Introduction to Machine Learning

CSE474/574: Lecture 6

Varun Chandola <chandola@buffalo.edu>

6 Feb 2015

Outline

Outline

Halving Algorithm

- $\xi_0(\mathcal{C}, x) = \{c \in \mathcal{C} : c(x) = 0\}$
- $\xi_1(\mathcal{C}, x) = \{c \in \mathcal{C} : c(x) = 1\}$

$V_0 \leftarrow \mathcal{C}$
for $i = 1, 2, \ldots$ **do**
  if $|\xi_1(V_{i-1}, x^{(i)})| \geq |\xi_0(V_{i-1}, x^{(i)})|$ **then**
    predict $c(x^{(i)}) = 1$
  else
    predict $c(x^{(i)}) = 0$
  end if
  if $c(x^{(i)}) \neq c_*(x^{(i)})$ **then**
    $V_i = V_{i-1} - \xi_{c(x^{(i)})}(\mathcal{C}, x^{(i)})$
  end if
end for

Halving Algorithm Analysis

- Every mistake results in halving of the version space
- Not computationally feasible
  - Need to store and access the version space
- Are there any *efficient implementable* learning algorithms
  - With comparable mistake bounds

Outline

Learning Monotone Disjunctions

- A restricted concept class: **monotone disjunctions** of at most k variables
  - $\mathcal{C} = \{x_{i_1} \vee x_{i_2} \vee \ldots x_{i_k}\}$
  - $|\mathcal{C}|$?
  - Mistake bound $= \log_2 |\mathcal{C}|$
- An efficient algorithm - *Winnow*

Linearly Separable Concepts

- Concept c is linearly separable if $\exists w \in \Re^d, \Theta \in \Re$ such that:

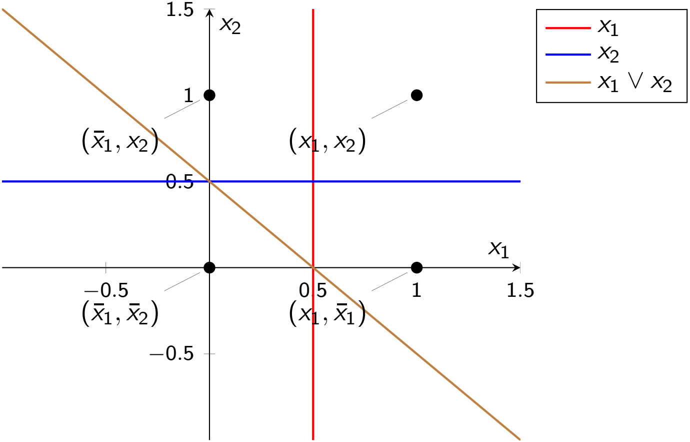
$$\forall x, c(x) = 1 \Leftrightarrow w^\top x \geq \Theta$$

- Monotone disjunctions **are linearly separable**
- For a disjunction $x_{i_1} \vee x_{i_2} \vee \ldots x_{i_k}$

$$x_{i_1} + x_{i_2} + \ldots + x_{i_k} = \frac{1}{2}$$

separates the points labeled 1 and 0 by the disjunctive concept.

A 2D Example


1.5
x_2
1
0.5
-0.5
x_1
-0.5
0.5
1
1.5
$(\bar{x}_1, x_2)$
(x_1, x_2)
$(\bar{x}_1, \bar{x}_2)$
$(x_1, \bar{x}_1)$
x_1
x_2
$x_1 \vee x_2$

Winnow Algorithm

```
Θ ← d/2
w ← (1, 1, ..., 1)
for i = 1, 2, ... do
    if wᵀx^(i) ≥ Θ then
        c(x^(i)) = 1
    else
        c(x^(i)) = 0
    end if
    if c(x^(i)) ≠ c_*(x^(i)) then
        if c_*(x^(i)) = 1 then
            ∀j : x_j^(i) = 1, w_j ← α w_j
        else
            ∀j : x_j^(i) = 1, w_j ← 0
        end if
    end if
end for
```

- *Move* the hyperplane when a mistake is made
- $\alpha > 1$, typically set to 2
- Θ is often set to $\frac{d}{2}$
- *Promotions* and *eliminations*

Analyzing Winnow

- Winnow1 makes $O(k \log_\alpha d)$ mistakes
- Optimal mistake bound
- One can use different values for α and Θ
- Other variants exist
  - *Arbitrary* disjunctions
  - k-DNF (disjunctive normal forms)
    - $(x_1 \wedge x_2) \vee (x_4) \vee (x_7 \wedge \neg x_3)$

Outline

Artificial Neurons

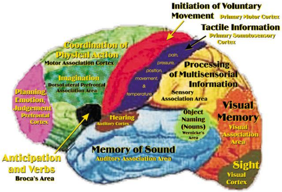


Figure: *Src: http://brainjackimage.blogspot.com/*

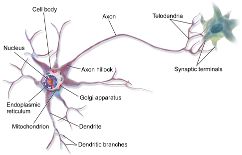
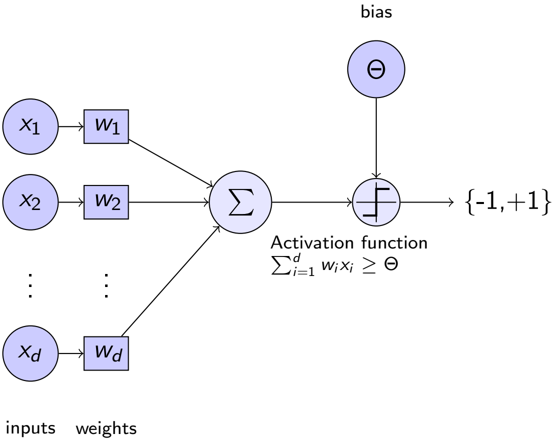


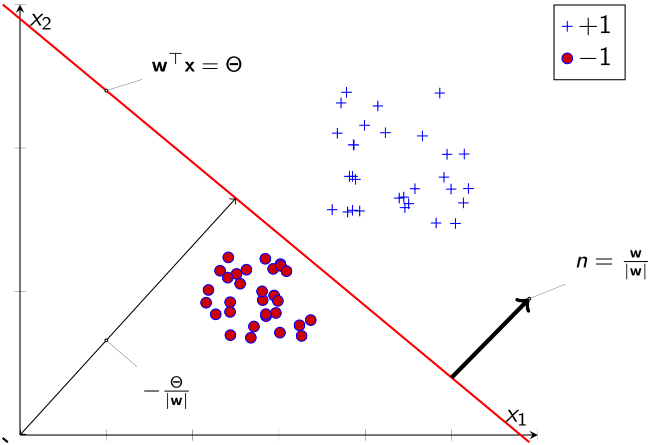
Figure: *Src: Wikipedia*

- Human brain has 10^{11} neurons
- Each connected to 10^4 neighbors

Perceptron [2, 1]



Geometric Interpretation



Eliminating Bias

- Add another attribute $x_{d+1} = 1$.
- w_{d+1} is $-\Theta$
- Desired hyperplane goes through origin in $(d + 1)$ space

Hypothesis Space

- **Assumption**: $\exists \mathbf{w} \in \Re^{d+1}$ such that $\mathbf{w}$ can *strictly* classify all examples correctly.
- *Hypothesis space*: Set of all hyperplanes defined in the $(d+1)$-dimensional space passing through origin
  - The target hypothesis is also called **decision surface** or **decision boundary**.

Perceptron Training - Perceptron Learning Rule

```
w ← (0, 0, ..., 0)_{d+1}
for i=1, 2, ... do
    if w^⊤ x^{(i)} > 0 then
        c(x^{(i)}) = 1
    else
        c(x^{(i)}) = 0
    end if
    if c(x^{(i)}) ≠ c_*(x^{(i)}) then
        w ← w + c_*(x^{(i)}) x^{(i)}
    end if
end for
```

- Every mistake *tweaks* the hyperplane
    - Rotation in $(d+1)$ space
    - Accomodate the offending point
- Stopping Criterion:
    - Exhaust all training example, or
    - No further updates

References

W. Mcculloch and W. Pitts.
A logical calculus of ideas immanent in nervous activity.
Bulletin of Mathematical Biophysics, 5:127–147, 1943.

F. Rosenblatt.
The perceptron: A probabilistic model for information storage and organization in the brain.
Psychological Review, 65(6):386–408, 1958.