CS281B/Stat241B (Spring 2006) Statistical Learning Theory **Lecture: 2**

Classification and the Perceptron Algorithm

Lecturer: Peter Bartlett *Scribe: Bonnie Kirkpatrick*

Last lecture, we discussed excess risk and pattern classification. Suppose we are given a training data set $(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)$ drawn i.i.d. from some probability distribution $(X_i, Y_i) \sim P$ where P is a distribution on $\mathcal{X} \times \mathcal{Y}$. Our discussion will focus on the case where the data have two classes, $Y_i \in \{\pm 1\}$. Recall that the goal of pattern classification is to design an algorithm that chooses some function $f : \mathcal{X} \to \mathcal{Y}$ which accurately predicts the class Y_i of each data point X_i. The algorithm is limited to choosing $f \in F$ where F is a class of functions available to the algorithm. Since each classification algorithm chooses a prediction rule based on the training data set, the prediction rule is a random variable which we refer to as $f_n : x \mapsto f_n(x; X_1, Y_1, ..., X_n, Y_n)$. We would like our algorithm to choose a prediction rule with a small expected risk, $\mathbb{E}[L(f_n)]$, where the risk, defined as $L(f) = \mathbb{E}l(f(X), Y)$, gives the expected number of misclassifications a function f will make.

1 Perceptron Algorithm

The perceptron algorithm performs pattern classification when F is the class of linear threshold functions, $F = \{x \mapsto \text{sign}(w^\top x) : w \in \mathbb{R}^d\}$ where

$$\text{sign}(\alpha) = \begin{cases} +1, & w^\top x > 0 \\ 0, & w^\top x = 0 \\ -1, & w^\top x < 0 \end{cases} .$$

Algorithm 1 Perceptron Algorithm

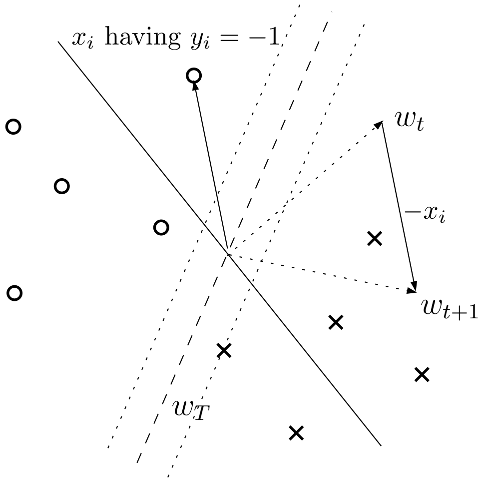
Given $(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n) \in \mathcal{X} \times \{\pm 1\}$
$w_0 := 0$, $t := 0$

while there exists $i \in \{1, ..., n\}$ such that $y_i \neq \text{sign}\{w_t^\top x_i\}$ **do**
  pick some $i \in \{1, ..., n\}$ with $y_i \neq \text{sign}\{w_t^\top x_i\}$
  $w_{t+1} := w_t + y_i x_i$
  $t := t + 1$
end while

return w_t

Notice that the algorithm can use any method for choosing a misclassified point x_i, $i \in \{1, ..., n\}$ for the update. The update step rotates the vector w_t towards x_i with positive y_i and away from x_i with negative y_i. So, the new hyperplane defined by w_{t+1} will be closer to classifying x_i correctly, see Fig 1. This version of the perceptron algorithm can easily be modified to add a constant term, b, to the threshold function: $\text{sign}\{w_t^\top x + b\}$. To estimate hyperplanes in this new function class, add another dimension to w_t to represent b, and add another dimension to x_i, for all i, containing a 1-entry in the new dimension.

Figure 1: **Perceptron algorithm step:** The update for point x_i having $y_i = -1$ where the solid line is the hyperplane defined by w_t, and the dashed line is the separating hyperplane w_T when the algorithm terminates.



2 Termination Bound

In order to understand the running-time of the perceptron algorithm, we need to characterize the number of update steps it takes before terminating. Since the number of updates is data dependent, the following result gives a bound on the number of steps in terms of some properties of the data.

Theorem 2.1. (a) If some $w \in \mathbb{R}^d$ with $\|w\| = 1$ correctly classifies all data points $y_i = \text{sign}(w^\top x_i)$ $\forall i = 1, ..., n$, then the perceptron algorithm terminates.
(b) Furthermore, it terminates after $t \leq \frac{R^2}{\delta^2}$ update steps where

$$R = \max_i \|x_i\|$$

and

$$\delta = \min_i \; w^\top x_i y_i \; .$$

Proof. Since (b) implies (a), we will prove (b). Fix some $w \in \mathbb{R}^d$ such that $y_i = \text{sign}(w^\top x_i)$ $\forall i = 1, ..., n$, so w separates the data. Suppose $\|w\| = 1$. Since $w_0 = 0$, we know that $w_0^\top w = 0$. By the update step and the definition of δ,

$$\begin{aligned} w_{t+1}^\top w &= (w_t + y_i x_i)^\top w \\ &= w_t^\top w + y_i x_i^\top w \\ &\geq w_t^\top w + \delta. \end{aligned}$$

Notice that we can write $w_t^\top w$ as a telescoping sum:

$$\begin{aligned} w_t^\top w &= w_0^\top w + \sum_{i=0}^{t-1} (w_{i+1}^\top w - w_i^\top w) \\ &\geq t\delta. \end{aligned}$$

Also $\|w_0\| = 0$. By again using the update step we have

$$
\begin{aligned}
\|w_{t+1}\|^2 &= \|w_t\|^2 + y_i^2\|x_i\|^2 + 2y_i w_t^\top x_i \\
&\leq \|w_t\|^2 + R^2
\end{aligned}
$$

since x_i are incorrectly classified by w_t, we have $2y_i x_i^\top w_t < 0$. Again we use a telescoping sum to get

$$\|w_t\|^2 \leq tR^2.$$

Using Cauchy-Schwartz, we complete the argument to obtain the bound on the number of update steps taken by the algorithm: $t\delta \leq w_t^\top w \leq ||w_t||||w|| \leq ||w_t|| \leq \sqrt{t}R$ which implies that $t \leq \frac{R^2}{\delta^2}$.

□

Remark. Even when the data are separable, the Perceptron Algorithm does not necessarily run in time linear to the number of input points. In fact, this results shows us that it is possible to select a small set of points that make δ exponentially small (i.e. x_i arbitrarily close to w).

One important feature of this algorithm is that it only uses inner products of the data points. The update step of the algorithm produces a w_t which is a linear combination of the x_i, so we write it as $w_t = \sum_{j=1}^n \alpha_j x_j$ for $\alpha_j \in \mathbb{R}$ and $\|\alpha\|_1 = \sum_{j=1}^n |\alpha_j| \leq t$. For this choice of w_t, the prediction rule becomes

$$\hat{y}_i = \text{sign}(w_t^\top x_i) = \text{sign}(\sum_j \alpha_j x_j^\top x_i).$$

Then the update becomes (i is one of the misclassified points):

$$\alpha_i^{(t+1)} := \alpha_i^{(t)} + y_i$$

since for the update $w_{t+1} := \sum_{j=1}^n \alpha_j x_j + y_i x_i$, the $y_i x_i$ only contributes to the x_i'th term of the sum.

This suggests the idea of mapping the data points to a higher-dimensional feature-space and considering inner products there. Let the mapping function be $\phi(x) : \mathbb{R}^d \to \mathbb{R}^D$ where $D \gg d$. Then the class of functions we consider is $F = \{x \mapsto \text{sign}(w^\top \phi(x)) : w \in \mathbb{R}^D\}$. Just like above, the update can be performed using α_i where $w^\top \phi(x_i) = \sum_j \alpha_j \phi(x_j)^\top \phi(x_i)$. For example, let $x = (x_1, x_2)^\top \in \mathbb{R}^2$and let $\phi(x) = (1, x_1, x_2, x_1^2, x_2^2, x_1 x_2)^\top \in \mathbb{R}^6$. The decision boundaries for this choice of ϕ are ellipses.

When using a dimension D larger than the sample size n, the data are always separable, it is not clear whether a small empirical risk will indicate a small risk. So, we must carefully consider the statistical properties of the risk.

3 Bounds on the Risk

The main theorem in this section shows that for any prediction rule (from the class of linear threshold functions), there is a probability distribution that is linearly separable, but for which the prediction rule does poorly when averaged over all training data sets. This result is general for any linear classification method, and we will see that the idea of a VC-dimension can further generalize this result to give a bound on the risk under other function classes.

Lemma 3.1. Suppose the sample size $n \leq d$, where d is the dimension of the data. If $\{x_1, x_2, ..., x_n\}$ are linearly independent, the data are separable and there is a w such that $\text{sign}(w^\top x_i) = y_i \ \forall i$.

PROOF. Since $\{x_1, x_2, ..., x_n\}$ are linearly independent, linear algebra tells us that for any $(y_1, y_2, ..., y_n) \in \{\pm 1\}^n$ we can find $w \in \mathbb{R}^d$ satisfying the set of linear equations $w^\top \left[x_1, x_2, ..., x_n\right] = (y_1, y_2, ..., y_n)$ where each x_i is a column in the matrix. Clearly this is a stronger condition than $\text{sign}(w^\top x_i) = y_i \ \forall i$. □

Theorem 3.2. For any prediction rule $f_n : \mathcal{X} \times (\mathcal{X} \times \{\pm 1\}) \to \mathcal{Y}$ with $\mathcal{X} = \mathbb{R}^d$ there is a probability distribution P on $\mathcal{X} \times \{\pm 1\}$ such that some linear threshold function $f \in F = \{x \mapsto \text{sign}(w^\top x) : w \in \mathbb{R}^d\}$ has perfect risk $L(f) = 0$, but

$$\mathbb{E}[L(f_n)] \geq \left(\frac{\min(n, d) - 1}{2}\right)\left(1 - \frac{1}{n}\right)^n.$$

PROOF. We will use the probabilistic method. We will choose P uniformly from some set $\mathcal{P}$ of probability distributions and show that the expectation of some expected risk, $\mathbb{E}[\mathbb{E}[L(f_n)]]$, is larger than some bound β. This implies that there is at least one $P \in \mathcal{P}$ for which $\mathbb{E}[L(f_n)] > \beta$.

Fix $1 > \epsilon > 0$. Choose a set $\{z_1, z_2, ..., z_d\} \in \mathcal{X} = \mathbb{R}^d$ which are linearly independent. Construct the set $\mathcal{P}$ such that for every $b = (b_1, b_2, ..., b_d) \in \{\pm 1\}^d$ there is a $P_b \in \mathcal{P}$ having joint density

$$P_b(x, y) = \begin{cases} \frac{\epsilon}{d-1} & if \ (x, y) \in \{(z_1, b_1), ..., (z_{d-1}, b_{d-1})\} \\ 1 - \epsilon & if \ (x, y) = (z_d, b_d) \\ 0 & \text{otherwise} \end{cases}.$$

If ϵ is small, this density ensures that most $(X, Y) \sim P_b$ will be (z_d, b_d), with a few points being (z_i, b_i) for $i \neq d$. No mass will be given to points (x, y) having $x \notin \{z_1, ..., z_d\}$.

From the lemma, for every b there exists a $w \in \mathbb{R}^d$ such that $\text{sign}(w^\top x_i) = b_i \ \forall i$. Since for any choice of b there is a $P \in \mathcal{P}$ with conditional distribution $P(Y = 1|X = z_i) = 1[b_i = 1]$, then the lemma implies that there is an $f \in F$ such that $L(f) = 0$ under P.

Now we will lower bound $\mathbb{E}[L(f_n)]$ by taking its expectation over b. Let $D = \{(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)\}$ be our training data set where $(X_i, Y_i) \sim P_b$ for some P_b. Define the random variable

$$N = |\{z_1, z_2, ..., z_{d-1}\} \ \setminus \ \{X_1, X_2, ..., X_n\}|$$

which is the number of points z_i that do not occur in the training data.

$$\begin{aligned}
\mathbb{E}_b\mathbb{E}_D[L(f_n)] &= \sum_{k=0}^{d-1} \mathbb{E}_b\mathbb{E}_D\Big[L(f_n)|N = k\Big]P[N = k] \\
&= \sum_{k=0}^{d-1} \mathbb{E}_b\mathbb{E}_D\Big[\mathbb{E}_{(X,Y)}[l(f_n(X), Y)|X_1, Y_1, ..., X_n, Y_n]\Big|N = k\Big]P[N = k] \\
&= \sum_{k=0}^{d-1} \mathbb{E}_b\mathbb{E}_D\Big[\sum_{i=1}^{d} l(f_n(X), b_i)P_b(z_i, b_i)\Big|X_1, \ldots, Y_n, N = k\Big]P[N = k] \\
&\geq \sum_{k=0}^{d-1} \mathbb{E}_b\mathbb{E}_D\Big[\sum_{i=1}^{d-1} l(f_n(X), b_i)P_b(z_i, b_i)\Big|X_1, \ldots, Y_n, N = k\Big]P[N = k] \\
&= \sum_{k=0}^{d-1} \mathbb{E}_D\Big[\sum_{i=1}^{d-1} \mathbb{E}_b l(f_n(X), b_i)P_b(z_i, b_i)\Big|X_1, \ldots, Y_n, N = k\Big]P[N = k] \\
&\geq \sum_{k=0}^{d-1} \left(\frac{1}{2}\right)\frac{\epsilon}{d-1}(k)\,P[N = k] \\
&= \frac{\epsilon}{2(d-1)}\mathbb{E}N
\end{aligned}$$

The first step expands the expectation of the risk using conditional expectation. The second step uses the definition of risk, while the third step simplifies the expectation over (X, Y) by only considering the loss for points having non-zero probability in P_b. The fourth step drops a positive term and hence makes the quantity smaller. The fifth step moves the expectation w.r.t. b inside. For the sixth step observe that, for the k z_i's we haven't seen in the data set, the labels b_i are independent uniform $\{\pm 1\}$-valued random variables (conditioned on the data we have seen) and hence cannot be guessed with a probability any better than $1/2$. The final step simply uses the definition of $\mathbb{E}N$.

Since

$$\begin{aligned} \mathbb{E}N &= \sum_{j=1}^{d-1} P[z_j \notin \{x_1, x_2, ..., x_n\}] \\ &= (d-1)\left(1 - \frac{\epsilon}{d-1}\right)^n, \end{aligned}$$

the above expectation implies that there is a probability distribution P for which $\mathbb{E}L(f_n) \geq \frac{\epsilon}{2}\left(1 - \frac{\epsilon}{d-1}\right)^n$. Choose ϵ to be

$$\epsilon = \begin{cases} \frac{d-1}{n} & if\ n \geq d-1 \\ \frac{n-1}{n} & \text{otherwise} \end{cases}.$$

Then for $n \geq d-1$, $\mathbb{E}L(f_n) \geq \frac{d-1}{2n}(1 - \frac{1}{n})^d$. Otherwise $\mathbb{E}L(f_n) \geq \frac{n-1}{2n}(1 - \frac{1}{n})^n$. $\square$

This theorem says that if $\frac{d}{n}$ is large, meaning that the number of data points is much smaller than the dimension, then for any f_n there is some bad probability distribution P. We will see in the following lecture that the bad P also has a small margin.

We can also use the Vapnik-Chervonenkis dimension to generalize the above theorem. A class of functions $F : \mathcal{X} \to \pm 1$ *s*hatters a set $\{x_1, x_2, ...x_n\} \subseteq \mathcal{X}$ if $\{(f(x_1), ..., f(x_n)) : f \in F\} = \{\pm 1\}^n$. Let the VC-dimension of F be defined as $d_{VC}(F) = \max\{n : F \text{ shatters some } \{x_1, ..., x_n\} \subseteq \mathcal{X}\}$. We can generalize the above result using $d \leq d_{VC}(F)$ and letting $\{z_1, ..., z_d\}$ be a set shattered by F.