COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire | Lecture # 12
Scribe: Indraneel Mukherjee | March 12, 2008

In the previous lecture, we were introduced to the SVM algorithm and its basic motivation for use in classification tasks. In this lecture we will see how to actually compute a largest margin classifier, and touch upon how to lift the restrictive assumption of linear separability of the data.

Review of SVM

Recall that SVM's try to find a large margin linear classifier for data labeled + or − lying in $\mathbb{R}^n$. Formally, if we have m examples $\mathbf{x}_i \in \mathbb{R}^n$ with label y_i, then the SVM algorithm outputs $\mathbf{v} \in \mathbb{R}^n$ satisfying the following program:

$$\begin{aligned} \max \quad & \delta \\ s.t. \quad & \|\mathbf{v}\| = 1 \\ \forall i: \quad & y_i \left(\mathbf{v} \cdot \mathbf{x}_i\right) \geq \delta \end{aligned} \tag{1}$$

SVM's are designed to explicitly maximize the margin δ, whereas boosting happens to do so accidentally. This indicates that the two algorithms may be related, and a summary of their similarities and differences is tabularized below. The column labeled **Boosting** requires explanation. The set of all weak hypotheses are denoted by $h_1, \ldots$. We replace each example x by the vector of predictions of the weak hypotheses $h_1(x), \ldots,$; these are its only relevant features for boosting. The final hypothesis output takes a weighted majority vote of the different weak hypotheses. These non-negative weights, scaled down to sum to 1, are denoted by $a_1, \ldots$.

	SVM	**Boosting**
example	$\mathbf{x} \in \mathbb{R}^n$ $\|x\|_2 \leq 1$	$\mathbf{h}(\mathbf{x}) = \langle h_1(x), \ldots \rangle$ $\|\mathbf{h}(x)\|_\infty \stackrel{\Delta}{=} \max_j \|h_j(x)\| = 1$
finds	$\mathbf{v} \in \mathbb{R}^n$ $\|\mathbf{v}\|_2 = 1$	weights $\mathbf{a} = \langle a_1, \ldots \rangle$ on weak hyp $a_i \geq 0, \sum_i a_i = 1 \implies \|\mathbf{a}\|_1 = 1$
predicts	sign $(\mathbf{v} \cdot \mathbf{x})$	sign $\left(\sum_j a_j h_j(x)\right)$ = sign $(\mathbf{a} \cdot \mathbf{h}(x))$
margin	$y\,(\mathbf{v} \cdot \mathbf{x})$	$y \sum_j a_j h_j(x) = y\,(\mathbf{a} \cdot \mathbf{h}(x))$

Computing the SVM hypothesis

We describe how to solve the optimization problem given in (1). We begin by rewriting (1) as follows

$$\begin{aligned} \max \quad & \delta \\ s.t. \quad & \|\mathbf{v}\| = 1 \\ \forall i: \quad & y_i \left(\frac{\mathbf{v}}{\delta} \cdot \mathbf{x}_i\right) \geq 1. \end{aligned} \tag{2}$$

Letting $\mathbf{w} \triangleq \frac{\mathbf{v}}{\delta}$, we get the relation $\|\mathbf{w}\| = \frac{1}{\delta}$ (using $\|\mathbf{v}\| = 1$). Hence our objective is to minimize $\mathbf{w}$ subject to the SVM constraints $b_i(\mathbf{w}) \geq 0$, where $b_i(\mathbf{w}) \triangleq y_i(\mathbf{w} \cdot \mathbf{x}_i) - 1$. Hence our task reduces to solving

$$\begin{aligned} \min \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ s.t.\ \forall i: \quad & y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1. \end{aligned} \tag{3}$$

Following standard techniques, we form the Lagrangean of the above optimization problem by linearly combining the objective function and the constraints

$$L(\mathbf{w}, \boldsymbol{\alpha}) \triangleq \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_i \left[y_i(\mathbf{w} \cdot \mathbf{x}_i) - 1\right]. \tag{4}$$

The Lagrangean is useful because it converts the constrained optimization task in (3) to the following unconstrained one:

$$\min_{\mathbf{w}} \max_{\boldsymbol{\alpha} \geq \mathbf{0}} L(\mathbf{w}, \boldsymbol{\alpha}). \tag{5}$$

Indeed, (5) is the value of a game with two players, Mindy and Max, where Mindy goes first, choosing $\mathbf{w} \in \mathbb{R}^n$, and Max, observing Mindy's choice $\mathbf{w}$, selects $\boldsymbol{\alpha} \in \mathbb{R}^n_+$ to maximize the resulting value of (5); Mindy, aware of Max's strategy, makes her initial choice to minimize (5). If Mindy's choice of $\mathbf{w}$ violated any constraint $b_i(\mathbf{w}) \geq 0$, Max could choose α_i sufficiently large to make (5) unbounded. If no $\mathbf{w}$ obeying all the constraints in (3) existed, both (3), (5) would be ∞. Otherwise, Mindy ensures $b_i(\mathbf{w}) \geq 0$ for each i, and Max chooses $\alpha_i = 0$ whenever $b_i(\mathbf{w})$ were positive, so that $\alpha_i b_i(\mathbf{w}) = 0$ for every i. Thus, for $\mathbf{w}$ obeying all constraints, $L(\mathbf{w}, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_i \alpha_i b_i(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2$, and Mindy's strategy boils down to solving (3).

Some minimax theory

Before computing (5), we consider the *dual* game where Max goes first. Even if Mindy ignores Max's move and plays the same w in the dual as she would in the primal, she would ensure that the value of the dual does not exceed that of the primal. It follows

$$\max_{\boldsymbol{\alpha} \geq \mathbf{0}} \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}) \leq \min_{\mathbf{w}} \max_{\boldsymbol{\alpha} \geq \mathbf{0}} L(\mathbf{w}, \boldsymbol{\alpha}).$$

Minimax theory tells us, for a large class of functions L, the values of both games are in fact equal. One such class of immediate relevance to us is where both arguments of L belong to a convex domain, L is convex in its first argument, and concave in the second. The Lagrangean formed in (4) has these properties, and hence equality holds for our games.

Let $\mathbf{w}^*, \boldsymbol{\alpha}^*$ be optimal choices of Mindy and Max for the primal and dual games, resp.

$$\mathbf{w}^* \triangleq \arg\min_{\mathbf{w}} \max_{\boldsymbol{\alpha} \geq 0} L(\mathbf{w}, \boldsymbol{\alpha}) \tag{6}$$

$$\boldsymbol{\alpha}^* \triangleq \arg\max_{\boldsymbol{\alpha} \geq 0} \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}) \tag{7}$$

We can derive the following chain of inequalities

$$\begin{aligned}
L(\mathbf{w}^*, \boldsymbol{\alpha}^*) &\leq \max_{\boldsymbol{\alpha} \geq 0} L(\mathbf{w}^*, \boldsymbol{\alpha}) \\
&= \min_{\mathbf{w}} \max_{\boldsymbol{\alpha} \geq 0} L(\mathbf{w}, \boldsymbol{\alpha}) \text{ (by (6))} \\
&= \max_{\boldsymbol{\alpha} \geq 0} \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}) \text{ (minimax theory)} \\
&= \min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}^*) \text{ (by (7))} \\
&\leq L(\mathbf{w}^*, \boldsymbol{\alpha}^*).
\end{aligned}$$

Since the first and last terms are the same, equality holds on all lines. As a consequence we obtain the following facts

$$\begin{aligned}
\mathbf{w}^* &= \arg\min_{\mathbf{w}} L(\mathbf{w}, \boldsymbol{\alpha}^*) & (8) \\
\boldsymbol{\alpha}^* &= \arg\max_{\boldsymbol{\alpha}} L(\mathbf{w}^*, \boldsymbol{\alpha}) & (9)
\end{aligned}$$

showing that $(\mathbf{w}^*, \boldsymbol{\alpha}^*)$ is a *saddle point* of the function L. Since $L(\cdot, \boldsymbol{\alpha})$ is convex, the value of $\mathbf{w}$ minimizing $L(\mathbf{w}, \boldsymbol{\alpha})$ for a fixed $\boldsymbol{\alpha}$ is obtained by setting derivatives to zero. Eq (8) now implies

$$\forall j : \frac{\partial L(\mathbf{w}^*, \boldsymbol{\alpha}^*)}{\partial w_j} = 0. \tag{10}$$

From our previous discussions and (6), we know that $\mathbf{w}^*$ obeys all constraints, and $\alpha_i^* b_i(\mathbf{w})$ is always zero

$$\begin{aligned}
\forall i : \quad & b_i(\mathbf{w}^*) \geq 0 \\
& \alpha_i^* b_i(\mathbf{w}^*) = 0. & (11)
\end{aligned}$$

Conditions (10) and (11), together with the nonnegativity constraints $\alpha_i^* \geq 0$ are known as the Karush, Kuhn, Tucker (KKT) conditions, and they characterize all optimal solutions. Note that our discussions show that any optimal solution satisfies the KKT conditions. Showing that the converse holds will be a homework problem.

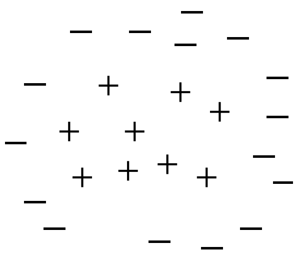
We return to solving the optimization task for SVMs. Recall that it suffices to compute the value of the dual game. As discussed before, the value of $\mathbf{w}$ minimizing $L(\mathbf{w}, \boldsymbol{\alpha})$ for fixed $\boldsymbol{\alpha}$ can be obtained by setting the derivative to zero:

$$\begin{aligned}
\forall j : \quad & \frac{\partial L}{\partial w_j} = w_j - \sum_i \alpha_i y_i x_i j = 0 \\
\Longrightarrow \quad & \mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i. & (12)
\end{aligned}$$

Plugging the expression for $\mathbf{w}$ into L, the value of the game is given by

$$\begin{aligned}
\max \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \left(\mathbf{x}_i \cdot \mathbf{x}_j\right) \\
s.t.\ \forall i : \quad & \alpha_i \geq 0
\end{aligned}$$

Figure 1: Data in $\mathbb{R}^2$ that is not linearly separable



The above program can be solved by standard hill-climbing techniques which will not be discussed. If $\boldsymbol{\alpha}^*$ solves the above program, (10) and (12) imply that $\mathbf{w}^* = \sum_i \alpha_i^* y_i \mathbf{x}_i$ is the SVM hypothesis. The KKT condition (11) implies that α_i^* is non-zero only when $y_i(\mathbf{w}^* \cdot \mathbf{x}_i) = 1$ i.e., $(\mathbf{x}_i, y_i)$ is a support vector. Hence the SVM hypothesis is a linear combination of only the support vectors. By a homework problem, if there are k support vectors, the generalization error of the classifier found by SVM is bounded by $O\left(\frac{k \ln m}{m}\right)$ with high probability. This gives an alternative method of analyzin SVM's.

SVM with non-linear classifiers

If the data $\mathbf{x}_{1:m}$ was not linearly separable, then no $\mathbf{w} \in \mathbb{R}^n$ satisfying the constraints in (1) would exist and the SVM algorithm would fail completely. To allow some noisy data, the constraints are often relaxed by a small amount ξ_i, and the objective function penalized by the net deviation $\sum_i \xi_i$ from the constraints. This gives rise to the soft margins SVM:

$$\begin{array}{rl} \min & \frac{1}{2}\|\mathbf{w}\|^2 + \sum_i \xi_i \\ s.t.\ \forall i: & y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{array}$$

Soft margins SVMs are useful when the data are inherently linearly separable but the labels are perturbed by some small amount of noise, a case often arising in practice. However, for some kinds of data (see figure 1) only higher dimensional surfaces can classify accurately. The basic SVM theory can still be applied, but the data has to be mapped to a higher dimensional space first. For example, we can take all possible monomial terms up to a certain degree. To illustrate, a data point $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ can be mapped to $\mathbb{R}^6$ as follows

$$\mathbf{x} = (x_1, x_2) \mapsto \psi(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2).$$

A hyperplane in the new space assumes the form

$$a + bx_1 + cx_2 + dx_1^2 + ex_2^2 + fx_1x_2 = 0$$

which is a degree 2 polynomial. This corresponds to a conic section (such as a circle, parabola, etc.) in the original space, which can correctly classify the data in figure 1. When considering all surfaces up to degree d, the dimension of a mapped example blows up as $O(n^d)$, where n was the original dimension. This might create severe computational problems since naively storing and operating the projected examples will require huge space and time complexity. Further, the high descriptive complexity of a linear classifier in the expanded space might pose problems of overfitting.

SVM's succesfully bypass both problems. The statistical problem is overcome by the fact that the optimal classifier is still given by a few support vectors; or alternatively, that the VC-dimension of a space of classifiers with large margin does not grow with the dimension of the space. Computational complexity is kept low via what is known as the kernel trick, which will be described after the spring break.