CS344: Introduction to Artificial Intelligence (associated lab: CS386)

Pushpak Bhattacharyya
CSE Dept.,
IIT Bombay

Lecture 25: Perceptrons; # of regions; training and convergence

14th March, 2011

Functions in one-input Perceptron

$$w.0 > \theta \quad i.e. \quad 0 > \theta$$
$$w.1 > \theta \quad \quad w > \theta$$

True function

$$w.0 \leq \theta \quad i.e. \quad \theta \geq 0$$
$$w.1 \leq \theta \quad \quad w \leq \theta$$

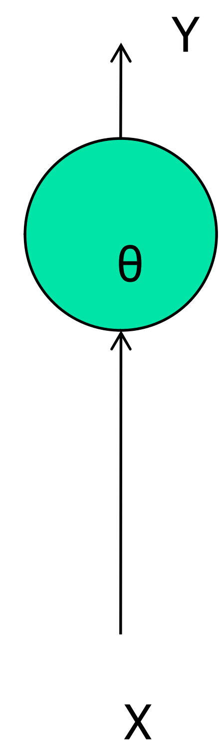
θ function

$$w.0 \leq \theta \quad i.e. \quad \theta \geq 0$$
$$w.1 > \theta \quad \quad w > \theta$$

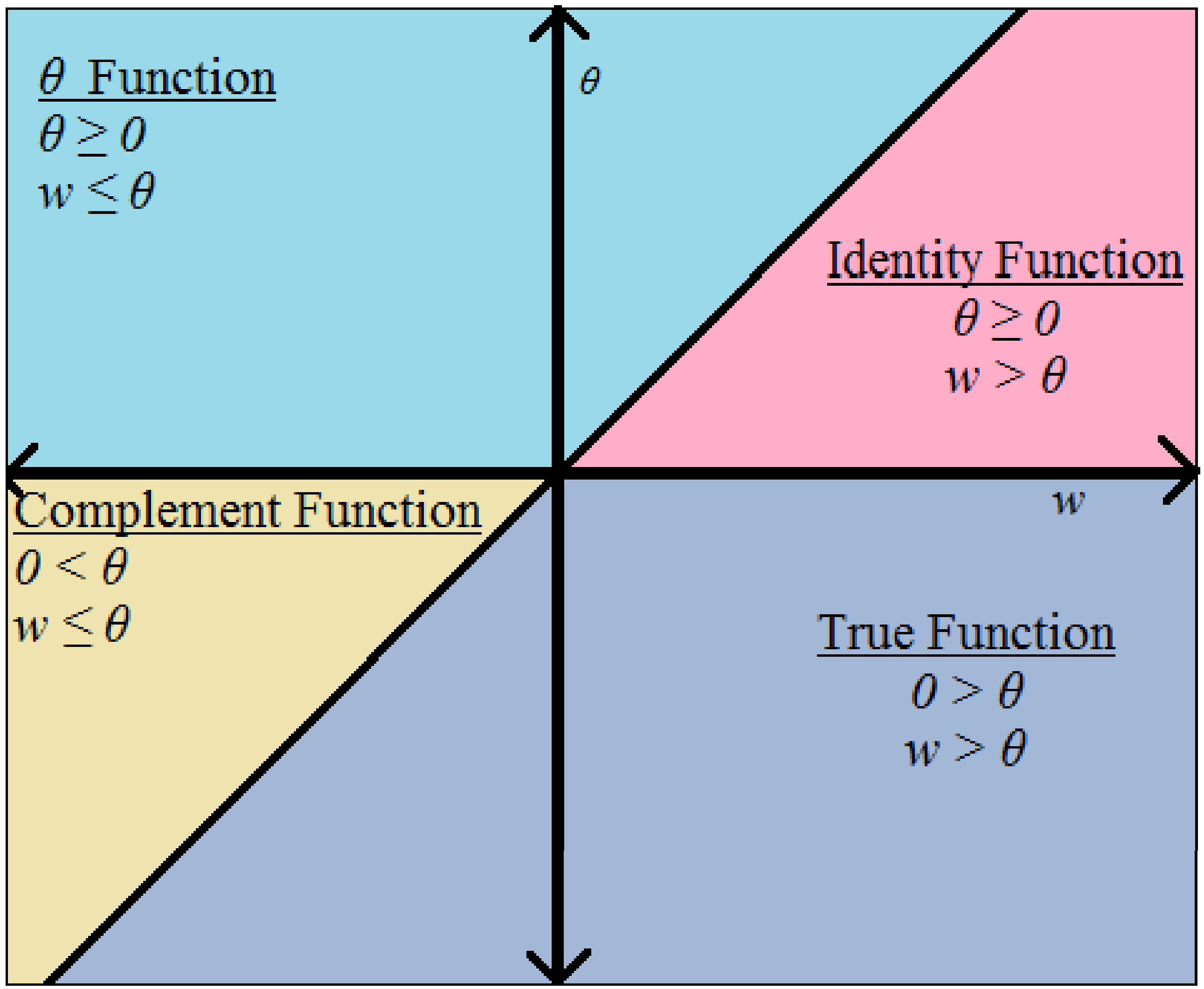
Identity function

$$w.0 > \theta \quad i.e. \quad 0 < \theta$$
$$w.1 \leq \theta \quad \quad w \leq \theta$$

Complement function



Functions in Simple Perceptron



Multiple representations of a concept

The concept of ANDing

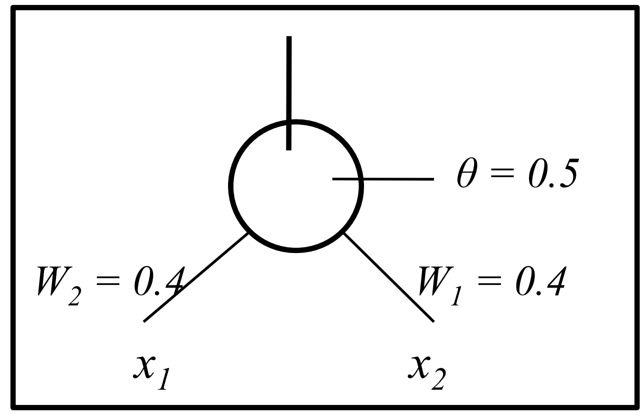
Plain English

AND

Algebra

$y = x_1 \& x_2$

Perceptron



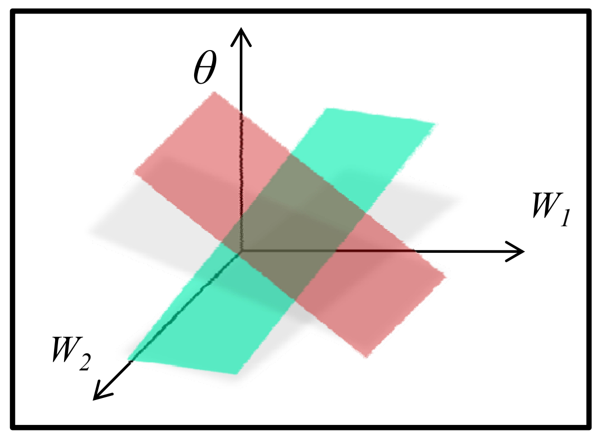
Linear Inequalities

$$\theta \geq 0$$
$$w_1 \leq \theta$$
$$w_2 \leq \theta$$
$$w_1 + w_2 > \theta$$

Truth Table

x_2	x_1	y
0	0	0
0	1	0
1	0	0
1	1	1

Hyperplanes



Inductive Bias

- Once we have decided to use a particular representation, we have assumed "inductive bias"
- The inductive bias of a learning algorithm is the set of assumptions that the learner uses to predict outputs given inputs that it has not encountered (Mitchell, 1980).

- You can refer to:

A theory of the Learnable
LG Valiant - Communications of the ACM, 1984

Fundamental Observation

- The number of TFs computable by a perceptron is equal to the number of regions produced by 2^n hyper-planes,obtained by plugging in the values $<x_1,x_2,x_3,\ldots,x_n>$ in the equation

$$\sum_{i=1}^{n} w_i x_i = \theta$$

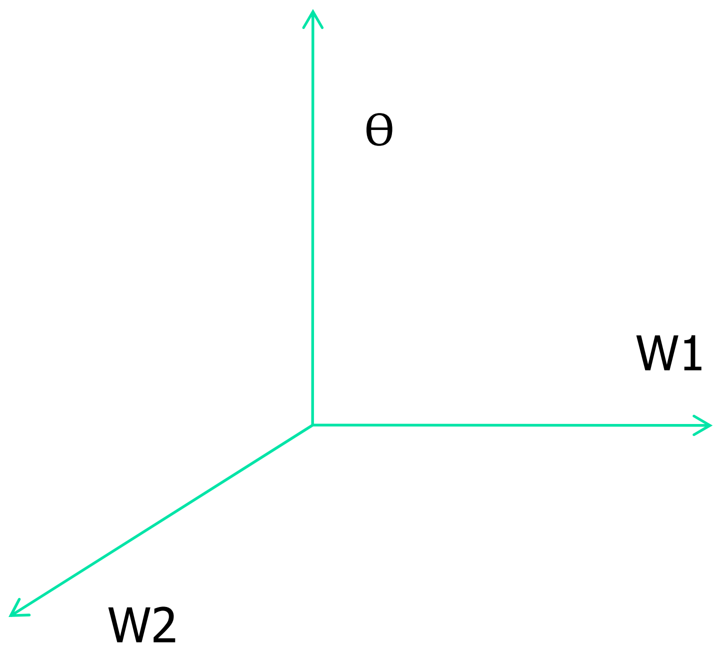
The geometrical observation

- **Problem:** *m* linear surfaces called hyper-planes (each hyper-plane is of *(d-1)*-dim) in d-dim, then what is the max. no. of regions produced by their intersection?

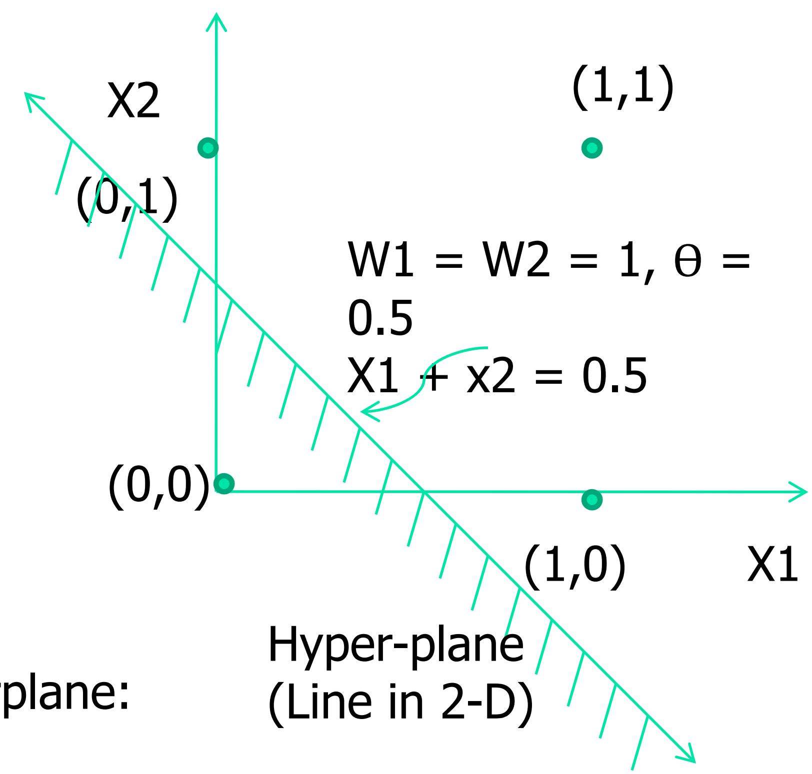
  i.e., $R_{m,d} = ?$

Co-ordinate Spaces

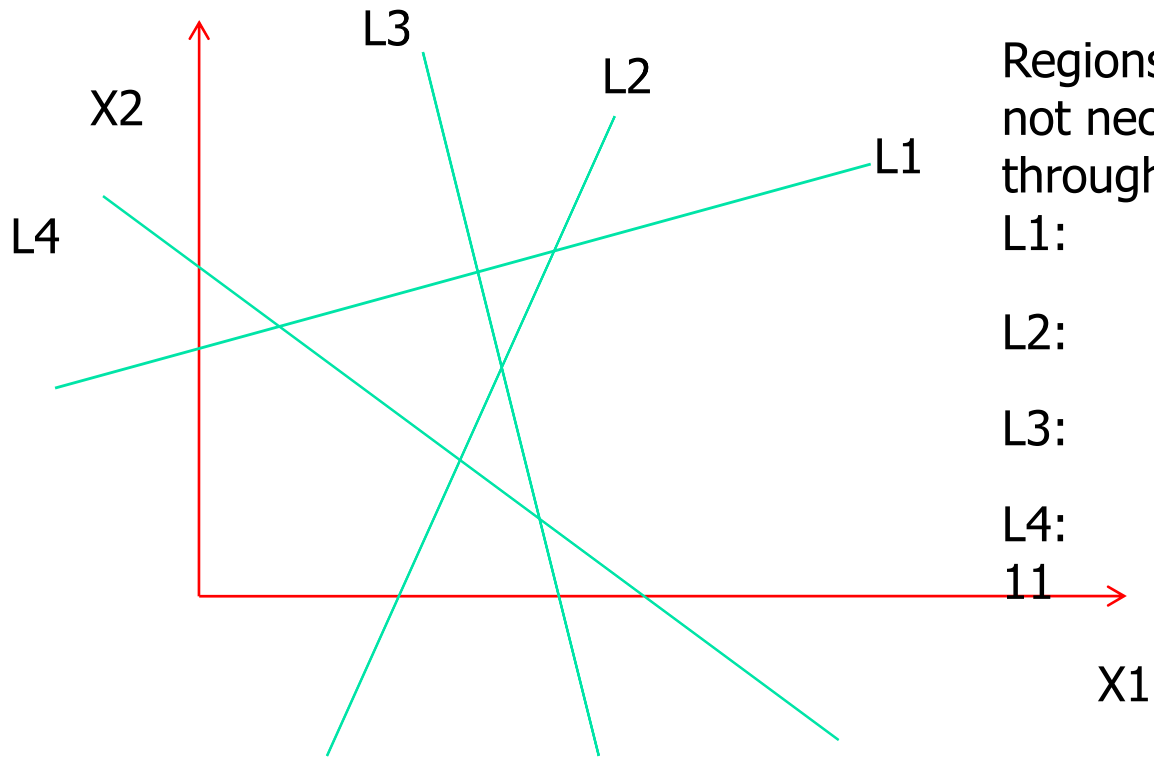
We work in the $<X_1, X_2>$ space or the $<w_1, w_2, \theta>$ space



General equation of a Hyperplane:
Σ Wi Xi = θ



Regions produced by lines



Regions produced by lines not necessarily passing through origin

L1: 2

L2: 2+2 = 4

L3: 2+2+3 = 7

L4: 2+2+3+4 = 11

New regions created = Number of intersections on the incoming line by the original lines

Total number of regions = Original number of regions + New regions created

Number of computable functions by a neuron

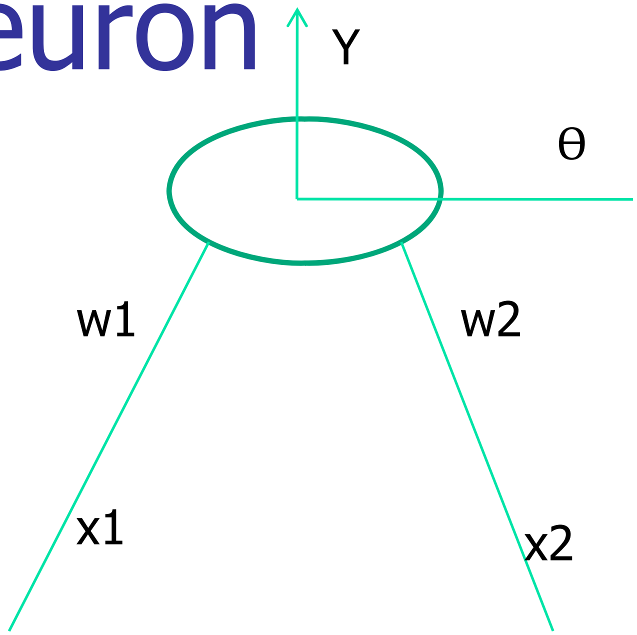
$$w1 * x1 + w2 * x2 = \theta$$

$$(0,0) \Rightarrow \theta = 0 : P1$$

$$(0,1) \Rightarrow w2 = \theta : P2$$

$$(1,0) \Rightarrow w1 = \theta : P3$$

$$(1,1) \Rightarrow w1 + w2 = \theta : P4$$

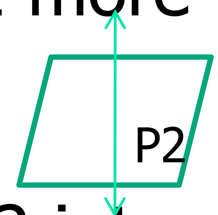


P1, P2, P3 and P4 are planes in the <W1,W2, θ> space

Number of computable functions by a neuron (cont…)

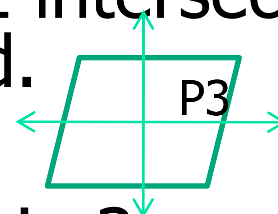
- P1 produces 2 regions
- P2 is intersected by P1 in a line. 2 more new regions are produced.

  Number of regions = 2+2 = 4



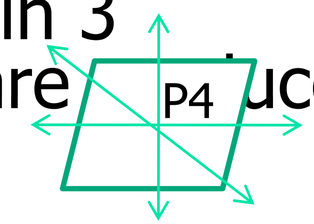
- P3 is intersected by P1 and P2 in 2 intersecting lines. 4 more regions are produced.

  Number of regions = 4 + 4 = 8



- P4 is intersected by P1, P2 and P3 in 3 intersecting lines. 6 more regions are produced.

  Number of regions = 8 + 6 = 14



- Thus, a single neuron can compute 14 Boolean functions which are linearly separable.

Points in the same region

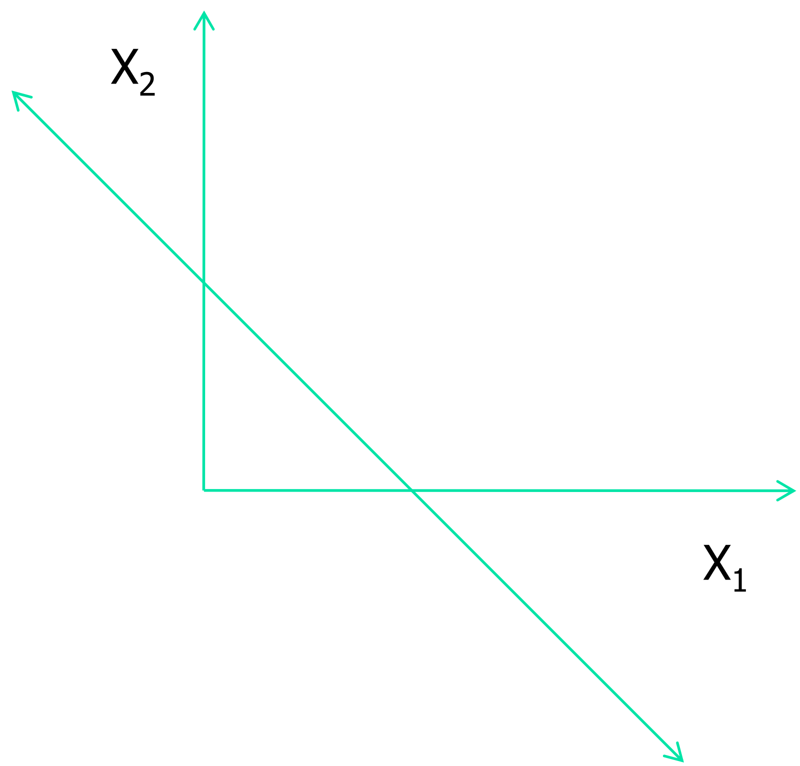
If

$W_1*X_1 + W_2*X_2 > \theta$

$W_1'*X_1 + W_2'*X_2 > \theta'$

Then

If $<W_1,W_2, \theta>$ and $<W_1',W_2', \theta'>$ share a region then they compute the same function



No. of Regions produced by Hyperplanes

Number of regions founded by n hyperplanes in d-dim passing through origin is given by the following recurrence relation

$$R_{n,d} = R_{n-1,d} + R_{n-1,d-1}$$

we use generating function as an operating function

Boundary condition:

$$R_{1,d} = 2$$ 1 hyperplane in d-dim

$$R_{n,1} = 2$$ n hyperplanes in 1-dim,
Reduce to n points thru origin

The generating function is

$$f(x,y) = \sum_{n=1}^{\infty}\sum_{d=1}^{\infty} R_{n,d} \cdot x^n y^d$$

From the recurrence relation we have,

$$R_{n,d} - R_{n-1,d} - R_{n-1,d-1} = 0$$

$R_{n-1,d}$ corresponds to 'shifting' n by 1 place, => multiplication by x

$R_{n-1,d-1}$ corresponds to 'shifting' n and d by 1 place => multiplication by xy

On expanding $f(x,y)$ we get

$$\begin{aligned} f(x,y) = {} & R_{1,1} \cdot xy + R_{1,2} \cdot x y^2 + R_{1,3} \cdot x y^3 + \ldots + R_{1,d} \cdot x y^d + \ldots..\infty \\ & + R_{2,1} \cdot x^2 y + R_{2,2} \cdot x^2 y^2 + R_{2,3} \cdot x^2 y^3 + \ldots + R_{2,d} \cdot x^2 y^d + \ldots..\infty \\ & \ldots.. \\ & + R_{n,1} \cdot x^n y + R_{n,2} \cdot x^n y^2 + R_{n,3} \cdot x^n y^3 + \ldots + R_{n,d} \cdot x^n y^d + \ldots..\infty \end{aligned}$$

$$f(x,y)=\sum_{n=1}^{\infty}\sum_{d=1}^{\infty}R_{n,d}\cdot x^{n}y^{d}$$

$$x\cdot f(x,y)=\sum_{n=1}^{\infty}\sum_{d=1}^{\infty}R_{n,d}\cdot x^{n+1}y^{d}=\sum_{n=2}^{\infty}\sum_{d=1}^{\infty}R_{n-1,d}\cdot x^{n}y^{d}$$

$$xy\cdot f(x,y)=\sum_{n=1}^{\infty}\sum_{d=1}^{\infty}R_{n,d}\cdot x^{n+1}y^{d+1}=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}R_{n-1,d-1}\cdot x^{n}y^{d}$$

$$x\cdot f(x,y)=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}R_{n-1,d}\cdot x^{n}y^{d}+\sum_{n=2}^{\infty}R_{n-1,1}\cdot x^{n}y$$

$$=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}R_{n-1,d}\cdot x^{n}y^{d}+2\cdot\sum_{n=2}^{\infty}x^{n}y$$

$$f(x,y)=\sum_{n=1}^{\infty}\sum_{d=1}^{\infty}R_{n,d}\cdot x^n y^d$$

$$=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}R_{n,d}\cdot x^n y^d+\sum_{d=1}^{\infty}R_{1,d}\cdot xy^d+\sum_{n=1}^{\infty}R_{n,1}\cdot x^n y-R_{1,1}\cdot xy$$

$$=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}R_{n,d}\cdot x^n y^d+2x\cdot\sum_{d=1}^{\infty}xy^d+2y\cdot\sum_{n=1}^{\infty}x^n y-2xy$$

After all this expansion,

$$f(x,y)-x\cdot f(x,y)-xy\cdot f(x,y)$$

$$=\sum_{n=2}^{\infty}\sum_{d=2}^{\infty}(R_{n,d}-R_{n-1,d}-R_{n-1,d-1})x^n y^d$$

$$+2y\cdot\sum_{n=1}^{\infty}x^d-2xy-2y\cdot\sum_{n=2}^{\infty}x+2x\cdot\sum_{d=1}^{\infty}y^d$$

$$=2x\cdot\sum_{d=1}^{\infty}y^d$$

since other two terms become zero

This implies

$$[1-x-xy]f(x,y)=2x\cdot\sum_{d=1}^{\infty}y^{d}$$

$$f(x,y)=\frac{1}{[1-x(1-y)]}\cdot 2x\cdot\sum_{d=1}^{\infty}y^{d}$$

$$=2x\cdot[y+y^{2}+y^{3}+\ldots+y^{d}+\ldots\ldots\infty]\cdot$$

$$[1+x(1+y)+x^{2}(1+y)^{2}+\ldots+x^{d}(1+y)^{d}+\ldots\ldots\infty]$$

also we have,

$$f(x,y)=\sum_{n=1}^{\infty}\sum_{d=1}^{\infty}R_{n,\,d}\cdot x^{n}y^{d}$$

Comparing coefficients of each term in RHS we get, Comparing co-efficients we get

$$R_{n,d} = \sum_{i=0}^{d-1} C_i^{n-1}$$

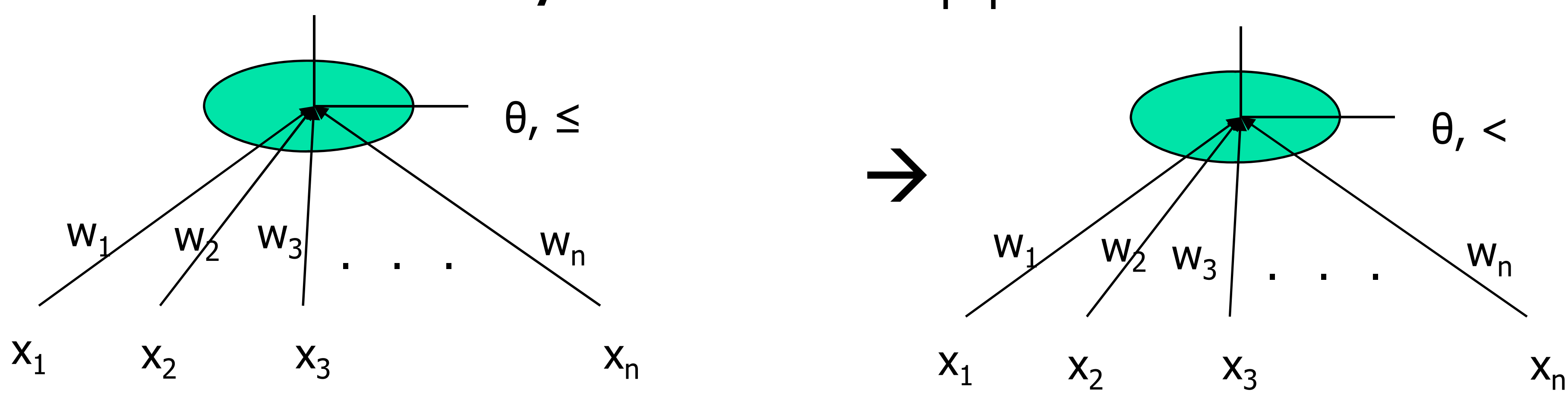
Perceptron Training Algorithm (PTA)

Preprocessing:

1. The computation law is modified to

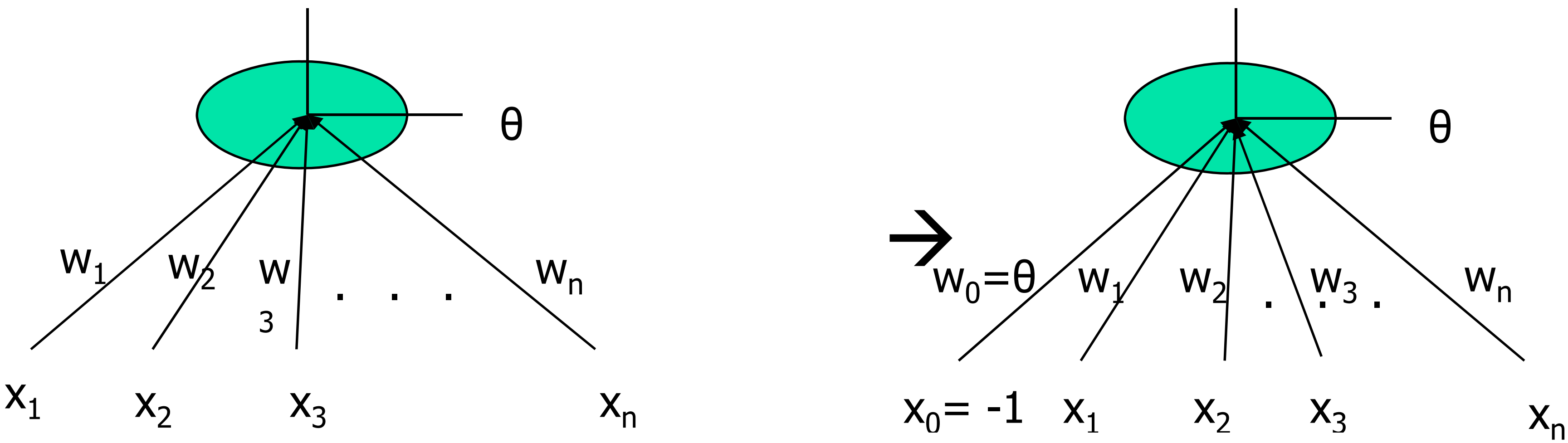
$$y = 1 \text{ if } \sum w_i x_i > \theta$$

$$y = 0 \text{ if } \sum w_i x_i < \theta$$



PTA – preprocessing cont…

2. Absorb θ as a weight



3. Negate all the zero-class examples

Example to demonstrate preprocessing

- **OR perceptron**

1-class   <1,1> , <1,0> , <0,1>

0-class   <0,0>

Augmented x vectors:-

1-class   <-1,1,1> , <-1,1,0> , <-1,0,1>

0-class   <-1,0,0>

Negate 0-class:-   <1,0,0>

Example to demonstrate preprocessing cont..

Now the vectors are

	x_0	x_1	x_2
X_1	-1	0	1
X_2	-1	1	0
X_3	-1	1	1
X_4	1	0	0

Perceptron Training Algorithm

1. Start with a random value of w
   ex: <0,0,0…>
2. Test for $wx_i > 0$
   If the test succeeds for i=1,2,…n
   then return w
3. Modify w, $w_{next} = w_{prev} + x_{fail}$

Tracing PTA on OR-example

w=<0,0,0>	wx_1 fails
w=<-1,0,1>	wx_4 fails
w=<0,0 ,1>	wx_2 fails
w=<-1,1,1>	wx_1 fails
w=<0,1,2>	wx_4 fails
w=<1,1,2>	wx_2 fails
w=<0,2,2>	wx_4 fails
w=<1,2,2>	success