Slides adapted from Prof. Carpuat

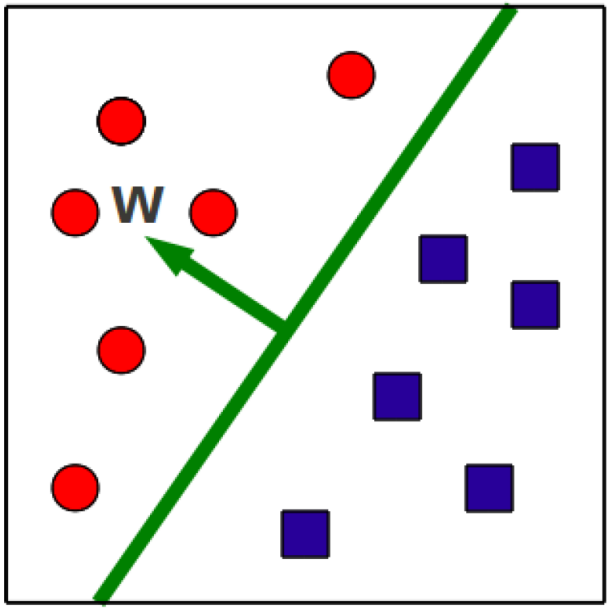
CMSC 422 Introduction to Machine Learning

Lecture 9 The Perceptron

Furong Huang / furongh@umd.edu

UNIVERSITY OF MARYLAND

Recap: Perceptron for binary classification



- Classifier = hyperplane that separates positive from negative examples

$$\hat{y} = sign(w^T x + b)$$

- Perceptron training
  - Finds such a hyperplane
  - Online & error-driven

Learning

- Find algorithm that gives us w and b for a given data set D $(\boldsymbol{x}, y)$
- Once we know w and b we can predict the class of a new data point $\boldsymbol{x}_i$ by evaluating

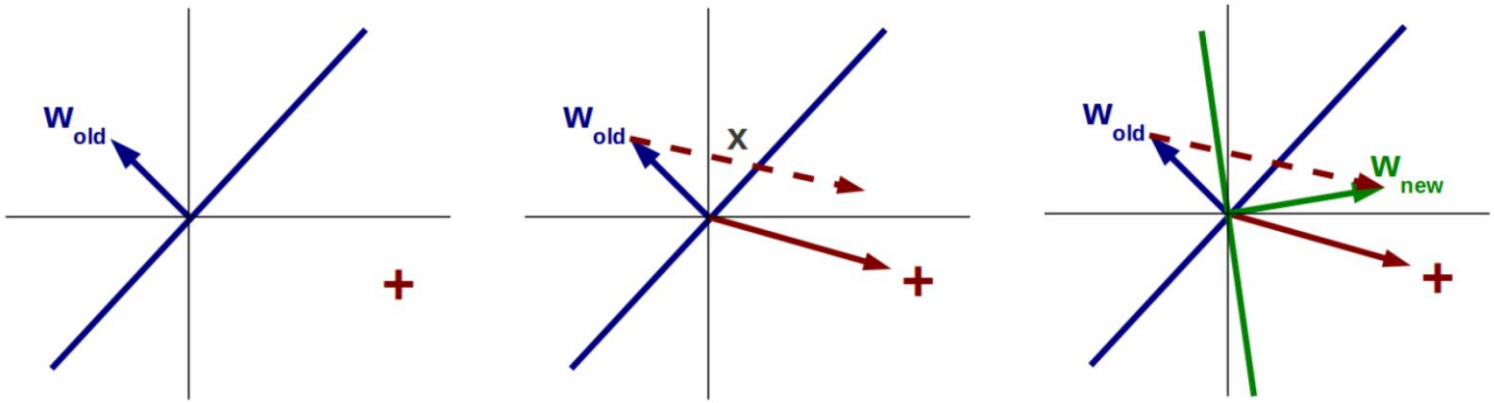
$$\hat{y} = sign(w^\top \boldsymbol{x}_i + b)$$

- We learned a particular way of finding these parameters– via the perceptron update rule
  - Iterative online algorithm–visits all the data over epochs

Recap: Perceptron updates

Update for a misclassified positive example: $y = 1$

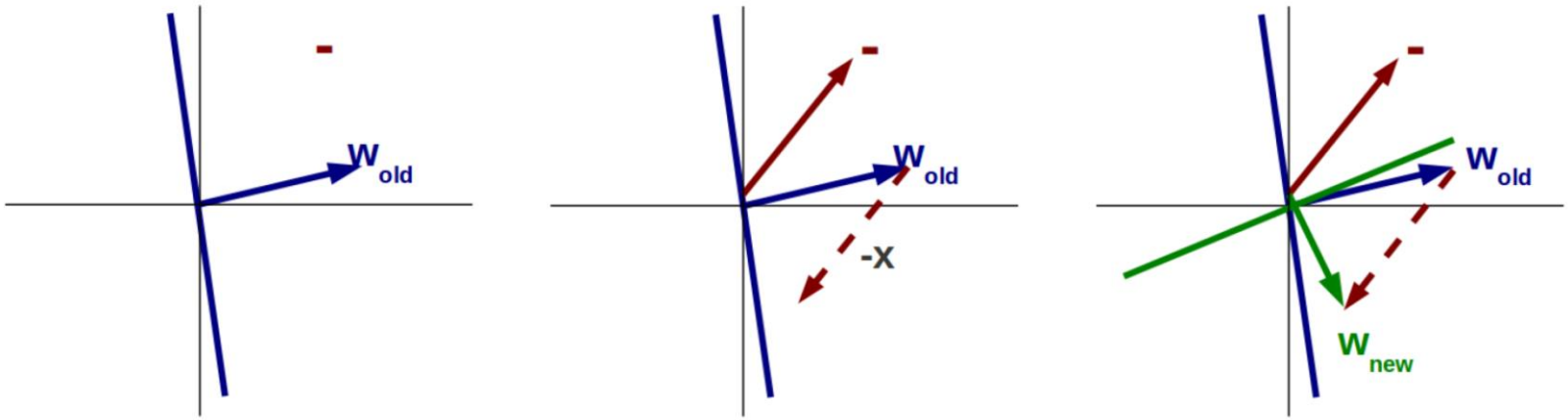
$$\mathbf{w}_{new} = \mathbf{w}_{old} + \mathbf{x}$$



Recap: Perceptron updates

Update for a misclassified negative example: $y = -1$

$$\mathbf{w}_{new} = \mathbf{w}_{old} - \mathbf{x}$$



Today

- Example of perceptron + averaged perceptron training
- Perceptron convergence proof
- Fundamental Machine Learning Concepts
  - Linear separability and margin of a dataset

Averaged perceptron decision rule

$$\hat{y} = \text{sign}\left(\sum_{k=1}^{K} c^{(k)}\left(\boldsymbol{w}^{(k)} \cdot \hat{\boldsymbol{x}} + b^{(k)}\right)\right)$$

can be rewritten as

$$\hat{y} = \text{sign}\left(\left(\sum_{k=1}^{K} c^{(k)}\boldsymbol{w}^{(k)}\right) \cdot \hat{\boldsymbol{x}} + \sum_{k=1}^{K} c^{(k)}b^{(k)}\right)$$

Averaged Perceptron: predict based on average of intermediate parameters

Algorithm 7 AVERAGEDPERCEPTRONTRAIN(**D**, *MaxIter*)

```
1:  w ← ⟨0, 0, ... 0⟩   ,   b ← 0                    // initialize weights and bias
2:  u ← ⟨0, 0, ... 0⟩   ,   β ← 0                    // initialize cached weights and bias
3:  c ← 1                                            // initialize example counter to one
4:  for iter = 1 ... MaxIter do
5:      for all (x,y) ∈ D do
6:          if y(w · x + b) ≤ 0 then
7:              w ← w + y x                          // update weights
8:              b ← b + y                            // update bias
9:              u ← u + y c x                        // update cached weights
10:             β ← β + y c                          // update cached bias
11:         end if
12:         c ← c + 1                                // increment counter regardless of update
13:     end for
14: end for
15: return w - (1/c) u, b - (1/c) β                  // return averaged weights and bias
```

$$\hat{y} = \text{sign}\left(\sum_{k=1}^{K} c^{(k)}\left(\boldsymbol{w}^{(k)} \cdot \hat{\boldsymbol{x}} + b^{(k)}\right)\right) \quad \Leftrightarrow \quad \text{sign}\left(\left(\sum_{k=1}^{K} c^{(k)}\boldsymbol{w}^{(k)}\right) \cdot \hat{\boldsymbol{x}} + \sum_{k=1}^{K} c^{(k)} b^{(k)}\right)$$

Updates

$$w^c \leftarrow w^{c-1} + y^c x^c \, \mathbb{I}\{incorrect\}$$
$$\mu^c \leftarrow \mu^{c-1} + c y^c x^c \, \mathbb{I}\{incorrect\}$$

Note: c is the index of the training example

Therefore,

$$w^c \leftarrow w^0 + \sum_{i=1}^{c} y^i x^i \, \mathbb{I}\{incorrect\}$$
$$\mu^c \leftarrow \mu^0 + \sum_{i=1}^{c} i \, y^i x^i \, \mathbb{I}\{incorrect\}$$

$$w^c - \frac{1}{c}\mu^c = \sum_{i=1}^{c} \frac{c-i}{c} y^i x^i \, \mathbb{I}\{incorrect\}$$

Averaged perceptron decision rule: averaged over history examples

Survival time $\frac{c-i}{c}$

- larger i, i.e., more recent examples, shorter survival time, lower weight
- smaller i, i.e., examples at the beginning, longer survival time, higher weight

Convergence of Perceptron

- The perceptron has converged if it can classify every training example correctly
  - i.e. if it has found a hyperplane that correctly separates positive and negative examples
- Under which conditions does the perceptron converge and how long does it take?

Convergence of Perceptron

Theorem (Block & Novikoff, 1962)

If the training data $D = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ is **linearly separable** with margin γ by a unit norm hyperplane w_* ($||w_*|| = 1$) with $b = 0$,

Then **perceptron training converges after** $\frac{R^2}{\gamma^2}$ **errors** during training

(assuming ($||x|| < R$) for all x).

Margin of a data set *D*

$$margin(\mathbf{D}, \boldsymbol{w}, b) = \begin{cases} \min_{(\boldsymbol{x},y)\in\mathbf{D}} y(\boldsymbol{w}\cdot\boldsymbol{x}+b) & \text{if } \boldsymbol{w} \text{ separates } \mathbf{D} \\ -\infty & \text{otherwise} \end{cases} \quad (4.8)$$

Distance between the hyperplane (w,b) and the nearest point in **D**

$$margin(\mathbf{D}) = \sup_{\boldsymbol{w},b} margin(\mathbf{D}, \boldsymbol{w}, b) \quad (4.9)$$

Largest attainable margin on **D**

Theorem (Block & Novikoff, 1962)

If the training data $D = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ is **linearly separable** with margin γ by a unit norm hyperplane w_* ($||w_*|| = 1$) with $b = 0$, then **perceptron training converges after $\frac{R^2}{\gamma^2}$ errors** during training (assuming $(||x|| < R)$ for all x).

Proof:

- Margin of $\mathbf{w}_*$ on any *arbitrary example* $(\mathbf{x}_n, y_n)$: $\frac{y_n \mathbf{w}_*^T \mathbf{x}_n}{||\mathbf{w}_*||} = y_n \mathbf{w}_*^T \mathbf{x}_n \geq \gamma$
- Consider the $(k+1)^{th}$ mistake: $y_n \mathbf{w}_k^T \mathbf{x}_n \leq 0$, and update $\mathbf{w}_{k+1} = \mathbf{w}_k + y_n \mathbf{x}_n$
- $\mathbf{w}_{k+1}^T \mathbf{w}_* = \mathbf{w}_k^T \mathbf{w}_* + y_n \mathbf{w}_*^T \mathbf{x}_n \geq \mathbf{w}_k^T \mathbf{w}_* + \gamma$ (why is this nice?)
- Repeating iteratively k times, we get $\mathbf{w}_{k+1}^T \mathbf{w}_* > k\gamma$ (1)
- $||\mathbf{w}_{k+1}||^2 = ||\mathbf{w}_k||^2 + 2 y_n \mathbf{w}_k^T \mathbf{x}_n + ||\mathbf{x}||^2 \leq ||\mathbf{w}_k||^2 + R^2$ (since $y_n \mathbf{w}_k^T \mathbf{x}_n \leq 0$)
- Repeating iteratively k times, we get $||\mathbf{w}_{k+1}||^2 \leq kR^2$ (2)

Theorem (Block & Novikoff, 1962)

If the training data $D = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ is **linearly separable** with margin γ by a unit norm hyperplane w_* ($||w_*|| = 1$) with $b = 0$, then **perceptron training converges after $\frac{R^2}{\gamma^2}$ errors** during training (assuming $(||x|| < R)$ for all x).

What does this mean?

- Perceptron converges quickly when margin is large, slowly when it is small
- Bound does not depend on number of training examples N, nor on number of features d
- **Proof guarantees that perceptron converges, but not necessarily to the max margin separator**

What you should know

- Perceptron concepts
  - training/prediction algorithms (standard, voting, averaged)
  - convergence theorem and what practical guarantees it gives us
  - how to draw/describe the decision boundary of a perceptron classifier
- Fundamental ML concepts
  - Determine whether a data set is linearly separable and define its margin
  - Error driven algorithms, online vs. batch algorithms

UNIVERSITY OF
MARYLAND

Furong Huang
4124 IRB, College Park, MD 20740
301.405.8010 / furongh@umd.edu