**CS281B/Stat241B (Spring 2006) Statistical Learning Theory** **Lecture: 3**

# Perceptron Risk Bounds and Kernel Classification

*Lecturer: Peter Bartlett* *Scribe: Halldor Isak Gylfason*

## 1 Recap

We have been considering a formalization of pattern classification, where $P$ is a probability distribution on $\mathcal{X} \times \mathcal{Y}$, $\mathcal{Y} \in \{\pm 1\}$, and we wish to find a function $f_n(X)$ that accurately predicts $Y$, based on $n$ training samples $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathcal{X} \times \mathcal{Y}$, drawn from $P$. We looked at the perceptron algorithm which finds a linear threshold function from $F = \{x \mapsto \text{sign}(w^T x) : w \in \mathbb{R}\}$, given that such a function that correctly classifies all instances of the training sample exists.

We proved that the number of updates made by the perceptron algorithm is bounded by $\frac{R^2}{\delta^2}$, where $R = \max \|x_i\|$ (i.e. the training samples are in a ball of radius R), and $\delta = \min(w^T x_i y_i / \|w\|)$ (i.e. the training samples are separable by margin $\delta$). Moreover, we proved a minimax lower bound on the expected risk, namely that for any function $f_n$, there exists a probability distribution P, which can be perfectly classified by a linear threshold function, but

$$\mathbb{E}L(f_n) \geq \Big(\frac{\min(n,d) - 1}{2n}\Big)\Big(1 - \frac{1}{n}\Big)^n \tag{1}$$

This bound can be understood as a minimax lower bound in the game theoretic sense, meaning that if one party chooses a "good" $f_n$, the other party can choose the probability distribution so that the expected risk is poor:

$$\min_{f_n} \max_{P \in \mathcal{P}_F} \mathbb{E}L(f_n) \geq \Big(\frac{\min(n,d) - 1}{2n}\Big)\Big(1 - \frac{1}{n}\Big)^n \tag{2}$$

where $\mathcal{P}_F = \{P \text{ on } \mathcal{X} \times \mathcal{Y} : \exists f \in F, L(f) = 0\}$.

## 2 Upper bound of Risk for the Perceptron algorithm

If we restrict our attention to probability distributions that have a large margin, we can prove an upper bound for the risk of the perceptron algorithm:

**Theorem 2.1.** Suppose, for some $w \in \mathbb{R}^d$, $\|w\| = 1, \delta > 0$, we have almost surely $\|X\| \leq R$ and $w^T XY > \delta$. For the perceptron algorithm, $f_{P,n}$, let $f_n(X; X_1, Y_1, \ldots, X_n, Y_n) = f_{P,M}(X; X_1, Y_1, \ldots, X_M, Y_M)$ where M is chosen uniformly from $\{1, \ldots, n\}$. Then we have

$$\mathbb{E}L(f_n) \leq \frac{R^2}{n\delta^2}$$

*Proof.*

$$\begin{aligned}\mathbb{E}L(f_n) &= \frac{1}{n}\sum_{M=1}^{n}\mathbb{E}\ell(f_{P,M}(X;X_1,Y_1,\ldots,X_M,Y_M),Y)\\ &= \mathbb{E}\frac{1}{n}\sum_{M=1}^{n}\ell(f_{P,M}(X_{M+1};X_1,Y_1,\ldots,X_M,Y_M),Y_{M+1})\\ &\quad \text{since } (X,Y) \text{ and } (X_{M+1},Y_{M+1}) \text{ are iid.}\end{aligned}$$

The summation in the expression above is less than the total number of mistakes made by the perceptron algorithm where we always pick a misclassified X, with the lowest index $i$. The total number of mistakes is in turn bounded by $R^2/\delta^2$. So we get

$$\mathbb{E}L(f_n) \le \frac{1}{n}\frac{R^2}{\delta^2}\ .$$

□

In the lower bound theorem, we were maximizing over all probability distributions. If we restrict the maximization to be over those probability distributions having a large margin we can obtain a better lower bound:

**Theorem 2.2.** For any $f_n, \delta > 0, R > 0, d > 0$ with $\frac{R^2}{\delta^2} \le \min(n,d)$ there is a probability distribution P on $\mathbb{R}^d \times \{\pm 1\}$ such that some $w \in \mathbb{R}^d$ satisfies $\|w\| = 1, w^T XY \ge \delta, \|X\| \le R$, but the expected risk has the following lower bound:

$$\mathbb{E}L(f_n) \ge \frac{\frac{R^2}{\delta^2}-1}{2n}\left(1-\frac{1}{n}\right)^n$$

*Proof idea.* Choose the marginal distribution $P_X$ to have support on a linearly independent set $S \subseteq \mathbb{R}^d$ of size $k = \frac{R^2}{\delta^2}$:

$$\begin{aligned}S &= \{Re_i : 1 \le i \le k\}\\ &\quad \text{where } e_i \text{ is the } i\text{-th unit vector}\end{aligned}$$

As before, we allow all $2^k$ labellings, $b \in \{\pm 1\}$. Then for any such P, we choose

$$w = \frac{1}{\sqrt{k}}\sum_{i=1}^{k} b_i e_i$$

Then $\|w\| = 1$ and $w^T XY = \frac{R}{\sqrt{k}} = \delta$ (i.e. the linear threshold function with normal $w$, classifies all points with margin $\delta$). This choice of distribution makes the proof go through as in the earlier case. □

# 3 Kernel Classification

The last section showed that if we have a large margin we obtain better bounds for the risk. Therefore, it makes intuitive sense to choose a linear threshold function that **maximises** the margin (see figure 1). This

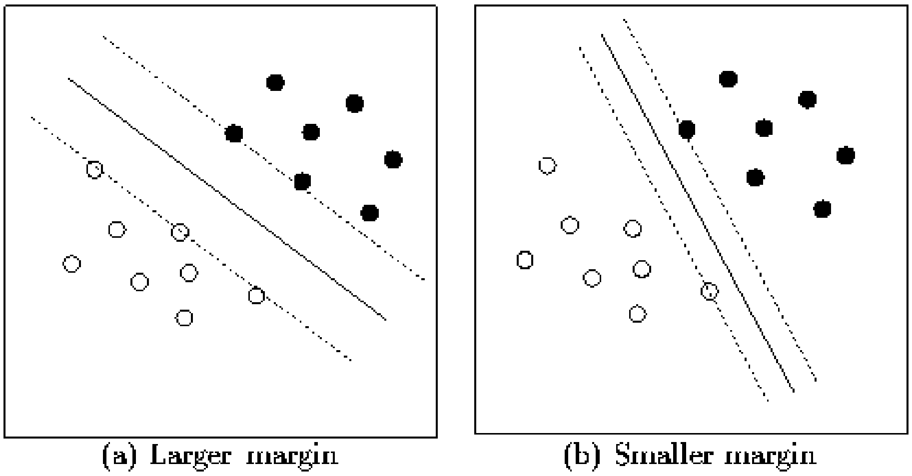


Figure 1: The classifier in (a) has wider margin than the one in (b)

means that we should maximise the minimum distance from the sample points to the threshold function $\{x : w^T x = 0\}$. This distance is given by $\frac{w^T x}{\|w\|}$.

## 3.1 Support Vector Machines

Support Vector Machines (SVMs) are linear classifiers $x \mapsto \text{sign}(w^T x)$ that solve the following optimization problem:

$$\begin{aligned} &\max_{\gamma, w} && \gamma \\ &\text{so that} && \frac{y_i w^T x_i}{\|w\|} \geq \gamma,\ i \in \{1, \ldots, n\} \end{aligned}$$

Obviously, the length of the vector w does not change the hyperplane, so we can in particular choose $\|w\| = 1$ to get the following equivalent problem:

$$\begin{aligned} &\max && \gamma \\ &\text{such that} && y_i w^T x_i \geq \gamma,\ i \in \{1, \ldots, n\} \\ & && \|w\|^2 = 1 \end{aligned}$$

Alternatively, we can simplify by insisting that $\|w\| = \frac{1}{\gamma}$:

$$\begin{aligned} &\min && \frac{1}{2}\|w\|^2 \\ &\text{such that} && y_i w^T x_i \geq 1,\ i \in \{1, \ldots, n\} \end{aligned}$$

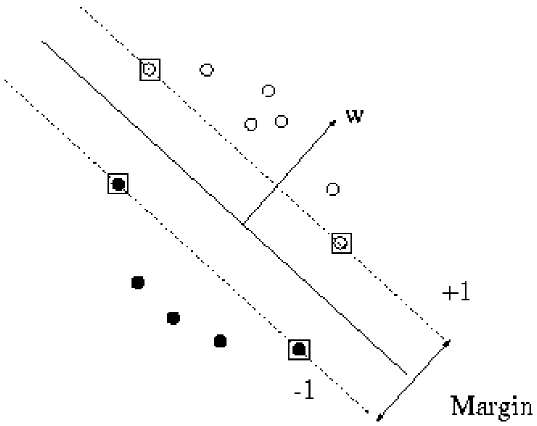


Figure 2: Support Vectors are in the rectangles

where we have scaled the objective function by $\frac{1}{2}$ to simplify the derivative (scaling obviously does not change the value of $w$ at which the minimum is achieved).

This is a quadratic program (QP) (Quadratic objective function with linear constraints). To solve this we introduce Lagrange multipliers $\alpha_i \geq 0, i = 1, \ldots, n$ and define the Lagrangian:

$$L(w, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i (y_i w^T x_i - 1) \tag{3}$$

We need to minimize with regards to $w$ (the primal variables), and maximize with regard to $\alpha$ (the dual variables). At solution we have:

$$\frac{\partial}{\partial w} L(w, \alpha) = 0 \Leftrightarrow w = \sum_i \alpha_i y_i x_i \tag{4}$$

We observe that:

- if $y_i w^T x_i < 1$ then a large value of $\alpha_i$ induces a large value for $L(w, \alpha)$, i.e. $w$ within the feasible region is better.
- If $y_i w^T x_i > 1$ then $\alpha_i = 0$ in order to minimize $L(w, \alpha)$

Hence, $\alpha_i > 0 \Rightarrow y_i w^T x_i = 1$, i.e. only the data points $x_i$ on the margin have a positive $\alpha_i$. These $x_i$ are called support vectors (see figure 2 ) and they alone determine the solution. In fact, if we throw away every other data point from the training sample, we get the same hyperplane as the solution to our optimization problem.