1 Basic Perceptron

Algorithm 1 Perceptron algorithm.

1: \textbf{procedure} \textsc{Perceptron}((x_1, y_1), ..., (x_m, y_m))
2: \hspace{1em} Initialize \( w_0 = 0 \) and \( t = 0 \)
3: \hspace{1em} \textbf{repeat}
4: \hspace{2em} if there exists \( i \) with \( y_i \neq \text{sign}(\langle w_t, x_i \rangle) \) then
5: \hspace{3em} \( w_{t+1} \leftarrow w_t + y_i x_i \)
6: \hspace{3em} \( t \leftarrow t + 1 \)
7: \hspace{2em} \textbf{end if}
8: \hspace{1em} \textbf{until} no such \( i \) exists
9: \hspace{1em} \textbf{return} \( w_t \)

Let’s define \( \text{margin}(w) = \min_i |\langle w, x_i \rangle|/\|w\| \). Note that \( \langle w, x_i \rangle \) is the length of the orthogonal projection of \( x_i \) onto the subspace \( \{ x : \langle w, x \rangle = 0 \} \). Alternatively, it is the cosine of the angle between \( w \) and \( x_i \).

**Theorem 1.1.** Let \( w^* \) be a consistent linear classifier with \( \|w^*\| = 1 \) such that \( \gamma := \text{margin}(w^*) \) is maximized. Then Algorithm 1 terminates after at most \( 1/\gamma^2 \) iterations.

This is a nice result because it doesn’t depend on the dimensionality of the data, just on the geometric margin properties of the data.

Why are the Perceptron updates a good idea? Suppose \( y_i = 1 \) but \( \langle w_t, x_i \rangle < 0 \). Then
\[
\langle w_{t+1}, x_i \rangle = \langle w_t + x_i, x_i \rangle = \langle w_t, x_i \rangle + \langle x_i, x_i \rangle = 1
\]
So the inner product between the solution and this example improves by 1, which seems good.

The formal analysis of the algorithm argues that \( w_t \) gets “closer” to \( w^* \).

**Claim 1.2.** \( \langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma \).
Proof. Suppose $y_i = 1$ but $\langle w_t, x_i \rangle < 0$. Then
\[
\langle w_{t+1}, w^* \rangle = \langle w_t + x_i, w^* \rangle = \langle w_t, w^* \rangle + \langle x_i, w^* \rangle \geq \gamma.
\]
The argument is similar if $y_i = -1$.

But this does not really show that $w_{t+1}$ gets closer to $w^*$. $w_{t+1}$ could be “cheating” by just increasing its norm. The next claim rules out excessive cheating.

Claim 1.3. $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$.
Proof. Again, suppose $y_i = 1$ but $\langle w_t, x_i \rangle < 0$. Then
\[
\langle w_{t+1}, w_{t+1} \rangle = \langle w_t + x_i, w_t + x_i \rangle = \langle w_t, w_t \rangle + 2 \langle w_t, x_i \rangle + \langle x_i, x_i \rangle \leq \langle w_t, w_t \rangle + 1.
\]

Proof (of Theorem 1.1). By induction, Claim 1.2 gives that $\langle w_t, w^* \rangle \geq t \cdot \gamma$. By induction, Claim 1.3 gives that $\|w_t\| \leq \sqrt{t}$. How can we combine these two? How can we relate inner products and norms? Cauchy-Schwarz of course.
\[
t \cdot \gamma \leq \langle w_t, w^* \rangle \leq \|w_t\| \|w^*\| \leq \sqrt{t}
\]

\[
\Rightarrow \quad \sqrt{t} \leq 1/\gamma
\]

\[
\Rightarrow \quad t \leq 1/\gamma^2.
\]

2 Margin Perceptron

The analysis of Algorithm 1 is elegant, but unsatisfying in one way. The hypothesis of the theorem is that there is a hypothesis with large margin. The hypothesis output by the algorithm is guaranteed to correctly classify all points, but there are no guarantees about its margin.

It turns out that we can modify the algorithm in a simple way so that we can analyze the margin too. See Algorithm 2. Roughly, any point that has small margin with respect to the current hypothesis is treated the same as a misclassified point.

Theorem 2.1. Suppose there is a hypothesis $w^*$ with margin at least $\gamma$. Then Algorithm 2 outputs a classifier with margin at least $\gamma/3$ after at most $3/\gamma^2$ iterations.

As before, we may assume that $\|w^*\| = 1$.

Claim 2.2. $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 3$.
Proof. Again assume $y_i = 1$. As before,
\[
\|w_{t+1}\|^2 = \|w_t\|^2 + 2 \langle w_t, x_i \rangle + \|x_i\|^2
\]
Now, either $x_i$ was a misclassification, in which case $\langle w_t, x_i \rangle < 0$, or it had poor margin, in which case $\langle w_t, x_i \rangle \leq 1$. In either case, we have $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 3$. 

2
Algorithm 2 The Margin-Perceptron algorithm.

1: procedure MarginPerceptron((x₁, y₁), ..., (xₘ, yₘ))
2:   Initialize w₁ = 0 and t = 1
3: repeat
4:   Find any i with
5:     (Misclassification) yᵢ ≠ sign(⟨wₜ, xᵢ⟩)
6:     (Poor margin) |⟨wₜ, xᵢ⟩| ≤ 1
7:   if such an i exists then
8:     wₜ₊₁ ← wₜ + yᵢxᵢ
9:     t ← t + 1
10: end if
11: until no such i exists
12: return wₜ

Thus, by induction, the cumulative increase in the norm of wₜ is

\[ \|wₜ\|^2 \leq 3t \implies \|wₜ\| \leq \sqrt{3t}. \]

Proof (of Theorem 2.1).

Number of iterations. Claim 1.2 holds without change so we have \( \langle wₜ, w^* \rangle \geq t \cdot \gamma \), as in Theorem 1.1. The bound on the number of iterations is similar:

\[ t \cdot \gamma \leq \langle wₜ, w^* \rangle \leq \|wₜ\| \|w^*\| \leq \sqrt{3t} \]

\[ \implies \sqrt{t} \leq \frac{\sqrt{3}}{\gamma} \]

\[ \implies t \leq \frac{3}{\gamma^2}. \]

Margin. The output classifier w has \( |\langle w, xᵢ \rangle| > 1 \) for each i. So

\[ \text{margin}(w) = \min_i \frac{|\langle w, xᵢ \rangle|}{\|w\|} > \frac{1}{\|w\|} \geq \frac{1}{\sqrt{3t}} \geq \frac{1}{\sqrt{3 \cdot (3/\gamma^2)}} = \gamma/3. \]

■