Math 547: Mathematical Foundations of Statistical Learning Theory Fall 2017

Lecture 7 — September 6-8

Instructor: Stas Minsker *Scribe: Mose Wintner, Yuxiao Yang*

7.1 Adaboost

Algorithm 1. The AdaBoost algorithm

- $w_j^{(0)} := \frac{1}{n}$, $j = 1, \ldots, n$.
- For $t = 0, \ldots, T$
  - call the weak learner (WL) that outputs $f_t(\cdot)$ with $e_{n,w^{(t)}}(f_t) \le \frac{1}{2}$.
  - set
$$\alpha_t = \frac{1}{2}\log\left(\frac{1 - e_{n,w^{(t)}}(f_t)}{e_{n,w^{(t)}}(f_t)}\right).$$
  - update weights:
$$w_j^{(t+1)} = \frac{w_j^{(t)}\exp\left(-Y_j \alpha_t f_t(X_j)\right)}{\mathbb{Z}_t},$$
$$\mathbb{Z}_t := \sum_{j=1}^{n} w_j^{(t)} \exp\left(-Y_j \alpha_t f_t(X_j)\right).$$
- Output: $\widehat{g}_T(\cdot) = \text{sign}(\sum_{j=1}^{T} \alpha_t f_t(\cdot))$.

Excercise 1. If f_t classifies X_j correctly, then $w_j^{(t+1)} \le w_j^{(t)}$. If f_t classifies X_j incorrectly, then $w_j^{(t+1)} \ge w_j^{(t)}$.

Theorem 1. Assume that at each step, WL outputs f_t such that
$$e_{n,w^{(t)}}(f_t) = \sum_{j=1}^{n} w_j^{(t)} I\{Y_j \ne f_t(X_j)\} \le \frac{1}{2} - \gamma,$$
for some $\gamma > 0$, then the training error satisfies
$$\frac{1}{n}\sum_{j=1}^{n} I\{Y_j \ne \widehat{g}_T(X_j)\} \le \exp\left(-2T\gamma^2\right).$$

Proof:

a) Note that $w_j^{(T+1)} = \frac{1}{n}\frac{e^{-Y_j\sum_{t=1}^T \alpha_t f_t(X_j)}}{\prod_{t=1}^T \mathbb{Z}_t}$.
b) We also have

$$\begin{aligned}\frac{1}{n}\sum_{j=1}^n I\{Y_j \neq \widehat{g}_T(X_j)\} &= \frac{1}{n}\sum_{j=1}^n I\{Y_j\sum_{t=1}^T \alpha_t f_t(X_j) \leq 0\}\\ &\leq \frac{1}{n}\sum_{j=1}^n e^{-Y_j\sum_{t=1}\alpha_t f_t(X_j)}\\ &= \frac{1}{n}\sum_{j=1}^n w_j^{(T+1)} n\prod_{t=1}^T \mathbb{Z}_t\\ &= \prod_{t=1}^T \mathbb{Z}_t.\end{aligned}$$

c) For $\mathbb{Z}_t$ at each step

$$\begin{aligned}\mathbb{Z}_t &= \sum_{j=1}^n w_j(t)\exp\left(-Y_j\alpha_t f_t(X_j)\right)\\ &= \sum_{j=1}^n w_j^{(t)} I\{Y_j = f_t(X_j)\}e^{-\alpha_t} + \sum_{j=1}^n w_j^{(t)} I\{Y_j \neq f_t(X_j)\}e^{\alpha_t} \pm \sum_{j=1}^n w_j I\{Y_j \neq f_t(X_j)\}e^{-\alpha_t}\\ &= e^{-\alpha_t} + (e^{\alpha_t} - e^{-\alpha_t})\sum_{j=1}^n w_j^{(t)} I\{Y_j \neq f_t(X_j)\},\end{aligned}$$

where the last multiplicand is $e_{n,w^{(t)}}(f_t)$. Recall that $\alpha_t = \frac{1}{2}\log\left(\frac{1-e_{n,w^{(t)}}(f_t)}{e_{n,w^{(t)}}(f_t)}\right)$, we thus have

$$\mathbb{Z}_t = 2\sqrt{e_{n,w^{(t)}}(f_t)(1-e_{n,w^{(t)}}(f_t))}.$$

d) The function $f(x) = x(1-x); x \in [0, \frac{1}{2}-\gamma]$ is maximized for $x = \frac{1}{2}-\gamma$, thus

$$\mathbb{Z}_t \leq 2\sqrt{(1/2-\gamma)(1/2+\gamma)} \leq \sqrt{1-4\gamma^2} \leq \sqrt{e^{-4\gamma^2}} = e^{-2\gamma^2},$$

since $1-x \leq e^{-x}$ for $x \in [0,1]$. Therefore

$$\frac{1}{n}\sum_{j=1}^n I\{Y_j \neq \widehat{g}_T(X_j)\} = \prod_{t=1}^T \mathbb{Z}_t \leq \exp\left(-2T\gamma^2\right). \quad \square$$

In conclusion, the training error goes to 0 exponentially fast. In fact, we are interested in the generalization error

$$P(Y\widehat{g}_T(X)) \leq 0.$$

Minimizing this generalization error turns out to be much harder.

7.2 Support Vector Machines (SVM)

Invented by V.Vapnik and C.Cortes around 1995.

Idea: $X_j \in \mathcal{S} = \mathbb{H}$, where $\mathbb{H}$ is a separable Hilbert space equipped with inner product $\langle \cdot, \cdot \rangle$ and norm $|| \cdot ||$, and X_j have binary label $Y_j \in \{\pm 1\}$. If there are only finitely many points, then one can always find a separating "hyperplane," which classifies the X_j. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized (figure 7.1).

Let $u \in \mathbb{H}$ such that $||u|| = 1$. Then for some $c \in \mathbb{H}$, the separating hyperplane is given by the affine subspace

$$L_{u,c} := \{x \in \mathbb{H} : \langle u, x \rangle + c = 0\}.$$

Let $y \in \mathbb{H}$. Then the distance between y and $L_{u,c}$ is

$$d(y, L_{u,c}) = |\langle u, y \rangle + c|.$$

Indeed, $y = \langle y, u \rangle u + y^\perp$, where $\langle y^\perp, u \rangle = 0$. $x \in L_{u,c}$ implies that $x = x_L + v$, where $v = -cu$ and $x_L \perp u$ since $\langle u, x \rangle + c = \langle u, x_L \rangle + \langle u, v \rangle + c = 0$. Finally,

$$d(y, L_{u,c}) = \inf_{x \in L} ||y - x|| = ||y - (y^\perp + v)|| = ||\langle y, u \rangle u - v|| = |\langle y, u \rangle + c|.$$

SVM aims to solve the following problem

Problem 1. Maximize d subject to

$$\begin{aligned} \langle u, X_j \rangle + c \geq d, \qquad & Y_j = 1 \\ \langle u, X_j \rangle + c \leq -d, \qquad & Y_j = -1 \end{aligned}$$

for $j = 1, \ldots, n$.

This problem is equivalent to the following problem

Problem 2. Minimize $\frac{1}{d}$ subject to

$$Y_j(\langle u, X_j \rangle + c) \geq d$$

for $j = 1, \ldots, n$.

Let $f_{u,c}(\cdot) = \langle u, \cdot \rangle + c$, the constraint becomes

$$\min_j Y_j f_{u,c}(X_j) \geq d \Leftrightarrow \min_j Y_j(\langle u/d, X_j \rangle + c/d) \geq 1.$$

Define $w = u/d \in \mathbb{H}$ and $b = c/d \in \mathbb{R}$, so that $||w|| = 1/d$ and we now seek to minimize $1/d$ subject to

$$\min_j Y_j f_{w,b}(X_j) \geq 1,$$

which we summarize as the following problem

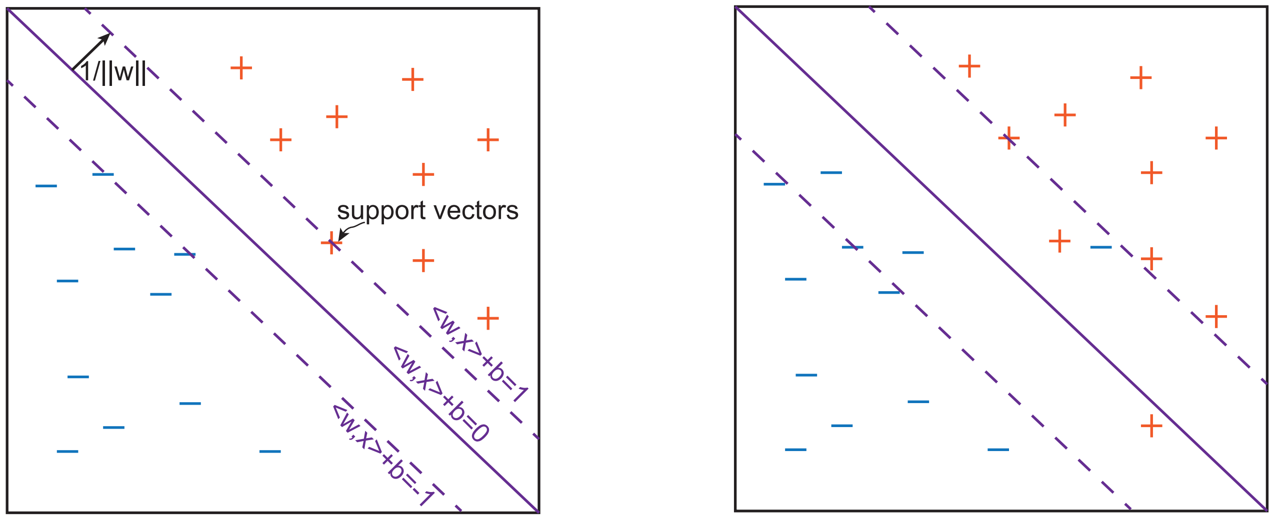

Figure 7.1. Hard-margin SVM (left) and soft-margin SYM (right)

Problem 3. Minimize $||w||$ subject to

$$\min_j Y_j f_{w,b}(X_j) \geq 1$$

for $j = 1, \ldots, n$.

This is a quadratic programming problem and is termed the "hard-margin SVM". The solution of Problem 3 has several key properties that we summarize below.

Theorem 2. (*Representer theorem*). The solution w^* of Problem 3 is in the linear span of $\{X_1, \ldots, X_n\}$.

Proof: Assume that $w^* = \tilde{w} + \tilde{w}^\perp$, where $\tilde{w} \in$ l.s.$\{X_1, \ldots, X_n\}$ and $\tilde{w}^\perp \perp$ l.s.$\{X_1, \ldots, X_n\}$. If w^* is feasible (i.e. satisfies the constraints), then $\tilde{w}$ is also feasible, because

$$\langle w^*, X_j \rangle = \langle \tilde{w}, X_j \rangle, \qquad \forall j.$$

Note $||w^*|| > ||\tilde{w}||$ if $\tilde{w}^\perp \neq 0$, and if $\tilde{w}^\perp \neq 0$ then the solution can be improved. $\square$

Definition 1. The **support vectors** are a subset $\{X_{i_1}, ..., X_{i_k}\}$ of $\{X_1, \ldots, X_n\}$ such that

$$Y_{i_j} f_{w^*,b^*}(X_{i_j}) = 1, \qquad j = 1, 2, ..., k.$$

Proposition 1. The solution w^* of problem (c) is in the linear span of $\{X_{i_1}, ..., X_{i_k}\}$.

Proof: Use the KKT (Karush-Kuhn-Tucker) conditions. In our case, they become the Fritz-John optimality conditions. Allowing inequality constraints, the KKT conditions generalize the method of Lagrange multipliers. The KKT conditions make applicable in a numerical setting the idea that continuous functions on closed sets are optimized on their boundaries. Here, the optimization problem is

$$w^* = \operatorname*{argmin}_{w \in \mathbb{H}} h(w) (= ||w||^2) \text{ s.t.}$$
$$g_j(w) := -(Y_j f_{w,b}(X_j) - 1) \le 0 \qquad \forall j.$$

Then the KKT conditions state that

$$\nabla h(w^*) + \sum_{i \in I} \alpha_i \nabla g_i(w^*) = 0,$$

where $I = \{i : \, g_i(w^*) = 0\}$

Excercise 2. Compute the gradients and complete the proof.

The above "hard-margin SVM" enforces a hard restriction on w and b. There is also "soft-margin SVM" which allows misclassification and can be applied cases in which the data are not linearly separable (figure. 7.1). Soft-margin SVM solves the following problem

Problem 4. Minimize $\lambda ||w||^2 + \frac{1}{n}\sum_{j=1}^n \xi_j$ subject to

$$\min_j Y_j f_{w,b}(X_j) \ge 1 - \xi_j, \quad \xi_j \ge 0.$$

for $j = 1, \ldots, n$.

Here, $\lambda > 0$ is a regularization parameter: as $\lambda \to \infty$, we recover hard-margin SVM.

Note that for any j, we have

$$Y_j f_{w^*,b^*}(X_j) = 1 - \xi_j^*,$$

since otherwise we can shrink ξ_j and hence shrink the objective function. Since ξ_j^* are non-negative, we have

$$\xi_j^* = (1 - Y_j f_{w^*,b^*}(X_j))_+ := \max(1 - Y_j f_{w^*,b^*}(X_j), 0).$$

Thus, Problem 4 becomes

Problem 5. Minimize $\frac{1}{n}\sum_{j=1}^n (1 - Y_j f_{w,b}(X_j))_+ + \lambda ||w||^2$ over w and b.

Recall that $Y_j f_{w,b}(X_j)$ is called the margin in binary classification. To build a connection to the Bayes classifier, we introduce the hinge loss function

$$\ell_{\text{hinge}}(y, g(x)) = (1 - y g(x))_+,$$

which is a convex function that also bounds the 0-1 loss function from above (similar with the exponential loss function, see figure 7.2). Now we define the function space (the "base set") as

$$\mathcal{F} = \mathcal{F}_{w,b} = \{f_{w,b} = \langle w, \cdot \rangle + b : \, w \in \mathbb{H}, \, b \in \mathbb{R}\}.$$

Then Problem 5 is recasted as

Problem 6. Find

$$f_{w^*,b^*} = \operatorname*{argmin}_{f \in \mathcal{F}} \left[\frac{1}{n} \sum_{j=1}^n \ell_{\text{hinge}}(Y_j f_{w,b}(X_j)) + \lambda ||w||^2 \right].$$

Excercise 3. Let $f_* = \operatorname{argmin}_{f \in \mathcal{F}} \mathbb{E}(1 - Y f(X))_+$. Then $\operatorname{sign}(f_*) = \operatorname{sign}(\eta)$ where η is the Bayes classifier.

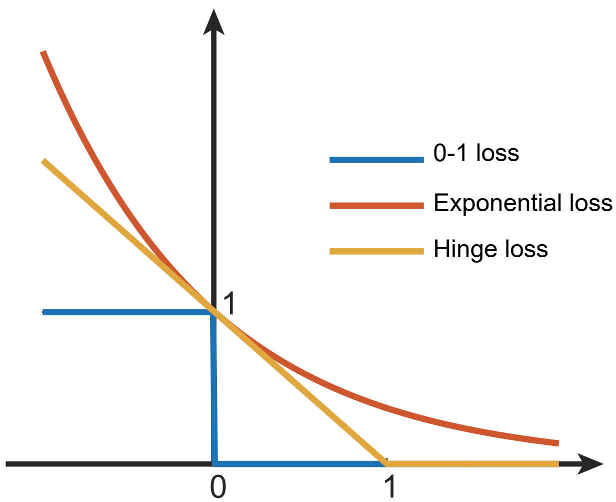

Figure 7.2. Loss fruntions: 0-1, exponential and hinge