CSE 151 Machine Learning
Instructor: Kamalika Chaudhuri
Announcements

Homework 2 due tomorrow, 5pm

Midterm 2 on May 21st in class

Midterm 2 will be harder than midterm 1

Material covered in midterm 2:
  decision trees, perceptron, kernels, and comparisons to k-NN
Linear Classification

Given labeled data:

\[(x_i, \ y_i) \quad i=1,\ldots,n\]

where \(y\) is \(+1\) or \(-1\), find a hyperplane to separate \(+1\) from \(-1\).
Linear Classification

Given labeled data \((x_i, y_i), i=1,...,n\), where \(y\) is \(+1\) or \(-1\),
Find a hyperplane through the origin to separate \(+\) from \(-\)

\[ w: \text{normal vector to the hyperplane} \]

For a point \(x\) on one side of the hyperplane,
\[ \langle w, x \rangle > 0 \]

For a point \(x\) on the other side,
\[ \langle w, x \rangle < 0 \]
Linear Classification Process

How do you classify a test example $x$?
Output $y = \text{sign}(\langle w, x \rangle)$
The Perceptron Algorithm

**Goal:** Given labeled data \((x_i, y_i), i=1,..,n\), where \(y\) is \(+1\) or \(-1\),
Find a vector \(w\) such that the corresponding hyperplane separates + from -

**Perceptron Algorithm:**

1. Initially, \(w_1 = y_1 x_1\)
2. For \(t=2, 3,\ldots\)
   - If \(y_t \langle w_t, x_t \rangle \leq 0\) then:
     \[ w_{t+1} = w_t + y_t x_t \]
   - Else:
     \[ w_{t+1} = w_t \]

What does \(y_t \langle w_t, x_t \rangle < 0\) mean?
- When \(y_t = 1\), \(y_t \langle w_t, x_t \rangle < 0\) means \(\langle w_t, x_t \rangle < 0\)
- When \(y_t = -1\), \(y_t \langle w_t, x_t \rangle < 0\) means \(\langle w_t, x_t \rangle > 0\)

This means \(w_t\) predicts the label of \(x_t\) incorrectly
Linear Separability

- **Linearly separable**
- **Not linearly separable**
Measure of Separability: Margin

For a unit vector $w$, and training set $S$, margin of $w$ is defined as:

$$
\gamma = \min_{x \in S} \frac{|\langle w, x \rangle|}{\|x\|}
$$

- min cosine of angle between $w$ and $x$, for any $x$ in $S$

**Low Margin**

**High Margin**
Measure of Separability: Margin

For a unit vector $w$, and training set $S$, margin of $w$ is defined as:

$$\gamma = \min_{x \in S} \frac{|\langle w, x \rangle|}{\|x\|}$$

min cosine of angle between $w$ and $x$, for any $x$ in data

Low Margin

High Margin
Real Examples: Margin

For a unit vector $w$, and training set $S$, margin of $w$ is defined as:

$$\gamma = \min_{x \in S} \frac{|\langle w, x \rangle|}{\|x\|}$$

min cosine of angle between $w$ and $x$, for any $x$ in data

Low Margin

High Margin
In Class Problem

Given \( w = (1, 0) \) and examples:

\[
( (1, -1), 1), \quad ( (-1, -1), 1), \quad ( (-1, 0), -1), \quad ( (0.01, 0), 1)
\]

What is the margin of \( w \) wrt the examples?
When does Perceptron work?

**Theorem:**
If the training points are linearly separable with margin $\gamma$ and if $||x_i|| = 1$ for all $x_i$ in the training data, then #mistakes made by perceptron is at most $1/\gamma^2$.
When does Perceptron work?

**Theorem:**
If the training points are linearly separable with margin \( \gamma \) and if \( ||x_i|| = 1 \) for all \( x_i \) in the training data, then the number of mistakes made by perceptron is at most \( 1/\gamma^2 \).

**Observations:**
1. Lower the margin, more mistakes
2. May need **more than one pass** over the training data to get a classifier which makes no mistakes on the training set.
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then #mistakes made by perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$. 

When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin \( \gamma \) and if \( \|x_i\| = 1 \) for all \( x_i \) in the training data, then the number of mistakes made by perceptron is at most \( 1/\gamma^2 \).

**Proof:** Let \( w^* \) be a linear separator with margin \( \gamma \) s.t. \( \|w^*\| = 1 \).

**Fact 1.** If there is a mistake at round \( t \), \( \langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma \)
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then the number of mistakes made by the perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$.

**Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

$$\langle w_{t+1}, w^* \rangle = \langle w_t + y_t x_t, w^* \rangle = \langle w_t, w^* \rangle + y_t \langle x_t, w^* \rangle$$

By definition of margin, $y_t \langle x_t, w^* \rangle \geq \gamma$.
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $||x_i|| = 1$ for all $x_i$ in the training data, then the number of mistakes made by Perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $||w^*|| = 1$.

**Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

$\langle w_{t+1}, w^* \rangle = \langle w_t + y_tx_t, w^* \rangle = \langle w_t, w^* \rangle + y_t \langle x_t, w^* \rangle$

By definition of margin, $y_t \langle x_t, w^* \rangle \geq \gamma$

**Fact 2.** If there is a mistake at round $t$, $||w_{t+1}||^2 \leq ||w_t||^2 + 1$.
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then the number of mistakes made by the perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$.

**Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

$$\langle w_{t+1}, w^* \rangle = \langle w_t + y_t x_t, w^* \rangle = \langle w_t, w^* \rangle + y_t \langle x_t, w^* \rangle$$

By definition of margin, $y_t \langle x_t, w^* \rangle \geq \gamma$.

**Fact 2.** If there is a mistake at round $t$, $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$

$$\|w_{t+1}\|^2 = \|w_t + y_t x_t\|^2 = \|w_t\|^2 + y_t^2 \|x_t\|^2 + 2y_t \langle w_t, x_t \rangle$$

As there is a mistake at round $t$, $y_t \langle w_t, x_t \rangle < 0$

Also: $\|x_t\|^2 \leq 1$
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then the number of mistakes made by perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$.

**Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

**Fact 2.** If there is a mistake at round $t$, $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then #mistakes made by perceptron is at most $1/\gamma^2$

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$

**Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$

**Fact 2.** If there is a mistake at round $t$, $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$

After $M$ mistakes, $\langle w_t, w^* \rangle \geq \gamma M$

$\|w_t\| \leq \sqrt{M}$
When does Perceptron work?

**Theorem:** If the training points are linearly separable with margin $\gamma$ and if $\|x_i\| = 1$ for all $x_i$ in the training data, then #mistakes made by perceptron is at most $1/\gamma^2$.

**Proof:** Let $w^*$ be a linear separator with margin $\gamma$ s.t. $\|w^*\| = 1$

- **Fact 1.** If there is a mistake at round $t$, $\langle w_{t+1}, w^* \rangle \geq \langle w_t, w^* \rangle + \gamma$
- **Fact 2.** If there is a mistake at round $t$, $\|w_{t+1}\|^2 \leq \|w_t\|^2 + 1$

After $M$ mistakes, $\langle w_t, w^* \rangle \geq \gamma M$

$$\|w_t\| \leq \sqrt{M}$$

As $w^*$ has norm at most 1, $\gamma M \leq \langle w_t, w^* \rangle \leq \|w_t\| \leq \sqrt{M}$

Solving,

$$M \leq \frac{1}{\gamma^2}$$
What if data is not linearly separable?

Ideally, we want to find a linear separator that makes the \textbf{minimum} number of mistakes on training data

But this is \textbf{NP Hard}
Almost linearly separable

If data is almost linearly separable (few mistakes, close to the boundary), then, perceptron will still work.

But it will never converge to a single \( w \), as we make more and more passes over the training data.
In Class Problem

Given examples:

( (1, 1), 1),  ( (1, -1), -1),  ( (-1, -1), 1),  ( (-1, 1), -1)

show that perceptron never converges on these examples
Voted Perceptron

**Perceptron:**

1. Initially:
   \[ m = 1, w_1 = y_1 x_1 \]
2. For \( t = 2, 3, \ldots \)
   
   \[
   \text{If } y_t \langle w_m, x_t \rangle \leq 0 \text{ then:} \\
   w_{m+1} = w_m + y_t x_t \\
   m = m + 1
   \]
3. Output \( w_m \)

**Voted Perceptron:**

1. Initially:
   \[ m = 1, w_1 = y_1 x_1 \]
2. For \( t = 2, 3, \ldots \)
   
   \[
   \text{If } y_t \langle w_m, x_t \rangle \leq 0 \text{ then:} \\
   w_{m+1} = w_m + y_t x_t \\
   m = m + 1 \\
   c_m = 1
   \]
   
   Else:
   \[ c_m = c_m + 1 \]
3. Output \((w_1, c_1), (w_2, c_2), \ldots, (w_m, c_m)\)
Voted Perceptron:

1. Initially:
   \[ m = 1, \ w_1 = y_1 x_1 \]
2. For \( t = 2, 3, \ldots \):
   
   If \( y_t \langle w_m, x_t \rangle \leq 0 \) then:
   \[ w_{m+1} = w_m + y_t x_t \]
   \[ m = m + 1 \]
   \[ c_m = 1 \]
   
   Else:
   \[ c_m = c_m + 1 \]
3. Output \( (w_1, c_1), (w_2, c_2), \ldots, (w_m, c_m) \)

How to classify example \( x \)?

Output: \( \text{sign} \left( \sum_{i=1}^{m} c_i \text{sign}(\langle w_i, x \rangle) \right) \)

Problem:
Have to store all the classifiers
**Averaged Perceptron**

**Averaged Perceptron:**

1. Initially:
   \[ m = 1, w_1 = y_1 x_1 \]
2. For \( t = 2, 3, \ldots \):
   - If \( y_t \langle w_m, x_t \rangle \leq 0 \) then:
     \[ w_{m+1} = w_m + y_t x_t \]
     \[ m = m + 1 \]
     \[ c_m = 1 \]
   - Else:
     \[ c_m = c_m + 1 \]
3. Output \((w_1, c_1), (w_2, c_2), \ldots, (w_m, c_m)\)

**How to classify example \( x \)?**

Output: \( \text{sign}\left(\sum_{i=1}^{m} c_i w_i, x\right) \)
Voted vs. Averaged

How to classify example x?

Averaged: \( \text{sign}(\langle \sum_{i=1}^{m} c_i w_i, x \rangle) \)

Voted: \( \text{sign}(\sum_{i=1}^{m} c_i \text{sign}(\langle w_i, x \rangle)) \)

Example where voted and averaged are different:
\( w_1 = x_1, \ w_2 = x_2, \ w_3 = x_3, \ c_1 = c_2 = c_3 = 1 \)

Averaged: \( x_1 + x_2 + x_3 > 0 \)
Voted: Any two of \( x_1, x_2, x_3 > 0 \)
Data:
- (4, 0), +1
- (1, 1), -1
- (0, 1), -1
- (-2, -2), +1

Perceptron: An Example
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \(w_1 = (4, 0), c_1 = 1\)
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( w_1 = (4, 0), c_1 = 1 \)
Step 2: \( (x_2, y_2) = ( (1, 1), -1) \)
Perceptron: An Example

Data:
- (4, 0), +1
- (1, 1), -1
- (0, 1), -1
- (-2, -2), +1

Step 1: $w_1 = (4, 0), c_1 = 1$

Step 2: $(x_2, y_2) = (1, 1), -1$

$y_2 \langle w_1, x_2 \rangle < 0$  Mistake!
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( w_1 = (4, 0), c_1 = 1 \)

Step 2: \( (x_2,y_2) = ( (1, 1), -1) \)
\[ y_2 \langle w_1, x_2 \rangle < 0 \text{  Mistake!} \]
Update: \( w_2 = (3, -1), c_2 = 1 \)
**Perceptron: An Example**

Data:

- \((4, 0), +1)\)
- \((1, 1), -1)\)
- \((0, 1), -1)\)
- \((-2, -2), +1)\)

Step 1: \(w_1 = (4, 0), c_1 = 1\)

Step 2: \((x_2, y_2) = (1, 1), -1)\)

\[ y_2 \langle w_1, x_2 \rangle < 0 \quad \text{Mistake!} \]

Update: \(w_2 = (3, -1), c_2 = 1\)

Step 3: \((x_3, y_3) = (0, 1), -1)\)

**Decision Boundary**
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( w_1 = (4, 0), c_1 = 1 \)

Step 2: \( (x_2, y_2) = ( (1, 1), -1) \)

\[ y_2 \langle w_1, x_2 \rangle < 0 \quad \text{Mistake!} \]

Update: \( w_2 = (3, -1), c_2 = 1 \)

Step 3: \( (x_3, y_3) = ( (0, 1), -1) \)

\[ y_3 \langle w_2, x_3 \rangle > 0 \quad \text{Correct!} \]
**Perceptron: An Example**

**Data:**
\[
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)
\]

**Step 1:** \(w_1 = (4, 0), c_1 = 1\)

**Step 2:** \((x_2, y_2) = ( (1, 1), -1)\)

\[y_2 \langle w_1, x_2 \rangle < 0 \quad \text{Mistake!}\]

Update: \(w_2 = (3, -1), c_2 = 1\)

**Step 3:** \((x_3, y_3) = ( (0, 1), -1)\)

\[y_3 \langle w_2, x_3 \rangle > 0 \quad \text{Correct!}\]

Update: \(c_2 = 2\)
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( \mathbf{w}_1 = (4, 0), c_1 = 1 \)

Step 2: \((x_2, y_2) = ( (1, 1), -1)\)
\[ y_2 \langle \mathbf{w}_1, x_2 \rangle < 0 \quad \text{Mistake!} \]
Update: \( \mathbf{w}_1 = (3, -1), c_2 = 1 \)

Step 3: \((x_3, y_3) = ( (0, 1), -1)\)
\[ y_3 \langle \mathbf{w}_2, x_3 \rangle > 0 \quad \text{Correct!} \]
Update: \( c_2 = 2 \)

Step 4: \((x_4, y_4) = ( (-2, -2), +1)\)
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( w_1 = (4, 0), c_1 = 1 \)

Step 2: \((x_2, y_2) = ( (1, 1), -1)\)
\[ y_2 \langle w_1, x_2 \rangle < 0 \quad \text{Mistake!} \]
Update: \( w_2 = (3, -1), c_2 = 1 \)

Step 3: \((x_3, y_3) = ( (0, 1), -1)\)
\[ y_3 \langle w_2, x_3 \rangle > 0 \quad \text{Correct!} \]
Update: \( c_2 = 2 \)

Step 4: \((x_4, y_4) = ( (-2, -2), +1)\)
\[ y_4 \langle w_2, x_4 \rangle < 0 \quad \text{Mistake!} \]
**Perceptron: An Example**

Data:
- \((4, 0), +1\), \((1, 1), -1\), 
- \((0, 1), -1\), \((-2,-2), +1\)

Step 1: \(w_1 = (4, 0), c_1 = 1\)

Step 2: \((x_2,y_2) = (1, 1), -1\)
- \(y_2 \langle w_1, x_2 \rangle < 0\) **Mistake!**
  - Update: \(w_2 = (3, -1), c_2 = 1\)

Step 3: \((x_3,y_3) = (0, 1), -1\)
- \(y_3 \langle w_2, x_3 \rangle > 0\) **Correct!**
  - Update: \(c_2 = 2\)

Step 4: \((x_4,y_4) = (-2, -2), +1\)
- \(y_4 \langle w_2, x_4 \rangle < 0\) **Mistake!**
  - Update: \(w_3 = (1, -3), c_3 = 1\)

**Decision Boundary**
Perceptron: An Example

Data:
( (4, 0), +1), ( (1, 1), -1),
 ( (0, 1), -1), ( (-2,-2), +1)

Step 1: \( w_1 = (4, 0), c_1 = 1 \)

Step 2: \((x_2, y_2) = ( (1, 1), -1)\)
\[ y_2\langle w_1, x_2 \rangle < 0 \quad \text{Mistake!} \]
Update: \( w_2 = (3, -1), c_2 = 1 \)

Step 3: \((x_3, y_3) = ( (0, 1), -1)\)
\[ y_3\langle w_2, x_3 \rangle > 0 \quad \text{Correct!} \]
Update: \( c_2 = 2 \)

Step 4: \((x_4, y_4) = ( (-2, -2), +1)\)
\[ y_4\langle w_2, x_4 \rangle < 0 \quad \text{Mistake!} \]
Update: \( w_3 = (1, -3), c_3 = 1 \)

Final Voted Perceptron:
\[ \text{sign}(\text{sign}(\langle w_1, x \rangle) + 2\text{sign}(\langle w_2, x \rangle) + \text{sign}(\langle w_3, x \rangle)) \]

Final Averaged Perceptron:
\[ \text{sign}(\langle w_1 + 2w_2 + w_3, x \rangle) \]