COS 511: Theoretical Machine Learning

Lecturer: Vladimir Vapnik
Scribe: Ankit Garg

Lecture # 13
March 26, 2013

1 Learning with nontrivial Teacher

Today we will learn about a new model of learning. In this model, a nontrivial teacher provides the learner with labelled examples plus some additional privileged information associated with the training examples. The motivation of this model comes from the role of teachers involved in human learning. In human learning, along with examples, the teacher provides students with some additional information, for example, explanations, comments, comparisons etc.

This additional (privileged) information is available only for the training examples. **It is not available for test examples.**

1.1 The Model - LUPI

In the Learning Using Privileged Information (LUPI) model, the learner is provided with a set of triplets,

$$(x_1, x_1^*, y_1), \ldots, (x_m, x_m^*, y_m), \;\; x_i \in X, \; x_i^* \in X^*, \; y_i \in \{-1, 1\}$$

generated according to a fixed but unknown probability distribution $P(x, x^*, y)$. The goal of the learner is to output from a class of functions $\mathcal{H}$, a function $h : X \to \{-, 1\}$ with low generalization error : $\Pr[h(x) \neq y]$. Note that the final classifier is only a function of x, not involving x^*. But the x^*'s provided on the training data help us construct a good classifier from the data.

Let us look at a few examples :

1. **Classification of proteins into families** - We aim to construct a rule for classification of proteins into families based on their amino-acid sequences. The space X is the space of amino-acid sequences. The privileged information space X^* is the the space of 3D structures of the proteins.

2. **Classification of digit** 5 **and digit** 8 - Given images of digits 5 and 8, we aim to classify them to either an image of 5 or an image of digit 8. The space X is the space of low quality images of digits 5 and 8. The privileged information space X^* is the the space of holistic descriptions of digit images. An example of the holistic descriptions are the Ying Yang style descriptions about the digit images (in fact about the person who might have written the digit) : "Straightforward, very active, hard, very masculine with rather clear intention. A sportsman or a warrior. Will never give a second thought to whatever. Upward-seeking. 40 years old."

3. **Dog/Cat classification** - Given images of dogs and cats, we aim to classify them to either an image of a cat or an image of a dog. The space X is the space of images of dogs and cats. The privileged information space X^* is the the space of descriptions of these images. For example, "The ear is small in proportion to its face", "The mouth

is narrow and non-prominent", "The color of the whole body is very dark and lacks diversity" , "Only see the part of the body" , "Only a dog in the picture", etc.

Privileged information exists for almost any learning problem and can play a crucial role in the learning process : it can significantly increase the speed of learning.

2 How privileged information can be used in SVM algorithms

Let us recall that in the linearly separable case, the SVM algorithm solves the following optimization problem:

$$\begin{aligned} \text{minimize}_{\mathbf{w}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i) \geq 1,\ \forall i \end{aligned}$$

In this case the generalization error is bounded by $\tilde{O}(\frac{\text{VC-dim}}{m})$, where the VC dimension can be expressed in terms of margins or the number of support vectors.

In the linearly non-separable case, the SVM algorithm solves the following optimization problem:

$$\begin{aligned} \text{minimize}_{(\mathbf{w},\boldsymbol{\xi})} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\cdot\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i) \geq 1-\xi_i,\ \forall i \\ & \xi_i \geq 0,\ \forall i \end{aligned}$$

In this case, the generalization error is at most $\tilde{O}\big(\sqrt{\frac{\text{VC-dim}}{m}}\big)$ larger than the training error, whereas in the linearly separable case, it was $\tilde{O}\left(\frac{\text{VC-dim}}{m}\right)$. The intuition for the difference is that in the linearly separable case, we estimate n parameters (of $\mathbf{w}$) using m examples, whereas in the inseparable case, one needs to estimate $n+m$ parameters (n parameters for $\mathbf{w}$ and m slack values ξ_i) using m examples.

2.1 The Oracle SVM

Suppose we are given triplets $(\mathbf{x}_1, \xi_1^0, y_1), \ldots, (\mathbf{x}_m, \xi_m^0, y_m)$, where $\xi_i^0 = \xi^0(\mathbf{x}_i)$ are the slack values w.r.t the best hyperplane (the one that minimizes the generalization error):

$$\xi^0(\mathbf{x}) = [1 - y_i(\mathbf{w}_0\cdot\mathbf{x})]_+,$$

where $\mathbf{w}_0$ defines the best hyperplane. Then we can solve the following optimization problem:

$$\begin{aligned} \text{minimize}_{\mathbf{w}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i) \geq 1-\xi_i^0,\ \forall i \end{aligned}$$

In this case, it can be shown that the generalization error is at most $\tilde{O}\big(\frac{\text{VC-dim}}{m}\big)$ larger than the training error, where the VC dimension here refers to the VC dimension of the admissible set of functions i.e. functions of the form sign($\mathbf{w}\cdot\mathbf{x}$) such that $y_i(\mathbf{w}\cdot\mathbf{x}_i) \geq 1-\xi_i^0$, for all i.

2.2 What can a real Teacher do?

In reality, however, a teacher does not know either the values of slacks or the Oracle function. Instead, he can supply the learner with privileged information $x^* \in X^*$ and with the admissible set of *correcting functions* $\xi(x^*, \delta)$, $\delta \in \Delta$ (where δ parametrizes the functions) that have a low VC dimension and contains the correcting function which defines the values of the oracle slack function

$$\xi(x^*, \delta_{\text{best}}) = \xi_0 = \xi^0(x).$$

Thus the learner gets triplets $(x_1, x_1^*, y_1), \ldots, (x_m, x_m^*, y_m)$ and he needs to simultaneously estimate the correcting function $\xi(x^*, \delta)$ and the decision hyperplane w. Now we need to solve the following optimization problem :

$$\begin{aligned} \text{minimize}_{(\mathbf{w},\delta)} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C \cdot \sum_{i=1}^{m} \xi(x_i^*, \delta) \\ \text{s.t.} \quad & y_i(\mathbf{w} \cdot \mathbf{x}_i) \geq 1 - \xi(x_i^*, \delta),\ \forall i \end{aligned}$$

Proposition 1. *With probability at least $1 - \eta$, it holds that*

$$Pr[y(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x}) < 0] \leq Pr[1 - \xi(x^*, \widehat{\delta}) < 0] + O\left(\frac{(h + h^*) \ln\left(\frac{2m}{h+h^*}\right) + \ln\left(\frac{1}{\eta}\right)}{m}\right) \tag{1}$$

where $(\widehat{\boldsymbol{w}}, \widehat{\delta})$ is the solution of the optimization problem, $Pr[y(\widehat{\boldsymbol{w}} \cdot \boldsymbol{x}) < 0]$ is the generalization error of $\widehat{\boldsymbol{w}}$, h is the VC dimension of the admissible set of hyperplanes, h^* is the VC dimension of the admissible set of correcting functions, and the probability η is with respective to the training set.

The problem is how good the teacher is: To be precise, we care about how fast the term $Pr[1 - \xi(x^*, \delta_m) < 0]$ converges to $Pr[1 - \xi(x^*, \delta_{\text{best}})]$, the generalization of the best hyperplane. This rate, together with the second term on the right hand side of (1), decides the convergence rate of the generalization error.

The goal of introducing a teacher (by introducing both the space X^* and the set of slack-functions $\xi(x^*, \delta)$, $\delta \in \Delta$) is to try to speed up the learning process, to decrease the convergence rate of the generalization error from $O\left(\sqrt{\frac{1}{m}}\right)$ towards $O\left(\frac{1}{m}\right)$. The difference between standard and fast methods is in the number of examples needed for training to achieve the same accuracy: roughly speaking, m for the standard methods and $\sqrt{m}$ for the fast methods.

3 The SVM+ algorithm

In the previous lecture, we introduced two ways of dealing with linearly inseparable data for the SVM algorithm : using slack values or mapping the data into a higher dimensional space. We could also use a combination of both: we first map $\mathbf{x}$ to $\psi(\mathbf{x})$, and then solve the following optimization problem:

$$\begin{aligned} \text{minimize}_{\mathbf{w}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C \cdot \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w} \cdot \psi(\mathbf{x}_i)) \geq 1 - \xi_i,\ \forall i \\ & \xi_i \geq 0,\ \forall i \end{aligned}$$

The decision function has the form

$$f(\mathbf{x}, \boldsymbol{\alpha}) = \text{sign}\left(\sum_{i=1}^{m} \alpha_i y_i \psi(\mathbf{x}_i) \cdot \psi(\mathbf{x})\right)$$

where the values α_i are the solutions to the dual problem:

$$\begin{aligned}
\text{maximize}_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \psi(\mathbf{x}_i) \cdot \psi(\mathbf{x}_j) \\
\text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i y_i = 0 \\
& 0 \le \alpha_i \le C, \ \forall i
\end{aligned}$$

As noted in the last lecture, we just need the dot product of $\psi(\mathbf{x}_i)$ and $\psi(\mathbf{x}_j)$ and don't really need to compute $\psi(\mathbf{x})$. Thus it is useful to choose mappings ψ for which the kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \psi(\mathbf{x}_i) \cdot \psi(\mathbf{x}_j)$ is easy to compute.

In the SVM+ algorithm, we map both the example space X and the privileged information space X^* to higher dimensional spaces using mappings ψ and ψ^*. We also use two kernel functions K and K^*. Define the slack functions in the form $\xi_i = \mathbf{w}^* \cdot \psi^*(\mathbf{x}^*)$. Then the optimization problem we need to solve is :

$$\begin{aligned}
\text{minimize}_{\mathbf{w}, \mathbf{w}^*} \quad & \|\mathbf{w}\|_2^2 + \gamma \|\mathbf{w}^*\|_2^2 + C \cdot \sum_{i=1}^{m} \mathbf{w}^* \cdot \psi^*(\mathbf{x}^*) \\
\text{s.t.} \quad & y_i(\mathbf{w} \cdot \psi(\mathbf{x}_i)) \ge 1 - \mathbf{w}^* \cdot \psi^*(\mathbf{x}^*), \ \forall i
\end{aligned}$$

The decision function has the form

$$f(\mathbf{x}, \boldsymbol{\alpha}) = \text{sign}\left(\sum_{i=1}^{m} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x})\right)$$

where the values α_i optimize the following problem :

$$\begin{aligned}
\text{maximize}_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{2\gamma} \sum_{i,j=1}^{m} (\alpha_i - \beta_i)(\alpha_j - \beta_j) K^*(\mathbf{x}_i^*, \mathbf{x}_j^*) \\
\text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i y_i = 0 \\
& \sum_{i=1}^{m} (\alpha_i - \beta_i) = 0 \\
& 0 \le \beta_i \le C, \ \forall i \\
& \alpha_i \ge 0, \ \forall i
\end{aligned}$$