**TMW5: Hebbian Learning**

**# implementation**

x1=[1,1]

x2=[1,-1]

x3=[-1,1]

x4=[-1,-1]

xilist=[x1,x2,x3,x4]

y=[1,-1,-1,-1]

w1=w2=bw=0

b=1

**def** heb\_learn():

**global** w1,w2,bw

print(**"dw1\tdw2\tdb\tw1\tw2\tb"**)

i=0

**for** xi **in** xilist:

dw1=xi[0]\*y[i]

dw2=xi[1]\*y[i]

db=y[i]

w1=w1+dw1

w2=w2+dw2

bw+=db

print(dw1,dw2,db,w1,w2,bw,sep=**'\t'**)

i+=1

print(**"Learning..."**)

heb\_learn()

print(**"Learning completed"**)

print(**"Output of AND gate using obtained w1,w2,bw:"**)

print(**"x1\tx2\ty"**)

**for** xi **in** xilist:

print(xi[0],xi[1],1 **if** w1\*xi[0]+w2\*xi[1]+b\*bw>0 **else** -1,sep=**'\t'**)

print(**"Final weights are: w1="**+str(w1) +**" w2="** +str(w2))

**Output:**

dw1 dw2 db w1 w2 b

1 1 1 1 1 1

-1 1 -1 0 2 0

1 -1 -1 1 1 -1

1 1 -1 2 2 -2

Learning completed

Output of AND gate using obtained w1,w2,bw:

x1 x2 y

1 1 1

1 -1 -1

-1 1 -1

-1 -1 -1

Final weights are: w1=2 w2=2