sign$(k\vec{w} \cdot \vec{x} + kb)$. This implies that any linear threshold function has a equivalent representation with $\| \vec{w} \| = 1$. We might as well try to explicitly restrict $\| \vec{w} \| = 1$ with our constraints so that we can use the simpler definition of margin.

We can formalize our optimization problem as:

$$\begin{aligned} \max_{\gamma, \vec{w}, b} \quad & \gamma \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x_i} + b) \geq \gamma, \quad i = 1, ..., k \\ & \| \vec{w} \| = 1 \end{aligned}$$

Unfortunately, the equality constraint $\| \vec{w} \| = 1$ is not affine, so this is not a standard form optimization problem and may be difficult to solve. So, we try a different approach and proceed on to the next attempt.

3.2 Attempt 2

Let's drop the constraint that $\| \vec{w} \| = 1$ and work with the more complicated definition of margin. Note that requiring the value of margin to be at least γ is equivalent to requiring $y_i(\vec{w} \cdot \vec{x_i} + b) \geq \gamma \| \vec{w} \|, \forall i$. We can therefore formalize the above optimization problem as:

$$\begin{aligned} \max_{\gamma, \vec{w}, b} \quad & \gamma \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x_i} + b) \geq \gamma \| \vec{w} \|, \quad i = 1, ..., k \end{aligned}$$

We see that the term $\| \vec{w} \|$ is still present in the inequality constraint. It turns out that we can reformulate this optimization problem in a clever way to get rid of this annoying constraint. To do this, we first replace $\gamma ||\vec{w}||$ with a new variable $\hat{\gamma}$, and rewrite the problem in terms of this variable:

$$\begin{aligned} \max_{\gamma, \vec{w}, b} \quad & \frac{\hat{\gamma}}{\| \vec{w} \|} \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x_i} + b) \geq \hat{\gamma}, \quad i = 1, ..., k \end{aligned}$$

Now, suppose that $\hat{\gamma} = \gamma^\star, \vec{w} = \vec{w}^\star$ and $b = b^\star$ is a solution to this problem. It can be easily verified that $\hat{\gamma} = 1, \vec{w} = \frac{\vec{w}^\star}{\gamma^\star}$ and $b = \frac{b^\star}{\gamma^\star}$ must *also* be a solution. Furthermore, since we have seen earlier that scaling the values of $\vec{w}$ and b in the term $\vec{w} \cdot \vec{x} + b$ does not affect the final classification result, we know that this solution corresponds to the *same* threshold. Using this, we can reformulate our optimization problem, removing $\hat{\gamma}$ entirely, without missing any potential solutions. We get:

$$\begin{aligned} \max_{\vec{w}, b} \quad & \frac{1}{\| \vec{w} \|} \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x_i} + b) \geq 1, \quad i = 1, ..., k \end{aligned}$$

which we know will have the same solutions as:

$$\begin{aligned} \min_{\vec{w}, b} \quad & \frac{1}{2} \cdot \| \vec{w} \|^2 \\ \text{s.t.} \quad & y_i(\vec{w} \cdot \vec{x_i} + b) \geq 1, \quad i = 1, ..., k \end{aligned}$$

We now have the optimization problem in the form we want! Furthermore, since the objective function is quadratic and we have linear inequality constraints, this convex optimization problem could be solved using any standard Quadratic Programming technique.

If we wanted, we could stop at this point and call ourselves done. However, it turns out that the dual of this problem satisfies a number of nice properties (in particular, it is easy to solve, and will allow us to use the "kernel trick" which will be described in the next lecture). So we go on and find the dual of the problem.

4 A Primer on Convex Optimization

To find the dual of this optimization problem (which we will attempt in the next class) we will need to review some key ideas from convex optimization.

Consider the generic standard form of optimization problem we defined earlier:

$$\begin{aligned} \min_{\vec{w}} \quad & f(\vec{w}) \\ \text{s.t.} \quad & g_i(\vec{w}) \leq 0, \quad i = 1, ..., k \\ & h_i(\vec{w}) = 0, \quad i = 1, ..., \ell \end{aligned}$$

The *Lagrangian* for this problem is defined as:

$$L(\vec{w}, \vec{\alpha}, \vec{\beta}) = f(\vec{w}) + \sum_{i=1}^{k} \alpha_i g_i(\vec{w}) + \sum_{i=1}^{l} \beta_i h_i(\vec{w})$$

where $\vec{\alpha}$ and $\vec{\beta}$ are called the Lagrange multipliers or KKT Multipliers. The reason that the Lagrangian is useful is that solving the above optimization problem is actually the *same* as finding the maximum value of the Lagrangian, where the maximization is over $\vec{\alpha} \geq 0$ and β. To see this, first verify that when the constraints are met, the Lagrangian is maximized when we set each α_i to 0 and each β_i to any arbitrary value. In this case, the second and third terms of the Lagrangian are 0, and we're left with just $f(\vec{w})$. You should also be able to verify that the maximum value is infinity if the constraints are not satisfied. That is,

$$\max_{\vec{\alpha}, \vec{\beta}: \alpha_i \geq 0} L(\vec{w}, \vec{\alpha}, \vec{\beta}) = \begin{cases} f(\vec{w}) & : \text{ if constraints are satisfied} \\ \infty & : \text{ otherwise} \end{cases}$$

Because of this, the standard form optimization problem that we care about is *equivalent to* the following **primal** optimization problem:

$$\min_{\vec{w}} \max_{\vec{\alpha}, \vec{\beta}: \alpha_i \geq 0} L(\vec{w}, \vec{\alpha}, \vec{\beta})$$

To define the **dual** optimization problem, we switch the $\min$ and $\max$:

$$\max_{\vec{\alpha}, \vec{\beta}: \alpha_i \geq 0} \min_{\vec{w}} L(\vec{w}, \vec{\alpha}, \vec{\beta})$$

Let's see how these problems are related. Let $p^\star$ be the value of primal problem and $d^\star$ be the value of dual problem. In general, we know that $p^\star \geq d^\star$. However, if f and the g_i are convex, the h_i are affine, and $\forall i$ $\exists \vec{w} : g_i(\vec{w}) < 0$, then it turns out that $p^\star = d^\star = L(\vec{w}^\star, \vec{\alpha}^\star, \vec{\beta}^\star)$ for some optimal solution $\vec{w}^\star, \vec{\alpha}^\star, \vec{\beta}^\star$. In this case, to obtain a solution to the primal problem, we can solve the dual instead.

4.1 KKT Conditions

Under the same set of conditions on the optimization problem, $\vec{w}^\star$, $\vec{\alpha}^\star$ and $\vec{\beta}^\star$ are a solution if and only if they satisfy the **KKT conditions:**

- Stationarity:

$$\frac{\partial}{\partial w_i} L(\vec{w}^\star, \vec{\alpha}^\star, \vec{\beta}^\star) = 0 \quad i = 1, ..., n$$

- Primal Feasibility:

$$\begin{aligned} h_i(\vec{w}^\star) &= 0 \quad i = 1, ..., l \\ g_i(\vec{w}^\star) &\leq 0 \quad i = 1, ..., k \end{aligned}$$

- Dual Feasibility:

$$\vec{\alpha}^\star \geq 0 \quad i = 1, ..., k$$

- Complementary Slackness:

$$\alpha_i^\star \cdot g_i(\vec{w}^\star) = 0 \quad i = 1, ..., k$$

In the next lecture, we will use these ideas to find the dual of our minimization problem, which will be the Support Vector Machine algorithm.