Module 2.6: Proof of Convergence

- Now that we have some faith and intuition about why the algorithm works, we will see a more formal proof of convergence ...

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n + 1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition:

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n + 1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition: If the sets P and N are finite and linearly separable,

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition: If the sets P and N are finite and linearly separable, the perceptron learning algorithm updates the weight vector $\mathbf{w}_t$ a finite number of times.

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition: If the sets P and N are finite and linearly separable, the perceptron learning algorithm updates the weight vector $\mathbf{w}_t$ a finite number of times. In other words: if the vectors in P and N are tested cyclically one after the other,

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n + 1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition: If the sets P and N are finite and linearly separable, the perceptron learning algorithm updates the weight vector $\mathbf{w}_t$ a finite number of times. In other words: if the vectors in P and N are tested cyclically one after the other, a weight vector $\mathbf{w}_t$ is found after a finite number of steps t which can separate the two sets.

Theorem

Definition: Two sets P and N of points in an n-dimensional space are called absolutely linearly separable if $n+1$ real numbers $w_0, w_1, ..., w_n$ exist such that every point $(x_1, x_2, ..., x_n) \in P$ satisfies $\sum_{i=1}^{n} w_i * x_i > w_0$ and every point $(x_1, x_2, ..., x_n) \in N$ satisfies $\sum_{i=1}^{n} w_i * x_i < w_0$.

Proposition: If the sets P and N are finite and linearly separable, the perceptron learning algorithm updates the weight vector $\mathbf{w}_t$ a finite number of times. In other words: if the vectors in P and N are tested cyclically one after the other, a weight vector $\mathbf{w}_t$ is found after a finite number of steps t which can separate the two sets.

Proof: On the next slide

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Setup:

- If $x \in N$ then $-x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^Tx < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^Tp \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$

$N \leftarrow inputs \quad with \quad label \quad 0;$

N^- *contains negations of all points in* N;

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
$N^- contains\ negations\ of\ all\ points\ in\ N;$
$P' \leftarrow P \cup N^-;$

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-;$
Initialize **w** randomly;

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-;$
Initialize **w** randomly;
while !*convergence* **do**

end
//the algorithm converges when all the inputs are classified correctly

Setup:

- If $x \in N$ then $\text{-}x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$

$N \leftarrow inputs \quad with \quad label \quad 0;$

N^- *contains negations of all points in* N;

$P' \leftarrow P \cup N^-;$

Initialize **w** randomly;

while !*convergence* **do**

    Pick random $\mathbf{p} \in P'$;

end

//the algorithm converges when all the inputs are classified correctly

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow$ *inputs with label* 1;
$N \leftarrow$ *inputs with label* 0;
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-$;
Initialize **w** randomly;
while !*convergence* **do**
  Pick random $\mathbf{p} \in P'$;
  if $\mathbf{w}.\mathbf{p} < 0$ **then**
  end
end
//the algorithm converges when all the inputs are classified correctly

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-;$
Initialize $\mathbf{w}$ randomly;
while !*convergence* **do**
    Pick random $\mathbf{p} \in P'$;

    if $\mathbf{w}.\mathbf{p} < 0$ **then**
        $\mathbf{w} = \mathbf{w} + \mathbf{p}$;
    end
end
//the algorithm converges when all the inputs are classified correctly
//notice that we do not need the other **if** condition because by construction we want all points in P' to lie in the positive half space $\mathbf{w}.\mathbf{p} \geq 0$

Setup:

- If $x \in N$ then $-x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$
- Further we will normalize all the p's so that $||p|| = 1$ (notice that this does not affect the solution $\because if \quad w^T \frac{p}{||p||} \geq 0$ then $w^T p \geq 0$)

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-;$
Initialize $\mathbf{w}$ randomly;
while !*convergence* **do**
    Pick random $\mathbf{p} \in P'$;

    if $\mathbf{w.p} < 0$ **then**
        $\mathbf{w} = \mathbf{w} + \mathbf{p}$;
    end
end
//the algorithm converges when all the inputs are classified correctly
//notice that we do not need the other **if** condition because by construction we want all points in P' to lie in the positive half space $\mathbf{w.p} \geq 0$

Setup:

- If $x \in N$ then -$x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$
- Further we will normalize all the p's so that $||p|| = 1$ (notice that this does not affect the solution $\because if \quad w^T \frac{p}{||p||} \geq 0$ then $w^T p \geq 0$)

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-$;
Initialize $\mathbf{w}$ randomly;
while !*convergence* **do**
    Pick random $\mathbf{p} \in P'$;
    $\mathbf{p} \leftarrow \frac{p}{||p||} \quad (so\ now, ||p|| = 1)$;
    if $\mathbf{w}.\mathbf{p} < 0$ **then**
        $\mathbf{w} = \mathbf{w} + \mathbf{p}$;
    end
end
//the algorithm converges when all the inputs are classified correctly
//notice that we do not need the other **if** condition because by construction we want all points in P' to lie in the positive half space $\mathbf{w}.\mathbf{p} \geq 0$

Setup:

- If $x \in N$ then $-x \in P$ ($\because w^T x < 0 \implies w^T(-x) \geq 0$)
- We can thus consider a single set $P' = P \cup N^-$ and for every element $p \in P'$ ensure that $w^T p \geq 0$
- Further we will normalize all the p's so that $||p|| = 1$ (notice that this does not affect the solution $\because if \quad w^T \frac{p}{||p||} \geq 0$ then $w^T p \geq 0$)
- Let w^* be the normalized solution vector (we know one exists as the data is linearly separable)

Algorithm: Perceptron Learning Algorithm

$P \leftarrow inputs \quad with \quad label \quad 1;$
$N \leftarrow inputs \quad with \quad label \quad 0;$
N^- *contains negations of all points in* N;
$P' \leftarrow P \cup N^-;$
Initialize $\mathbf{w}$ randomly;
while !*convergence* **do**
  Pick random $\mathbf{p} \in P'$;
  $\mathbf{p} \leftarrow \frac{p}{||p||} \quad (so\ now, ||p|| = 1)$;
  if $\mathbf{w.p} < 0$ **then**
    $\mathbf{w} = \mathbf{w} + \mathbf{p}$;
  end
end
//the algorithm converges when all the inputs are classified correctly
//notice that we do not need the other **if** condition because by construction we want all points in P' to lie in the positive half space $\mathbf{w.p} \geq 0$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$Numerator = w^* \cdot w_{t+1}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$Numerator = w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i)$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$Numerator = w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i)$$
$$= w^* \cdot w_t + w^* \cdot p_i$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned}
Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\
&= w^* \cdot w_t + w^* \cdot p_i \\
&\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\
&\geq w^* \cdot (w_{t-1} + p_j) + \delta \\
&\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta
\end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \\ &\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\ &\geq w^* \cdot w_{t-1} + 2\delta \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \\ &\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\ &\geq w^* \cdot w_{t-1} + 2\delta \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step
- We make a correction only if $w^T \cdot p_i \leq 0$ at that time step

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned}
Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\
&= w^* \cdot w_t + w^* \cdot p_i \\
&\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\
&\geq w^* \cdot (w_{t-1} + p_j) + \delta \\
&\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\
&\geq w^* \cdot w_{t-1} + 2\delta
\end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step
- We make a correction only if $w^T \cdot p_i \leq 0$ at that time step
- So at time-step t we would have made only k $(\leq t)$ corrections

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \\ &\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\ &\geq w^* \cdot w_{t-1} + 2\delta \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step
- We make a correction only if $w^T \cdot p_i \leq 0$ at that time step
- So at time-step t we would have made only k $(\leq t)$ corrections
- Every time we make a correction a quantity δ gets added to the numerator

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i|\forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \\ &\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\ &\geq w^* \cdot w_{t-1} + 2\delta \end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step
- We make a correction only if $w^T \cdot p_i \leq 0$ at that time step
- So at time-step t we would have made only k ($\leq t$) corrections
- Every time we make a correction a quantity δ gets added to the numerator
- So by time-step t, a quantity $k\delta$ gets added to the numerator

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned}
Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\
&= w^* \cdot w_t + w^* \cdot p_i \\
&\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\
&\geq w^* \cdot (w_{t-1} + p_j) + \delta \\
&\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\
&\geq w^* \cdot w_{t-1} + 2\delta
\end{aligned}$$

Observations:

- w^* is some optimal solution which exists but we don't know what it is
- We do not make a correction at every time-step
- We make a correction only if $w^T \cdot p_i \leq 0$ at that time step
- So at time-step t we would have made only k ($\leq t$) corrections
- Every time we make a correction a quantity δ gets added to the numerator
- So by time-step t, a quantity $k\delta$ gets added to the numerator

Proof:

- Now suppose at time step t we inspected the point p_i and found that $w^T \cdot p_i \leq 0$
- We make a correction $w_{t+1} = w_t + p_i$
- Let β be the angle between w^* and w_{t+1}

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||}$$

$$\begin{aligned} Numerator &= w^* \cdot w_{t+1} = w^* \cdot (w_t + p_i) \\ &= w^* \cdot w_t + w^* \cdot p_i \\ &\geq w^* \cdot w_t + \delta \quad (\delta = min\{w^* \cdot p_i | \forall i\} \\ &\geq w^* \cdot (w_{t-1} + p_j) + \delta \\ &\geq w^* \cdot w_{t-1} + w^* \cdot p_j + \delta \\ &\geq w^* \cdot w_{t-1} + 2\delta \\ &\geq w^* \cdot w_0 + (k)\delta \quad (By\ induction) \end{aligned}$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

Proof (continued:)

So far we have, $\quad w^T \cdot p_i \leq 0 \quad$ *(and hence we made the correction)*

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

Proof (continued:)

$$So\ far\ we\ have, \quad w^T \cdot p_i \leq 0 \quad (and\ hence\ we\ made\ the\ correction)$$

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

$$Denominator^2 = ||w_{t+1}||^2$$

Proof (continued:)

$$So\ far\ we\ have, \quad w^T \cdot p_i \leq 0 \quad (and\ hence\ we\ made\ the\ correction)$$

$$\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i)
\end{aligned}$$

Proof (continued:)

$$So\ far\ we\ have, \quad w^T \cdot p_i \leq 0 \quad (and\ hence\ we\ made\ the\ correction)$$

$$\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2)
\end{aligned}$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$
\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2) \\
&\leq ||w_t||^2 + ||p_i||^2 \quad (\because w_t \cdot p_i \leq 0)
\end{aligned}
$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$
\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2) \\
&\leq ||w_t||^2 + ||p_i||^2 \quad (\because w_t \cdot p_i \leq 0) \\
&\leq ||w_t||^2 + 1 \quad (\because ||p_i||^2 = 1)
\end{aligned}
$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2) \\
&\leq ||w_t||^2 + ||p_i||^2 \quad (\because w_t \cdot p_i \leq 0) \\
&\leq ||w_t||^2 + 1 \quad (\because ||p_i||^2 = 1) \\
&\leq (||w_{t-1}||^2 + 1) + 1
\end{aligned}$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2) \\
&\leq ||w_t||^2 + ||p_i||^2 \quad (\because w_t \cdot p_i \leq 0) \\
&\leq ||w_t||^2 + 1 \quad (\because ||p_i||^2 = 1) \\
&\leq (||w_{t-1}||^2 + 1) + 1 \\
&\leq ||w_{t-1}||^2 + 2
\end{aligned}$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$
\begin{aligned}
cos\beta &= \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition) \\
Numerator &\geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction) \\
Denominator^2 &= ||w_{t+1}||^2 \\
&= (w_t + p_i) \cdot (w_t + p_i) \\
&= ||w_t||^2 + 2w_t \cdot p_i + ||p_i||^2) \\
&\leq ||w_t||^2 + ||p_i||^2 \quad (\because w_t \cdot p_i \leq 0) \\
&\leq ||w_t||^2 + 1 \quad (\because ||p_i||^2 = 1) \\
&\leq (||w_{t-1}||^2 + 1) + 1 \\
&\leq ||w_{t-1}||^2 + 2 \\
&\leq ||w_0||^2 + (k) \quad (By\ same\ observation\ that\ we\ made\ about\ \delta)
\end{aligned}
$$

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

$$Denominator^2 \leq ||w_0||^2 + k \quad (By\ same\ observation\ that\ we\ made\ about\ \delta)$$

Proof (continued:)

$$\textit{So far we have,} \quad w^T \cdot p_i \leq 0 \quad \textit{(and hence we made the correction)}$$

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (\textit{by definition})$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (\textit{proved by induction})$$

$$Denominator^2 \leq ||w_0||^2 + k \quad (\textit{By same observation that we made about } \delta)$$

$$cos\beta \geq \frac{w^* \cdot w_0 + k\delta}{\sqrt{||w_0||^2 + k}}$$

Proof (continued:)

$$So\ far\ we\ have,\quad w^T \cdot p_i \leq 0 \quad (and\ hence\ we\ made\ the\ correction)$$

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

$$Denominator^2 \leq ||w_0||^2 + k \quad (By\ same\ observation\ that\ we\ made\ about\ \delta)$$

$$cos\beta \geq \frac{w^* \cdot w_0 + k\delta}{\sqrt{||w_0||^2 + k}}$$

- $cos\beta$ thus grows proportional to $\sqrt{k}$

Proof (continued:)

$$\textit{So far we have,} \quad w^T \cdot p_i \leq 0 \quad \textit{(and hence we made the correction)}$$

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad \textit{(by definition)}$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad \textit{(proved by induction)}$$

$$Denominator^2 \leq ||w_0||^2 + k \quad \textit{(By same observation that we made about } \delta\textit{)}$$

$$cos\beta \geq \frac{w^* \cdot w_0 + k\delta}{\sqrt{||w_0||^2 + k}}$$

- $cos\beta$ thus grows proportional to $\sqrt{k}$
- As k (number of corrections) increases $cos\beta$ can become arbitrarily large

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

$$Denominator^2 \leq ||w_0||^2 + k \quad (By\ same\ observation\ that\ we\ made\ about\ \delta)$$

$$cos\beta \geq \frac{w^* \cdot w_0 + k\delta}{\sqrt{||w_0||^2 + k}}$$

- $cos\beta$ thus grows proportional to $\sqrt{k}$
- As k (number of corrections) increases $cos\beta$ can become arbitrarily large
- But since $cos\beta \leq 1$, k must be bounded by a maximum number

Proof (continued:)

So far we have, $w^T \cdot p_i \leq 0$ *(and hence we made the correction)*

$$cos\beta = \frac{w^* \cdot w_{t+1}}{||w_{t+1}||} \quad (by\ definition)$$

$$Numerator \geq w^* \cdot w_0 + k\delta \quad (proved\ by\ induction)$$

$$Denominator^2 \leq ||w_0||^2 + k \quad (By\ same\ observation\ that\ we\ made\ about\ \delta)$$

$$cos\beta \geq \frac{w^* \cdot w_0 + k\delta}{\sqrt{||w_0||^2 + k}}$$

- $cos\beta$ thus grows proportional to $\sqrt{k}$
- As k (number of corrections) increases $cos\beta$ can become arbitrarily large
- But since $cos\beta \leq 1$, k must be bounded by a maximum number
- Thus, there can only be a finite number of corrections (k) to w and the algorithm will converge!

Coming back to our questions ...

- What about non-boolean (say, real) inputs?
- Do we always need to hand code the threshold?
- Are all inputs equal? What if we want to assign more weight (importance) to some inputs?
- What about functions which are not linearly separable ?

Coming back to our questions ...

- What about non-boolean (say, real) inputs? Real valued inputs are allowed in perceptron
- Do we always need to hand code the threshold?
- Are all inputs equal? What if we want to assign more weight (importance) to some inputs?
- What about functions which are not linearly separable ?

Coming back to our questions ...

- What about non-boolean (say, real) inputs? Real valued inputs are allowed in perceptron
- Do we always need to hand code the threshold? No, we can learn the threshold
- Are all inputs equal? What if we want to assign more weight (importance) to some inputs?
- What about functions which are not linearly separable ?

Coming back to our questions ...

- What about non-boolean (say, real) inputs? Real valued inputs are allowed in perceptron
- Do we always need to hand code the threshold? No, we can learn the threshold
- Are all inputs equal? What if we want to assign more weight (importance) to some inputs? A perceptron allows weights to be assigned to inputs
- What about functions which are not linearly separable ?

Coming back to our questions ...

- What about non-boolean (say, real) inputs? Real valued inputs are allowed in perceptron
- Do we always need to hand code the threshold? No, we can learn the threshold
- Are all inputs equal? What if we want to assign more weight (importance) to some inputs? A perceptron allows weights to be assigned to inputs
- What about functions which are not linearly separable ? Not possible with a single perceptron but we will see how to handle this ..