The perceptron algorithm

The perceptron algorithm is a simple method for learning a linear classifier. It works on a stream of examples \((x_t, y_t)\) where \(x_t\) is in some vector space \(V\) and \(y_t \in \{-1, 1\}\).

The state of the perceptron algorithm is a vector \(w_t \in V\) that represents our linear classifier: we predict according to whether \(w_t\) has positive or negative inner product with the next example,

\[ \hat{y}_t = \text{sgn}(w_t \cdot x_t). \]

(We can break ties arbitrarily if \(w_t \cdot x_t = 0\).)

The perceptron algorithm initializes \(w_1\) to 0, and updates \(w_t\) as it processes the stream of examples. The perceptron update has three cases:

- If we predict correctly for example \(t\) (that is, if \(y_t = \text{sgn}(x_t \cdot w_t)\)), then we keep \(w_{t+1} = w_t\).
- If we make a mistake on a positive example, we update \(w_{t+1} = w_t + x_t\).
- If we make a mistake on a negative example, we update \(w_{t+1} = w_t - x_t\).

The perceptron update makes sense since it moves us toward a correct prediction. For example, on a positive example \(x_t \neq 0\), if we see the same \(x_t\) again, our dot product increases:

\[ w_{t+1} \cdot x_t = w_t \cdot x_t + x_t \cdot x_t > w_t \cdot x_t. \]

Mistake bound
The perceptron algorithm satisfies many nice properties. Here we'll prove a simple one, called a mistake bound: if there exists an optimal parameter vector \( w^* \) that can classify all of our examples correctly, then the perceptron algorithm will make at most a bounded number of mistakes before discovering some optimal parameter vector.

In more detail, suppose that \( w^* \cdot x_t \geq \epsilon \) for all positive examples, and \( w^* \cdot x_t \leq -\epsilon \) for all negative ones. Also assume that our examples are bounded: there is a constant \( U \) such that \( \|x_t\| \leq U \) for all \( t \).

Then, we will show that the perceptron algorithm will make at most

\[
\frac{U^2\|w^*\|^2}{\epsilon^2}
\]

mistakes in total. For example, if our examples have norm at most 2, if our optimal parameter vector has norm \( \|w^*\| = 3 \), and if \( \epsilon = \frac{1}{2} \), then we make no more than 144 mistakes in total.

To track mistakes, define \( M_t \) to be the total number of mistakes we make up to (but not including) example \( t \). So, \( M_1 = 0 \) (we have not yet had a chance to make any mistakes yet), and \( M_{t+1} \) is equal to either \( M_t \) (if we get example \( t \) right) or \( 1 + M_t \) (if we make a mistake on example \( t \)).

**Tools**

In our proof we'll use some properties of inner product spaces. One of the key ones is Hölder's inequality: for any two vectors \( u, v \), we have

\[
u \cdot v \leq \|u\| \|v\|\]

We'll also use the usual axioms for addition, scalar multiplication, norm, and inner product, such as the fact that inner product distributes over addition and the fact that \( \|u\|^2 = u \cdot u \).

**Proof I:**

First we show a lower bound on \( w_t \cdot w^* \). We'll use induction and proof by cases.

After a mistake on a positive example, we have
since, by assumption, \( x_t \cdot w^* \geq \epsilon \). Similarly, on a negative example, we have

\[
\begin{align*}
w_{t+1} \cdot w^* &= w_t \cdot w^* - x_t \cdot w^* \\
&\geq w_t \cdot w^* + \epsilon
\end{align*}
\]

since, by assumption, \( x_t \cdot w^* \leq -\epsilon \). So, for all \( t \), we have by induction that

\[
w_t \cdot w^* \geq \epsilon M_t
\]

The LHS starts at \( w_1 \cdot w^* = 0 \), doesn't change when we predict correctly, and increases by at least \( \epsilon \) with each mistake; the RHS starts at \( \epsilon M_1 = 0 \), doesn't change when we predict correctly, and increases by exactly \( \epsilon \) with each mistake.

**Proof II:**

Next we show an upper bound on \( \| w_t \| \). Again we use induction and proof by cases.

After a mistake on a positive example, we have

\[
\begin{align*}
w_{t+1} \cdot w_{t+1} &= w_t \cdot w_t + 2w_t \cdot x_t + x_t \cdot x_t \\
&\leq w_t \cdot w_t + 0 + U^2
\end{align*}
\]

To see why, note that \( w_t \cdot x_t \leq 0 \), since we (mistakenly) classified this example as negative. And, \( x_t \cdot x_t = \| x_t \|^2 \leq U^2 \) by assumption.

Similarly, after a mistake on a negative example, we have

\[
\begin{align*}
w_{t+1} \cdot w_{t+1} &= w_t \cdot w_t - 2w_t \cdot x_t + x_t \cdot x_t \\
&\leq w_t \cdot w_t + 0 + U^2
\end{align*}
\]

In this case, \( w_t \cdot x_t \geq 0 \), since we (mistakenly) classified this example as positive.

Just as in the previous section, we can now do induction on \( t \): for all \( t \), we have

\[
w_t \cdot w_t \leq M_t U^2.
\]
The LHS $w_t \cdot w_t$ starts at zero when $t = 1$, doesn't change unless we make a mistake, and increases by at most $U^2$ on each mistake; the RHS $M_t U^2$ starts at 0 when $t = 1$, doesn't change unless we make a mistake, and increases by exactly $U^2$ on each mistake.

**Proof III:**

If we divide the conclusion of part I by $\epsilon$ and then square both sides, we get

$$M_t^2 \leq \left( \frac{w_t \cdot w^*}{\epsilon} \right)^2$$

(We are implicitly using $M_t \geq 0$ so that squaring preserves order.) By Hölder's inequality, we therefore have

$$M_t^2 \leq \left( \frac{\|w_t\| \|w^*\|}{\epsilon} \right)^2$$

Substituting in the conclusion of part II using $w_t \cdot w_t = \|w_t\|^2$, we get

$$M_t^2 \leq \frac{M_t U^2 \|w^*\|^2}{\epsilon^2}$$

and dividing through by $M_t$ we get

$$M_t \leq \frac{U^2 \|w^*\|^2}{\epsilon^2}$$

as claimed.