Lecture topics:

- Support vector machine and kernels
- Kernel optimization, selection

Support vector machine revisited

Our task here is to first turn the support vector machine into its *dual form* where the examples only appear in inner products. To this end, assume we have mapped the examples into feature vectors $\phi(\mathbf{x})$ of dimension d and that the resulting training set $(\phi(\mathbf{x}_1), y_1), \ldots, (\phi(\mathbf{x}_n), y_n)$ is linearly separable. Finding the maximum margin linear separator in the feature space now corresponds to solving

$$\text{minimize } \|\theta\|^2/2 \text{ subject to } y_t(\theta^T\phi(\mathbf{x}_t) + \theta_0) \geq 1, \quad t = 1, \ldots, n \tag{1}$$

We will discuss later on how slack variables affect the resulting kernel (dual) form. They merely complicate the derivation without changing the procedure. Optimization problems of the above type (convex, linear constraints) can be turned into their *dual* form by means of Lagrange multipliers. Specifically, we introduce a non-negative scalar parameter α_t for each inequality constraint and cast the estimation problem in terms of θ and $\alpha = \{\alpha_1, \ldots, \alpha_n\}$:

$$J(\theta, \theta_0; \alpha) = \|\theta\|^2/2 - \sum_{t=1}^{n} \alpha_t \Big[y_t(\theta^T\phi(\mathbf{x}_t) + \theta_0) - 1 \Big] \tag{2}$$

The original minimization problem for θ and θ_0 is recovered by *maximizing* $J(\theta, \theta_0; \alpha)$ with respect to α. In other words,

$$J(\theta, \theta_0) = \max_{\alpha \geq 0} J(\theta, \theta_0; \alpha) \tag{3}$$

where $\alpha \geq 0$ means that all the components α_t are non-negative. Let's try to see first that $J(\theta, \theta_0)$ really is equivalent to the original problem. Suppose we set θ and θ_0 such that at least one of the constraints, say the one corresponding to $(\mathbf{x}_i, y_i)$, is violated. In that case

$$-\alpha_i \Big[y_i(\theta^T\phi(\mathbf{x}_i) + \theta_0) - 1 \Big] > 0 \tag{4}$$

for any $\alpha_i > 0$. We can then set $\alpha_i = \infty$ to obtain $J(\theta, \theta_0) = \infty$. You can think of the Lagrange multipliers playing an adversarial role to enforce the margin constraints. More

formally,

$$J(\theta, \theta_0) = \begin{cases} \|\theta\|^2/2 \text{ if } y_t(\theta^T \phi(\mathbf{x}_t) + \theta_0) \geq 1, \quad t = 1, \ldots, n \\ \infty, \text{ otherwise} \end{cases} \tag{5}$$

So the minimizing θ and θ_0 are therefore those that satisfy the constraints. On the basis of a general set of criteria governing the optimality when dealing with Lagrange multipliers, criteria known as *Slater conditions*, we can actually switch the maximizing over α and the minimization over $\{\theta, \theta_0\}$ and get the same answer:

$$\min_{\theta,\theta_0} \max_{\alpha \geq 0} J(\theta, \theta_0; \alpha) = \max_{\alpha \geq 0} \min_{\theta,\theta_0} J(\theta, \theta_0; \alpha) \tag{6}$$

The left hand side, equivalent to minimizing Eq.(5), is known as the *primal form*, while the right hand side is the *dual form*. Let's solve the right hand side by first obtaining θ and θ_0 as a function of the Lagrange multipliers (and the data). To this end

$$\frac{d}{d\theta_0} J(\theta, \theta_0; \alpha) = -\sum_{t=1}^{n} \alpha_t y_t = 0 \tag{7}$$

$$\frac{d}{d\theta} J(\theta, \theta_0; \alpha) = \theta - \sum_{t=1}^{n} \alpha_t y_t \phi(\mathbf{x}_t) = 0 \tag{8}$$

So, again the solution for θ is in the span of the feature vectors corresponding to the training examples. Substituting this form of the solution for θ back into the objective, and taking into account the constraint corresponding to the optimal θ_0, we get

$$J(\alpha) = \min_{\theta,\theta_0} J(\theta, \theta_0; \alpha) \tag{9}$$

$$= \begin{cases} \sum_{t=1}^{n} \alpha_t - (1/2) \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j [\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)], \text{ if } \sum_{t=1}^{n} \alpha_t y_t = 0 \\ -\infty, \text{ otherwise} \end{cases} \tag{10}$$

The dual form of the solution is therefore obtained by *maximizing*

$$\sum_{t=1}^{n} \alpha_t - (1/2) \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j [\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)], \tag{11}$$

$$\text{subject to} \quad \alpha_t \geq 0, \quad \sum_{t=1}^{n} \alpha_t y_t = 0 \tag{12}$$

This is the dual or kernel form of the support vector machine, and is also a *quadratic optimization problem*. The constraints are simpler, however. Moreover, the dimension of

the input vectors does not appear explicitly as part of the optimization problem. It is formulated solely on the basis of the Gram matrix:

$$\mathbf{K} = \begin{bmatrix} \phi(\mathbf{x}_1)^T\phi(\mathbf{x}_1) & \cdots & \phi(\mathbf{x}_1)^T\phi(\mathbf{x}_n) \\ \cdots & \cdots & \cdots \\ \phi(\mathbf{x}_n)^T\phi(\mathbf{x}_1) & \ldots & \phi(\mathbf{x}_n)^T\phi(\mathbf{x}_n) \end{bmatrix} \tag{13}$$

We have already seen that the maximum margin hyperplane can be constructed on the basis of only a subset of the training examples. This should also also in terms of the feature vectors. How will this be manifested in the $\hat{\alpha}_t$'s? Many of them will be exactly zero due to the optimization. In fact, they are *non-zero* only for examples (feature vectors) that are *support vectors*.

Once we have solved for $\hat{\alpha}_t$, we can classify any new example according to the discriminant function

$$\hat{y}(\mathbf{x}) = \hat{\theta}^T\phi(\mathbf{x}) + \hat{\theta}_0 \tag{14}$$

$$= \sum_{t=1}^{n} \hat{\alpha}_t y_t [\phi(\mathbf{x}_t)^T\phi(\mathbf{x})] + \hat{\theta}_0 \tag{15}$$

$$= \sum_{t\in \mathrm{SV}} \hat{\alpha}_t y_t [\phi(\mathbf{x}_t)^T\phi(\mathbf{x})] + \hat{\theta}_0 \tag{16}$$

where SV is the set of support vectors corresponding to non-zero values of α_t. We don't know which examples (feature vectors) become as support vectors until we have solved the optimization problem. Moreover, the identity of the support vectors will depend on the feature mapping or the kernel function.

But what is $\hat{\theta}_0$? It appeared to drop out of the optimization problem. We can set θ_0 after solving for $\hat{\alpha}_t$ by looking at the support vectors. Indeed, for all $i \in \mathrm{SV}$ we should have

$$y_i(\hat{\theta}^T\phi(\mathbf{x}_i) + \hat{\theta}_0) = y_i \sum_{t\in \mathrm{SV}} \hat{\alpha}_t [\phi(\mathbf{x}_t)^T\phi(\mathbf{x}_i)] + y_i\hat{\theta}_0 = 1 \tag{17}$$

from which we can easily solve for $\hat{\theta}_0$. In principle, selecting any support vector would suffice but since we typically solve the quadratic program over α_t's only up to some resolution, these constraints may not be satisfied with equality. It is therefore advisable to construct $\hat{\theta}_0$ as the median value of the solutions implied by the support vectors.

What is the geometric margin we attain with some kernel function $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}')$?

It is still $1/\|\hat{\theta}\|$. In a kernel form

$$\hat{\gamma}_{geom} = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \hat{\alpha}_i \hat{\alpha}_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right)^{-1/2} \tag{18}$$

Would it make sense to compare geometric margins we attain with different kernels? We could perhaps use it as a criterion for selecting the best kernel function. Unfortunately this won't work without some care. For example, if we multiply all the feature vectors by 2, then the resulting geometric margin will also be twice as large (we just expanded the space; the relations between the points remain the same). It is necessary to perform some normalization before any comparison makes sense.

We have so far assumed that the examples in their feature representations are linearly separable. We'd also like to have the kernel form of the relaxed support vector machine formulation

$$\text{minimize } \|\theta\|^2/2 + C \sum_{t=1}^{n} \xi_t \tag{19}$$

$$\text{subject to } \; y_t(\theta^T \phi(\mathbf{x}_t) + \theta_0) \geq 1 - \xi_t, \;\; t = 1, \ldots, n \tag{20}$$

The resulting dual form is very similar to the simple one we derived above. In fact, the only difference is that the Lagrange multipliers α_t are now also bounded from above by C (the same C as in the above primal formulation). Intuitively, the Lagrange multipliers α_t serve to enforce the classification constraints and adopt larger values for constraints that are harder to satisfy. Without any upper limit, they would simply reach ∞ for any constraint that cannot be satisfied. The limit C specifies the point when we should stop from trying to satisfy such constraints. More formally, the dual form is

$$\sum_{t=1}^{n} \alpha_t - (1/2) \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j [\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)], \tag{21}$$

$$\text{subject to } \;\; 0 \leq \alpha_t \leq C, \;\; \sum_{t=1}^{n} \alpha_t y_t = 0 \tag{22}$$

The resulting discriminant function has the same form except that the $\hat{\alpha}_t$ values can be different. What about $\hat{\theta}_0$? To solve for $\hat{\theta}_0$ we need to identify classification constraints that are satisfied with equality. These are no longer simply the ones for which $\hat{\alpha}_t > 0$ but those corresponding to $0 < \hat{\alpha}_t < C$. In other words, we have to exclude points that violate the margin constraints. These are the ones for which $\hat{\alpha}_t = C$.

Kernel optimization

Whether we are interested in (linear) classification or regression we are faced with the problem of selecting an appropriate kernel function. A step in this direction might be to tailor a particular kernel a bit better to the available data. We could, for example, introduce additional parameters in the kernel and optimize those parameters so as to improve the performance. These parameters could be simple as the β parameter in the radial basis kernel, weight each dimension of the input vectors, or more flexible as finding the best convex combination of basic (fixed) kernels. Key to such an approach is the measure we would optimize. Ideally, this measure would be the generalization error but we obviously have to settle for a surrogate measure. The surrogate measure could be cross-validation or an alternative criterion related to the generalization error (e.g., margin).

Kernel selection

We can also explicitly select among possible kernels and cast the problem as a *model selection* problem. By choosing a kernel we specify the feature vectors on the basis of which linear predictions are made. Each model[1] (class) refers to a set of linear functions (classifiers) based on the chosen feature representation. In many cases the models are nested in the sense that the more "complex" model contains the "simpler" one. We will continue from this further at the next lecture.

[1] In statistics, a model is a family/set of distributions or a family/set of linear separators.