hw1

October 2020

1 .

eigenvalues of A=$\begin{pmatrix} 2 & 0 & 0 \\ 1 & 2 & 1 \\ -1 & 0 & 1 \end{pmatrix}$

$det(A-\lambda*I)=\begin{vmatrix} 1-\lambda & -3 & 3 \\ 3 & -5-\lambda & 3 \\ 6 & -6 & 4-\lambda \end{vmatrix}$

$=(-\lambda+1)*(-\lambda-5)*(-\lambda+4)+(-3)*3*6+3*3*(-6)-6*(-\lambda-5)*3-(-6)*3*(-\lambda+1)-$
$(-\lambda+4)*3*(-3)=-\lambda^3+12*\lambda+16=-(\lambda+2)*(\lambda^2-2*\lambda-8)=-(\lambda+2)*(\lambda+2)*(\lambda-4)$

$\lambda_1=-2, \lambda_2=4$

For every value we find its own vectors:

$\lambda_1=-2$

$A-\lambda_1*I=\begin{pmatrix} 3 & -3 & 3 \\ 3 & -3 & 3 \\ 6 & -6 & 6 \end{pmatrix}\xrightarrow{Line_2=Line_2-Line_1}\begin{pmatrix} 3 & -3 & 3 \\ 0 & 0 & 0 \\ 6 & -6 & 6 \end{pmatrix}\xrightarrow{Line_3=Line_3-2*Line_3}\begin{pmatrix} 3 & -3 & 3 \\ 0 & 0 & 0 \\ 0 & -0 & 0 \end{pmatrix}$

for $(A-\lambda_1*I)X=0 => X=\begin{pmatrix} x_2-x_3 \\ x_2 \\ x_3 \end{pmatrix}$

The solution set:$\{x_2\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}x_3\begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}\}$

Let $x_2=1, x_3=0, v_1=\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ Let $x_2=0, x_3=1, v_2=\begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$

$\lambda_2=4$

$A-\lambda_2*I=\begin{pmatrix} -3 & -3 & 3 \\ 3 & -9 & 3 \\ 6 & -6 & 0 \end{pmatrix}\xrightarrow{Line_2=Line_2+Line_1}\begin{pmatrix} -3 & -3 & 3 \\ 0 & -12 & 6 \\ 6 & -6 & 0 \end{pmatrix}\xrightarrow{Line_3=Line_3+2*Line_1}\begin{pmatrix} -3 & -3 & 3 \\ 0 & -12 & 6 \\ 0 & -12 & 6 \end{pmatrix}\xrightarrow{Line_3=Line_3-Line_2}$
$\begin{pmatrix} -3 & -3 & 3 \\ 0 & -12 & 6 \\ 0 & 0 & 0 \end{pmatrix}\xrightarrow{Line_1/-3, Line_2/-12}\begin{pmatrix} 1 & 1 & -1 \\ 0 & 1 & \frac{-1}{2} \\ 0 & 0 & 0 \end{pmatrix}$

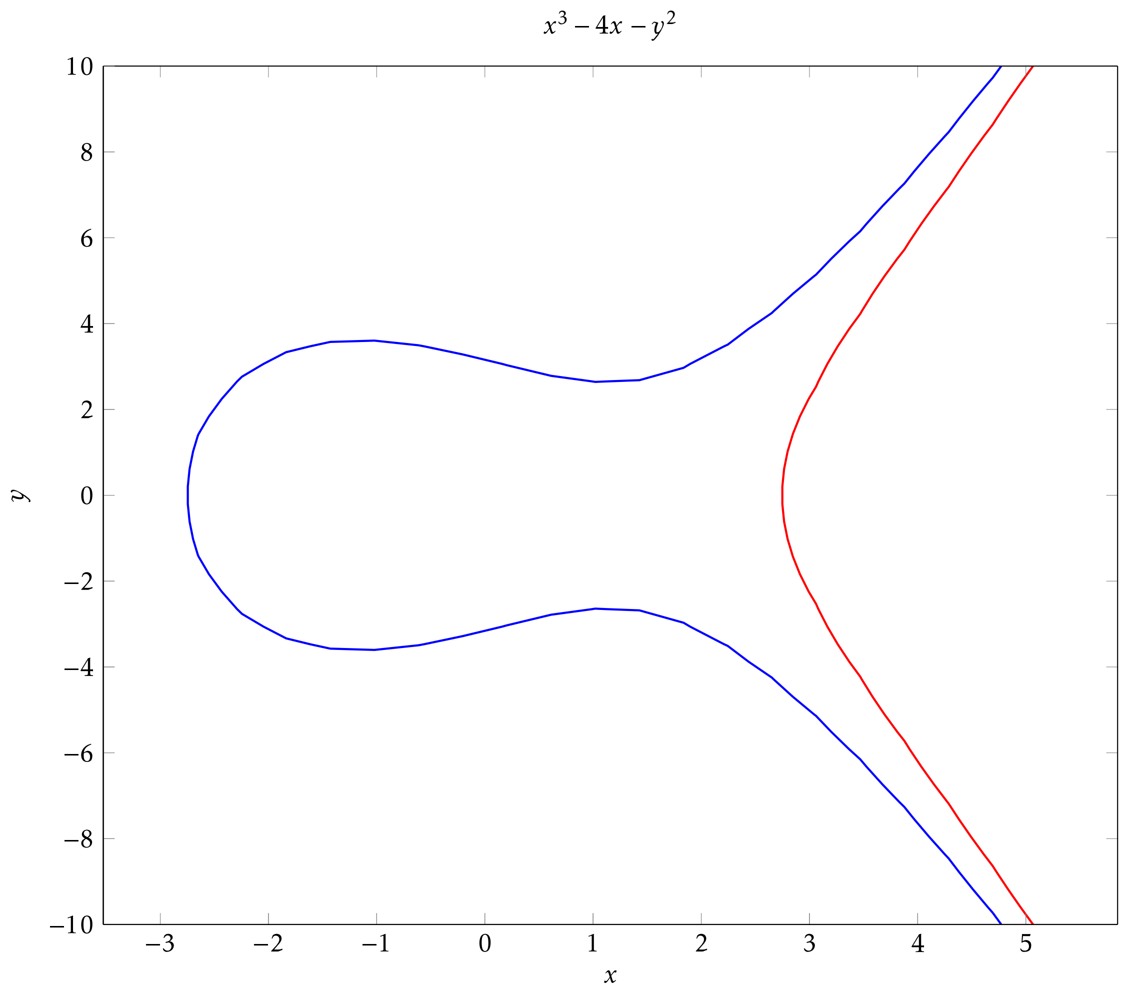
$$\xrightarrow{Line_1=Line_1-Line_2}\begin{pmatrix}1 & 0 & \frac{-1}{2}\\ 0 & 1 & \frac{-1}{2}\\ 0 & 0 & 0\end{pmatrix}$$

So $x_2 = 1/2 * x_3$ and $x_1 = 1/2 * x_3$ => for $(A - \lambda_2 * I)X = 0 \Rightarrow X = \begin{pmatrix}\frac{1}{2} * x_3\\ \frac{1}{2} * x_3\\ x_3\end{pmatrix}$

The solution set:$\{x_3 \begin{pmatrix}1/2\\ 1/2\\ 1\end{pmatrix}$

Let $x_3 = 1, v_3 = \begin{pmatrix}1/2\\ 1/2\\ 1\end{pmatrix}$

2 .



D = fxx(a,b) fyy(a,b) - $fxy^2(a,b)$
a) If D > 0 and fxx (a,b) > 0, then f has a relative minimum at (a,b).
b) If D > 0 and fxx (a,b) < 0, then f has a relative maximum at (a,b).
c) If D < 0, then f has a saddle point at (a,b).
d) If D = 0, then no conclusion can be drawn.

for fx(x,y)=3x^2-4=fy(x,y)=2y=0 =>
y=0,x=2/ $\sqrt[2]{3}$

fxx(x,y)=6x, fyy(x,y)=2 ,fxy(x,y)=0
D=fxx(2,0)fyy(2,0) - $fxy^2(2,0)$=12*2 - 0=24>0 also fxx(2,0)>0 so f has a local minimum at the point (2,0,f(2,0)) = (2,0,0)

3 .

P2)

$$f(x,y) = x^2 + 2x + y^2 + 4$$

$x_0 = (2,1)$

$$\nabla f(x,y) = \begin{pmatrix} 2x+2 \\ 2y \end{pmatrix}$$

$$\|\nabla f(x)\| = \sqrt{\nabla^T f(x) \nabla f(x)} = 2\sqrt{x^2+2x+1+y^2}$$

$$x_0 = (2,1)^T \to 2\sqrt{10}$$

$$\frac{-\nabla f(x_0)}{\|\nabla f(x_0)\|} = -\frac{1}{2\sqrt{10}} \begin{pmatrix} 6 \\ 2 \end{pmatrix} = -\frac{1}{\sqrt{10}} \begin{pmatrix} 3 \\ 1 \end{pmatrix}$$

Apa, $x_1 = (2,1)^T - \lambda_0 \frac{1}{\sqrt{10}} (3,1)^T = \left(2 - \frac{3\lambda_0}{\sqrt{10}},\ 1 - \frac{\lambda_0}{\sqrt{10}}\right)^T$

- $f(x_1) = \left(2 - \frac{3\lambda_0}{\sqrt{10}}\right)^2 + 2\left(2 - \frac{3\lambda_0}{\sqrt{10}}\right) + \left(1 - \frac{\lambda_0}{\sqrt{10}}\right)^2 + 4$

Optimal λ_0:

$$\frac{df(x_1)}{d\lambda_0} = 2\left(2 - \frac{3\lambda_0}{\sqrt{10}}\right)\left(-\frac{3}{\sqrt{10}}\right) - \frac{6}{\sqrt{10}} + 2\left(1 - \frac{\lambda_0}{\sqrt{10}}\right)\left(-\frac{1}{\sqrt{10}}\right) = 0$$

$$\lambda_0 = \sqrt{10} \simeq 3{,}16$$

$$x_1 = (-1,0)^T$$

Second iteration

$x_1 (-1,0)^T \to \|\nabla f(x_1)\| = 0 \Rightarrow \frac{-\nabla f(x_1)}{\|\nabla f(x_1)\|} \to \infty$

So there is only 1 iteration with $x = (-1,0)^T$, $\lambda = \sqrt{10}$

Figure 1:

4 .

A.

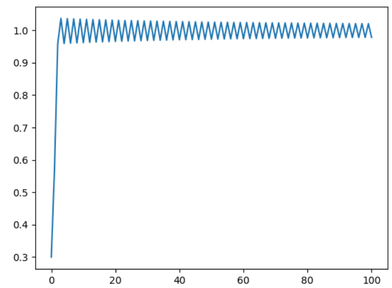


Figure 2: The function becomes periodic with T decreasing,$u = 2, x_0 = 0.3$

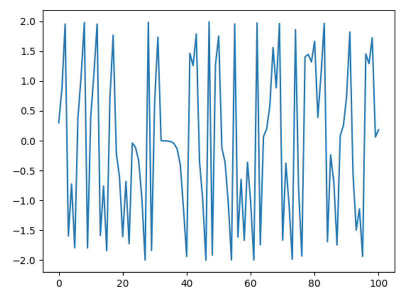


Figure 3: The function becomes chaotic $u = 3, x_0 = 0.3$

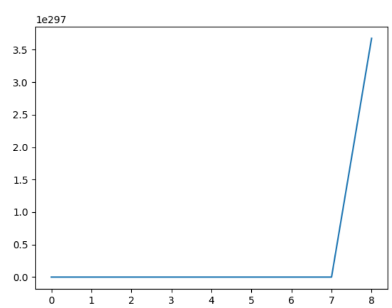


Figure 4: The function becomes chaotic by going to infinity $u = 4, x_0 = 0.3$

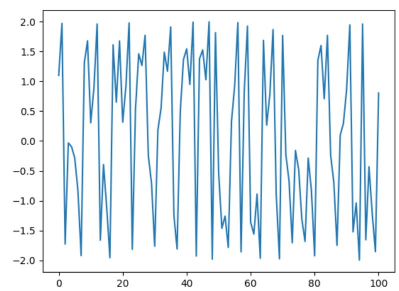


Figure 5: The function becomes chaotic $u = 3, x_0 = 0.3$

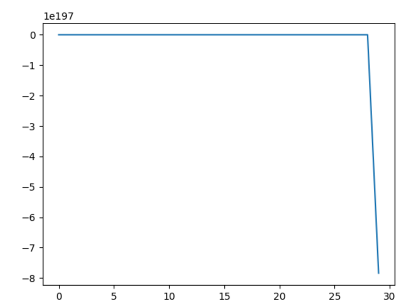


Figure 6: The function becomes chaotic by going to infinity, $u = 3, x_0 = 1.1$

5 .

a)

S'=$\frac{\mathrm{d}}{\mathrm{d}x}\left[\frac{1}{\mathrm{e}^{-x}+1}\right] = -\frac{\mathrm{e}^{-x}}{(\mathrm{e}^{-x}+1)^2} = -\frac{(\mathrm{e}^{-x}+1)-1}{(\mathrm{e}^{-x}+1)^2} = -\frac{1}{\mathrm{e}^{-x}+1} + \frac{1}{(\mathrm{e}^{-x}+1)^2} = S(S-1)$

b)

S=$\frac{\mathrm{d}}{\mathrm{d}x}[\tanh(x)] = \frac{1}{\cosh^2(x)} = \frac{\cosh^2(x)-\sinh^2(x)}{\cosh^2(x)} = 1-\tanh(x)$ =1-S

c)

Set S_a as S from subquestion a),

$S = x * S_a => S' = S_a + x * S_a' = S_a + x(S_a(S_a-1)) = S_a(1-x+xS_a) = \frac{S(1-x+S)}{x}$

d)
Set S_b as S from subquestion b),
$S = x * S_b \Rightarrow S' = S_b + x * S_b' = S_b + x(1 - S_b^2) = S_b(1 - x * S_b) + x = \dfrac{S(1-S) + x^2}{x}$

6 .

$$S(x) = \frac{1}{1 + e^{-cx}}, c > 0 \Rightarrow$$

0< $S(x) < 1$ for every value of x $1 + e^{-cx} = 1/S(x) \Rightarrow e^{-cx} = 1/S(x) - 1 \Rightarrow$

$$x = ln(\frac{S(x)}{-S(x)+1})/c$$

$$\frac{d}{ds}\left[\ln\left(\frac{s}{1-s}\right)\right] = \frac{1-s}{s} \cdot \frac{d}{ds}\left[\frac{s}{1-s}\right] = \frac{+s-s+1}{(1-s)s} = \frac{-1}{(s-1)s} > 0$$

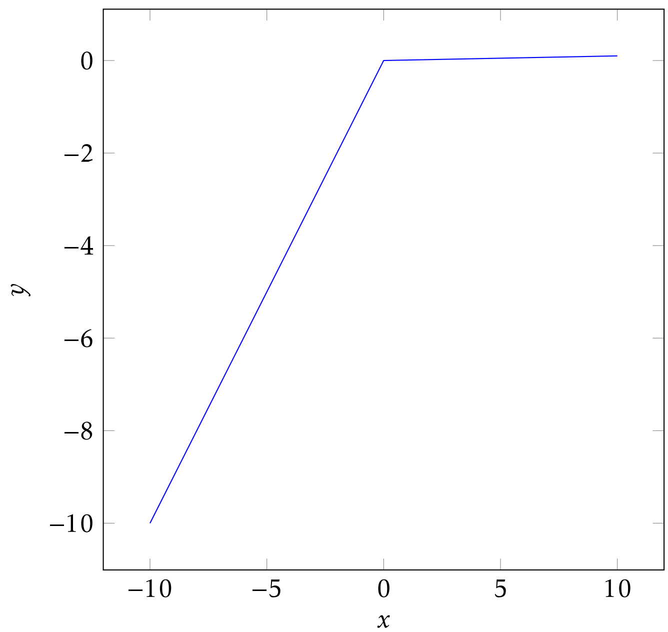
because 0< $s < 1$ So x strictly increases with S

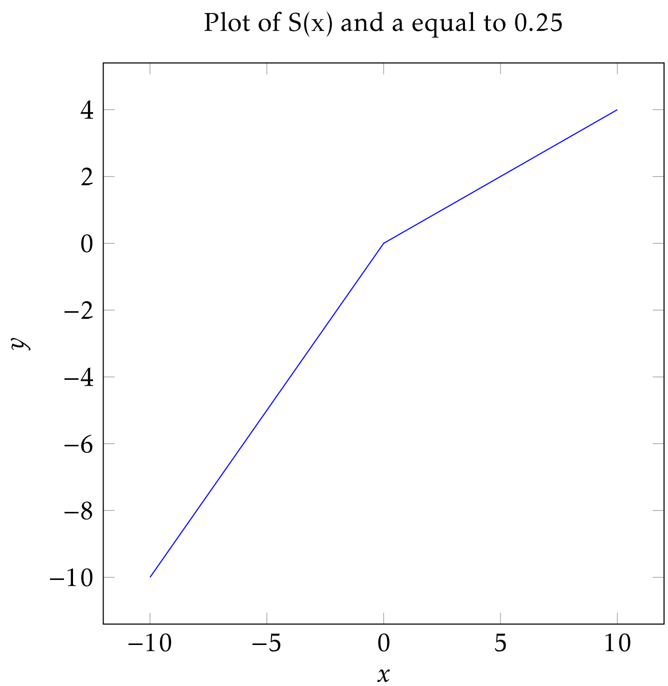
$x = S^{-1}(S(x)) \Rightarrow$
$dx/dx = d(S^{-1}(S(x)))/dx \Rightarrow$
$1 = S^{-1\prime}(S(x)) * S'(x) \Rightarrow$
$1/S'(x) = S^{-1\prime}(S(x))$
Because S'(x)>0 ,$S^{-1\prime}(S(x)) > 0$ so the "inverse function" x increases with S, if S'>0.

7 .

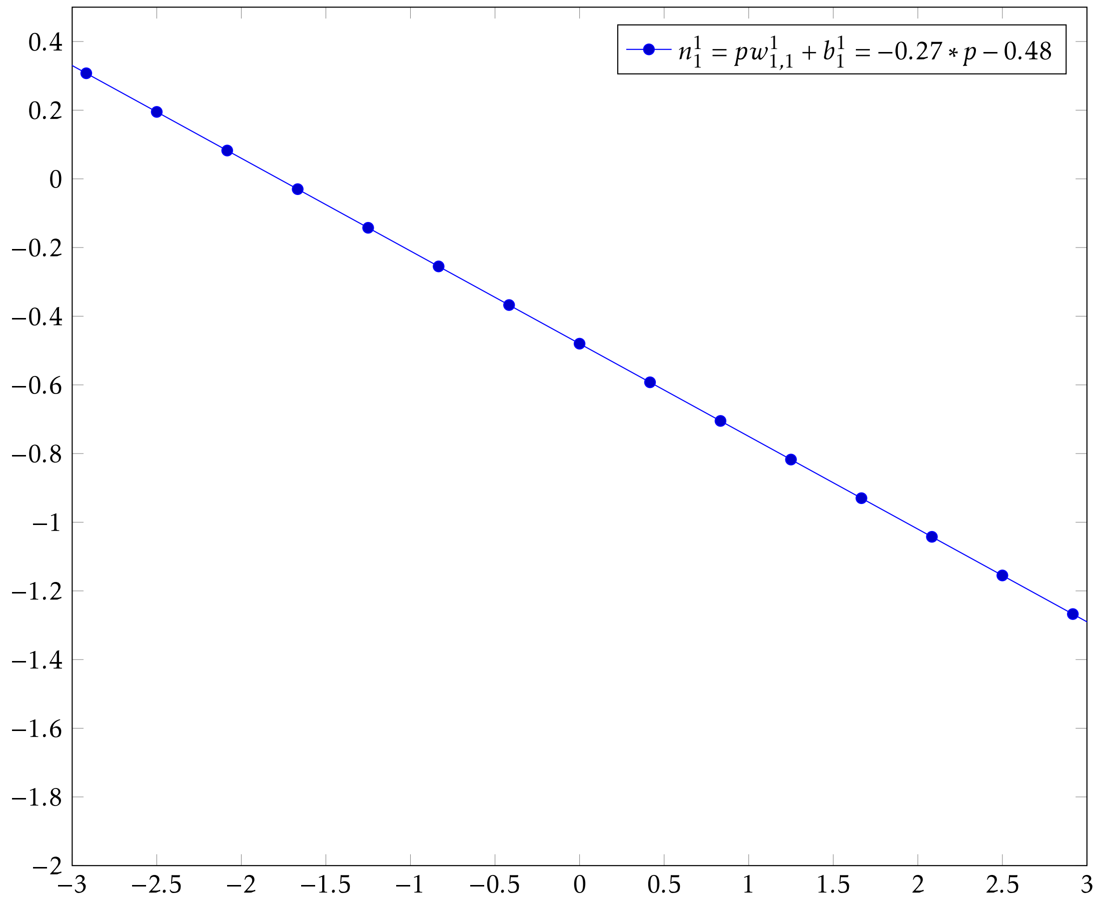
S(x)=max(0,x)+a min(0,x),which for x>0=> S(x)=x=>S'(x)=1, else S(x)=a*x=>S'(x)=a.

Plot of S(x) and a equal to 0.01

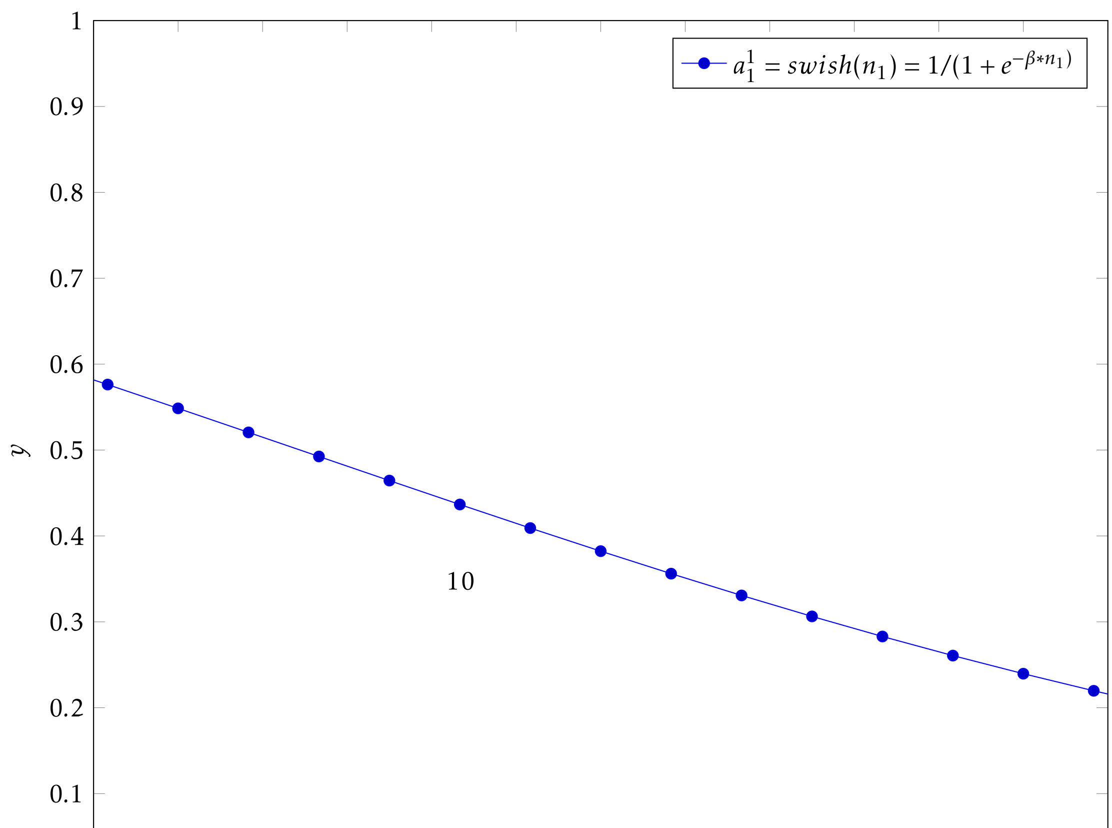



Plot of S(x) and a equal to 0.25
4
2
0
−2
−4
−6
−8
−10
y
−10
−5
0
5
10
x

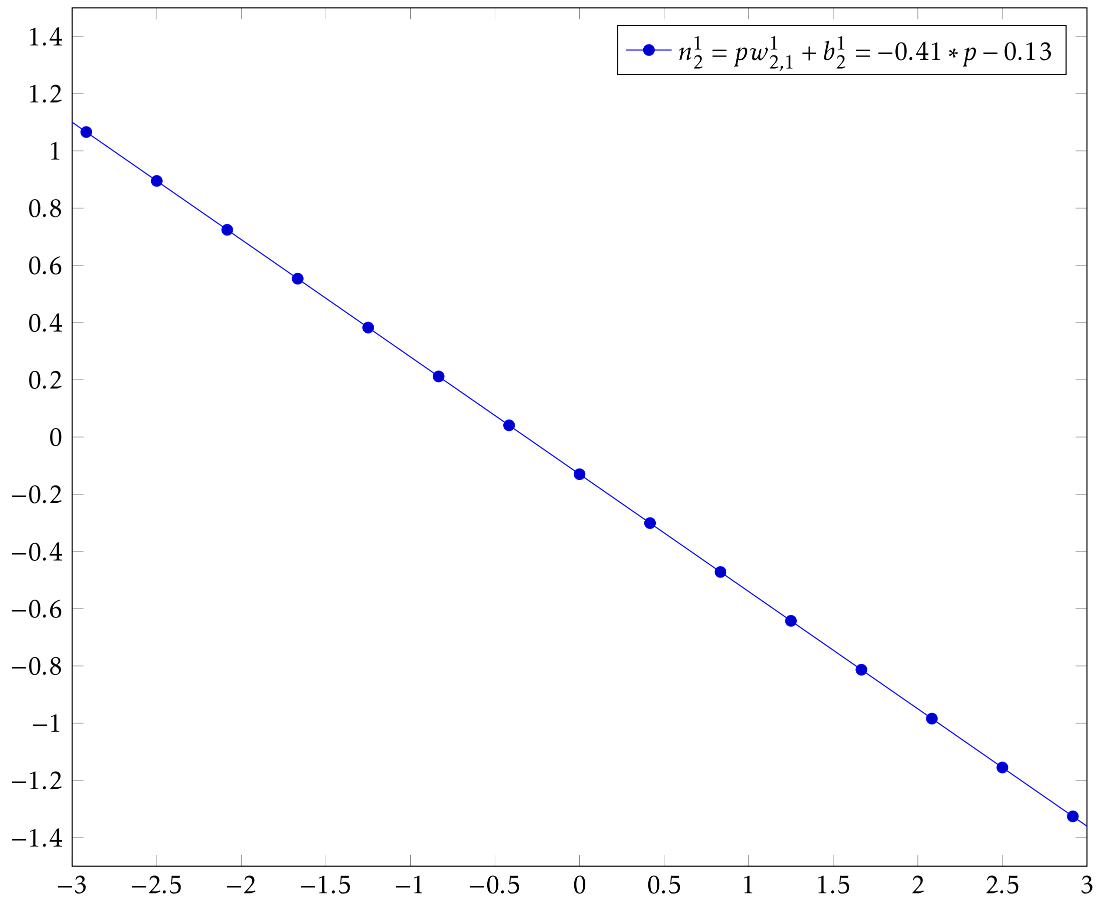
8 .

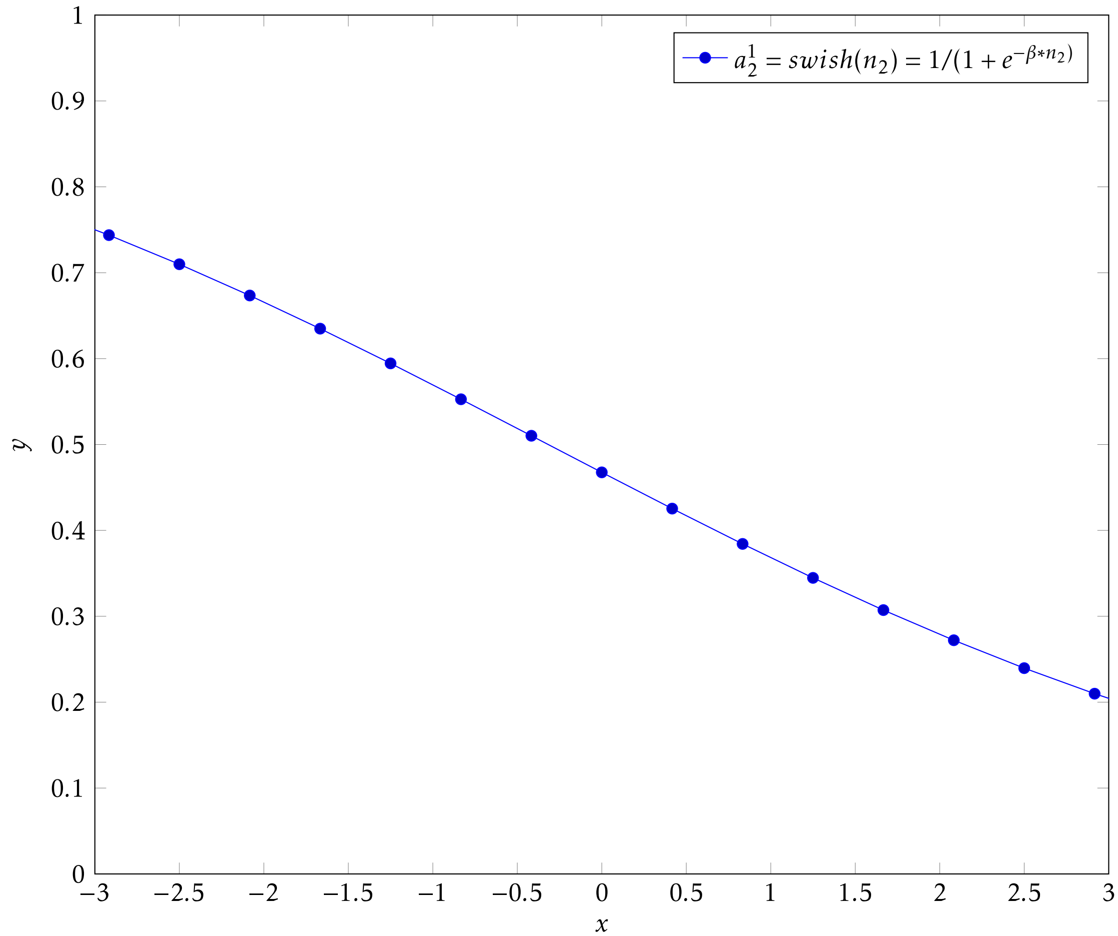


$n_1^1 = pw_{1,1}^1 + b_1^1 = -0.27 * p - 0.48$
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1
−1.2
−1.4
−1.6
−1.8
−2
−3
−2.5
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
2.5
3

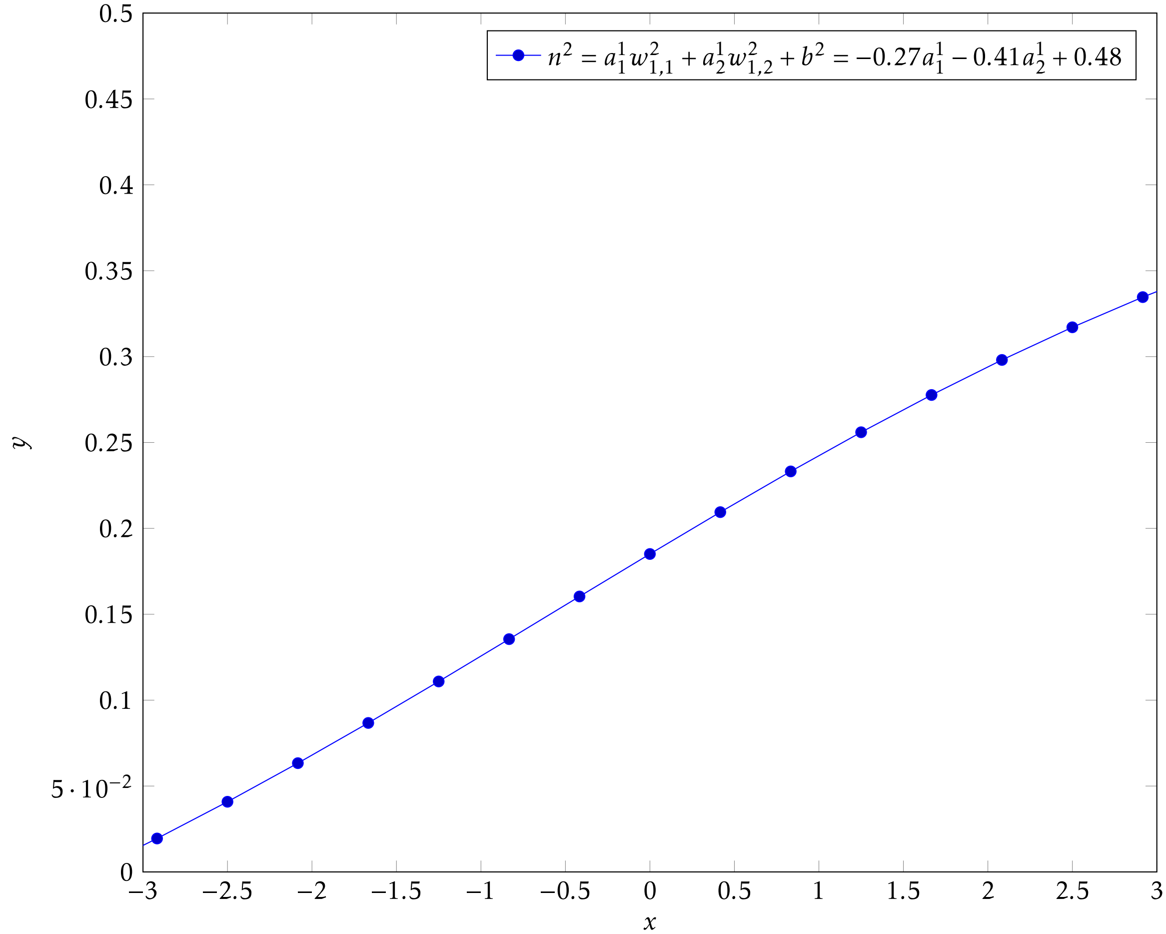


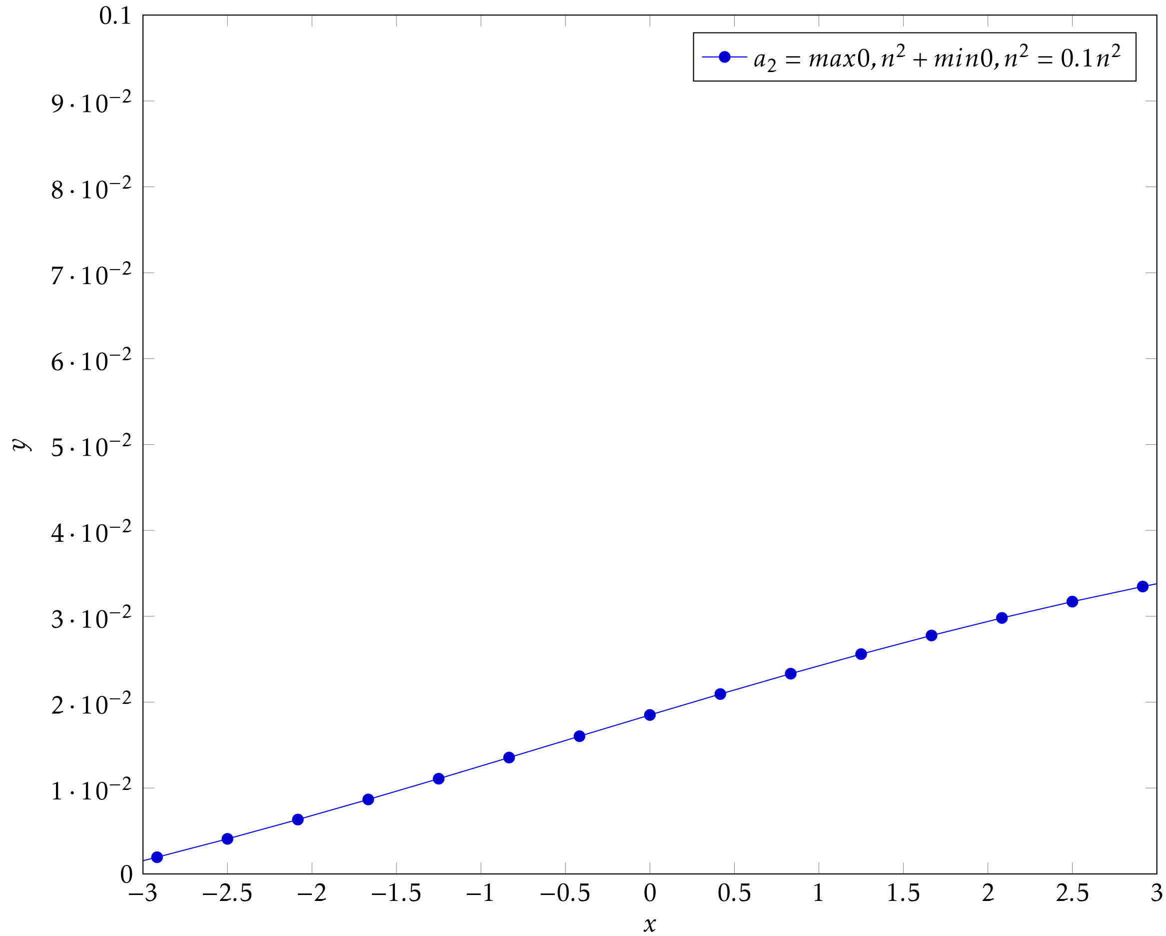
$a_1^1 = swish(n_1) = 1/(1 + e^{-\beta * n_1})$
y
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1


$n_2^1 = pw_{2,1}^1 + b_2^1 = -0.41 * p - 0.13$
1.4
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1
−1.2
−1.4
−3
−2.5
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
2.5
3



$a_2^1 = swish(n_2) = 1/(1 + e^{-\beta * n_2})$
y
x
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
−3
−2.5
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
2.5
3


$n^2 = a_1^1 w_{1,1}^2 + a_2^1 w_{1,2}^2 + b^2 = -0.27a_1^1 - 0.41a_2^1 + 0.48$
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
$5 \cdot 10^{-2}$
0
y
−3
−2.5
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
2.5
3
x



9 .

a)
p = [[1,0],[1,-1],[0,-1],[0,1],[-1,0], [1,1]]
t = [0,0,0,1,1,1]
w = [0,0]
b = 1

while True:
counter = 0
for i in range(8):
a = vectormul(w,p[i]) + b
Calculate error
e = t[i] - a

```
if e == 0:
counter += 1
w[0] = w[0] + e*p[i][0]
w[1] = w[1] + e*p[i][1]
b = b + e
if counter == 8:
break

    print (f"Final weight: w and bias: b")

    for i in range(8):
a = hardlim(vectormul(w,p[i]) + b)
print (f"Pi is a and is supposed to be t[i]")
```

Final weight: [-1, 2] and bias: -1

b)

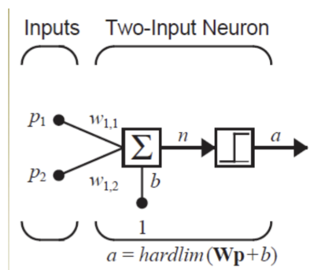
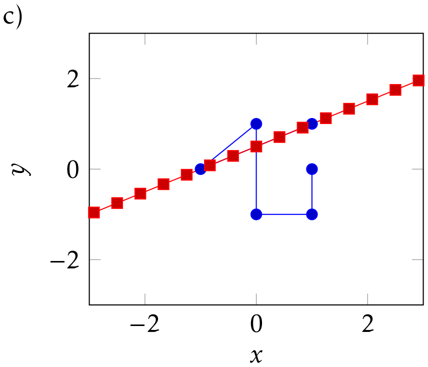


Figure 7: Diagram

c)



d)

By adding the vector[0 0] to Class I the network classifies it correctly

e)

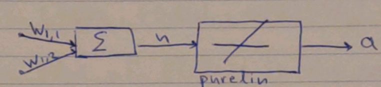
The vector is already classified correctly,therefore there is no need for any change

10 .

Problem 10.

A. $P_1 = \{0, 0\}$ $t_1 = -1$
$P_2 = \{0, 1\}$ $t_2 = -1$
$P_3 = \{1, 0\}$ $t_3 = 1$
$P_4 = \{2, 0\}$ $t_4 = 1$
$P_5 = \{-1, -1\}$ $t_5 = -1$
$P_6 = \{0, -2{,}5\}$ $t_6 = -1$
$P_7 = \{1{,}5, -1{,}5\}$ $t_7 = 1$

B.



C. The mean square error can be expressed as $F(x) = c - 2x^T h + x^T R x$, and c, h and R can be calculated by:

$$c = E[t^2] = 0{,}25(-1)^2 + 0{,}25(-1)^2 + 0{,}1(1)^2 + 0{,}1(1)^2 + 0{,}1(-1)^2 + 0{,}1(-1) + \\ + 0{,}1(1)^2 = 1$$

$$h = E[tz] = 0{,}25(-1)\begin{bmatrix}0\\0\end{bmatrix} + 0{,}25(-1)\begin{bmatrix}0\\1\end{bmatrix} + 0{,}1(1)\begin{bmatrix}1\\0\end{bmatrix} + 0{,}1(1)\begin{bmatrix}2\\0\end{bmatrix} + \\ + 0{,}1(-1)\begin{bmatrix}-1\\-1\end{bmatrix} + 0{,}1(-1)\begin{bmatrix}0\\-2{,}5\end{bmatrix} + 0{,}1(1)\begin{bmatrix}1{,}5\\-1{,}5\end{bmatrix} = \begin{bmatrix}0{,}55\\-0{,}05\end{bmatrix}$$

$$R = E[zz^T] = 0{,}25\begin{bmatrix}0\\0\end{bmatrix}\begin{bmatrix}0 & 0\end{bmatrix} + 0{,}25\begin{bmatrix}0\\1\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix} + 0{,}1\begin{bmatrix}1\\0\end{bmatrix}\begin{bmatrix}1 & 0\end{bmatrix} + \\ + 0{,}1\begin{bmatrix}2\\0\end{bmatrix}\begin{bmatrix}2 & 0\end{bmatrix} + 0{,}1\begin{bmatrix}-1\\-1\end{bmatrix}\begin{bmatrix}-1 & -1\end{bmatrix} + 0{,}1\begin{bmatrix}0\\-2{,}5\end{bmatrix}\begin{bmatrix}0 & -2{,}5\end{bmatrix} + 0{,}1\begin{bmatrix}1{,}5\\-1{,}5\end{bmatrix}\begin{bmatrix}1{,}5 & -1{,}5\end{bmatrix}$$

Figure 8:

$$R = \begin{bmatrix} 0,825 & -0,125 \\ -0,125 & 1,2 \end{bmatrix}$$

So the mean square error performance index is

$$F(x) = 1 - 2\begin{bmatrix} w_{1,1} & w_{1,2} \end{bmatrix}\begin{bmatrix} 0,55 \\ -0,05 \end{bmatrix} + \begin{bmatrix} w_{1,1} & w_{1,2} \end{bmatrix}\begin{bmatrix} 0,825 & -0,125 \\ -0,125 & 1,2 \end{bmatrix}\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix}$$

$$= 1 - 2(0,55w_{1,1} - 0,05w_{1,2}) + 0,825\,w_{1,1}^2 - 0,25\,w_{1,1}w_{1,2} + 1,2\,w_{1,2}^2$$

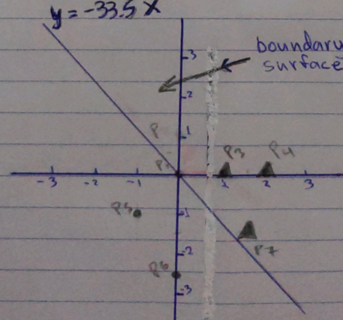
$$F(x) = 1 - 1,1w_{1,1} + 0,1w_{1,2} + 0,825\,w_{1,1}^2 - 0,25\,w_{1,1}w_{1,2} + 1,2\,w_{1,2}^2$$

* The plot of $F(x)$ is in the end of the exercise

D.

$$x^* = R^{-1}h = \begin{bmatrix} 1,231 & 0,128 \\ 0,128 & 0,846 \end{bmatrix}\begin{bmatrix} 0,55 \\ -0,05 \end{bmatrix} = \begin{bmatrix} 0,67 \\ -0,02 \end{bmatrix}$$

Thus, we have a minumum at $w_{1,1} = 0,67$
$w_{1,2} = 0,02$

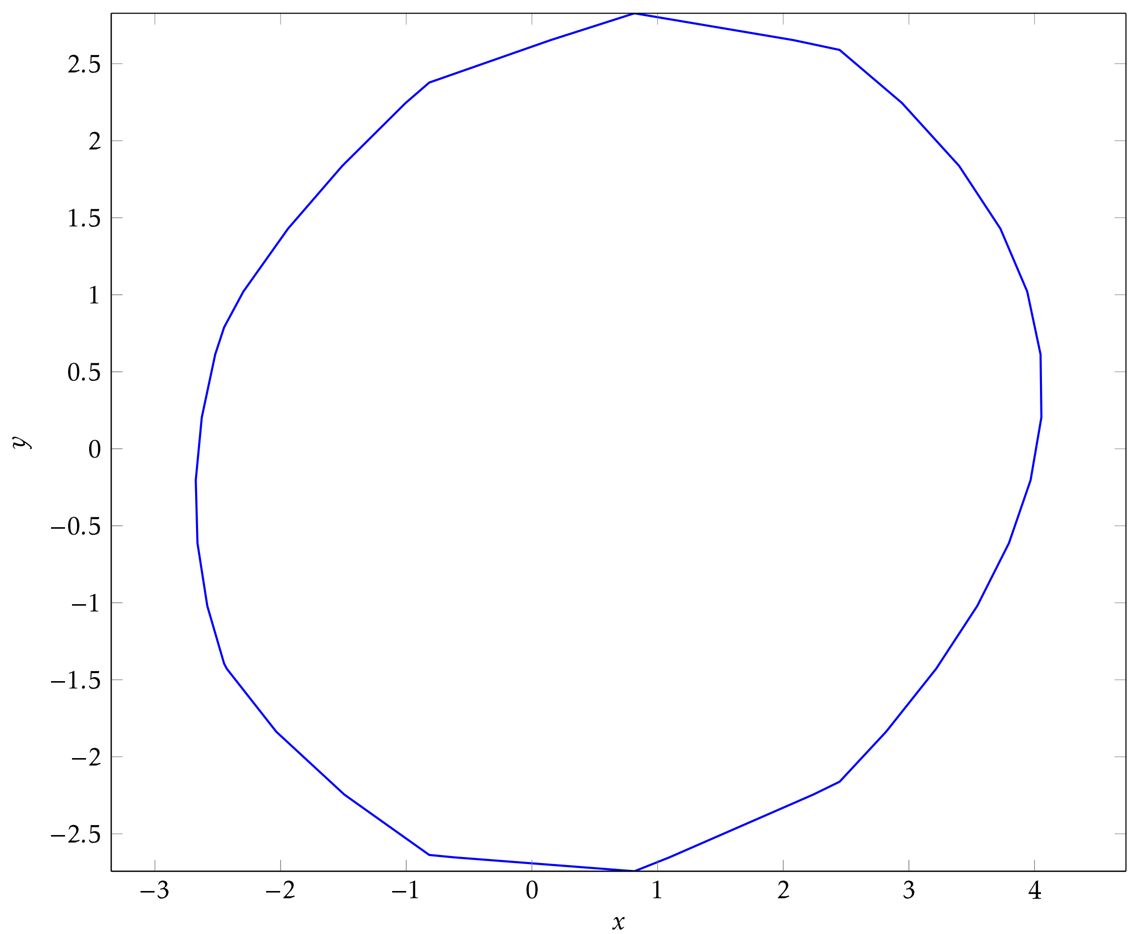


Misses one

E. The maximum stable learning rate will not change, because R will not change if we change the target values to opposite values.

Figure 9:

$1 - 1.1 * x + 0.1 * y + 0.825 * x * x - 0.25 * x * y + 1.2 * y * y$



11 .

The mean square error can be expressed as $F(X) = c - 2X^T h + X^T RX$, and c, h and R can be calculated by:

$$c = E[t^2] = 0.75(1)^2 + 0.25(-1)^2 = 1$$

$$h = E[tZ] = 0.75(1)\begin{bmatrix} 1 \\ 1 \end{bmatrix} + 0.25(-1)\begin{bmatrix} -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$R = E[ZZ^T] = 0.75\begin{bmatrix} 1 \\ 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \end{bmatrix} + 0.25\begin{bmatrix} -1 \\ -1 \end{bmatrix}\begin{bmatrix} -1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

so the mean square error performance index is

$$F(X) = 1 - 2\begin{bmatrix} w_{1,1} & w_{1,2} \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} w_{1,1} & w_{1,2} \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix}$$

$$= 1 - 2(w_{1,1} + w_{1,2}) + (w_{1,1} + w_{1,2})^2$$

The eigen values and eigen vectors of Hessian matrix of F(x) is
A=2*[1 1;1 1];
[V,D] = eig (A)
V =
0.7071 0.7071
-0.7071 0.7071
D =
0 0
0 4.0000
Since all the eigen values are non-negative, and one is zero, there's a weak minimum in this problem. The surface and contour plot of F(x) are shown in Fig.1 and Fig.2.

1. The maximum stable learning rate will satisfy $\alpha < \frac{2}{\lambda_{\max}} = \frac{2}{4} = 0.5$.

1. Take $\alpha = 0.2$. After 40 iterations we got $w_{1,1} = 0.5, w_{1,2} = 0.5$ (W_1 in the code), and the trajectory is shown in Fig.2 as dotted symols.

```
clear
[X,Y] = meshgrid(-3 : .1 : 3);
F = 1 - 2 * (X + Y) + (X + Y).∧2; surf(X,Y,F) figure; contour(X,Y,F) hold on;
%Initialize data
P = [1 -1;1 -1];
T = [1 -1];
alfa = 0.2;
W1 = [0;0];
```

```
W2 = [1;1];
for k = 1 : 2
    if (k == 1)
        W = W1;
    else
        W = W2;
    end
plot(W(1), W(??),'r*')
text(-0.3,-0.3,'W_0 =(0,0)');
text(1,1.2,'W_0 =(1,1)');
for step = 1 : 20
    for i = 1 : 2
        a = purelin(W' * P(:,i));
        e = T(i) - a;
        W = W + 2 * alfa * e * P(:,i);
        if (k == 1)
            plot(W(1), W(??),'k.')
            W1 = W;
        else
            plot(W(1), W(??),'b+')
            W2 = W;
        end
        end
    end
end
W1
W2
hold off;
W1 =
    0.5000
    0.5000
W2 =
    0.5000
    0.5000
```

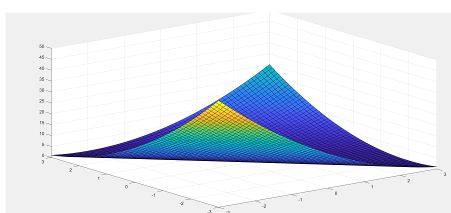


Figure 10: Mean square error

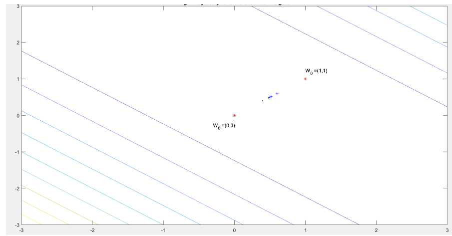


Figure 11: Trajectory

iv. For initial weights [1 1], after 40 iterations, we got (W2 in the code), and the trajectory is also shown in Fig.2 as ”+” symols. And the decision boundary is given in Fig.3.

```
P = [1 -1;1 -1]; W = [0.5 0.5];
figure;
plot(P(1,1),P(2,1),'r+');
hold on;
plot(P(1,2),P(2,2),'r+');

x = -2 : .1 : 2;
y =(-W(1)*x )/W(2);
plot(x,y);
axis([-2 2 -2 2]);

hold off;
```

img3

e)

From the figure above, we can see that LMS algorithm, unlike the perceptron learning rule, places the decision boundary as far from the patterns as possible.

Also, we see that for 2 different initial weights, the solution converges to the same value. However, if other arbitrary initial points are selected, we could get different solutions, since the performance index only has a weak minimum, i.e. the minimum is not unique. There’s a line on which all the mean square error is minimum (zero).

12 .

13 .

p	q	$\neg p$	$\neg p \wedge q$	p $\vee(\neg p \wedge q)$	$\neg(p \vee (p \wedge q))$	$\neg(p \vee (p \wedge q)) \wedge q$
0	0	1	1	0	1	1
0	0.5	1	1	0	1	1
0	1	1	1	0	1	1
0.5	0	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	0.5	1	0.5	0.5	1
1	0	0	0	0	1	1
1	0.5	0	0.5	0.5	0.5	0.5
1	1	0	1	1	0	1

heightp	q	$\neg p$	$\neg p \wedge q$	p $\vee(\neg p \wedge q)$	$\neg(p \vee (p \wedge q))$	$\neg(p \vee (p \wedge q)) \wedge q$
0	0	1	1	0	1	1
0	1	1	1	0	1	1
1	0	0	0	0	1	1
1	1	0	1	1	0	1

If we use crisp logic over fuzzy the "(P(x) and (P(x)$-$ > Q(x))) $-$ > Q(x)" becomes tautology.

14 .

Dimitris and Fany go to park if it is a beautiful day and it is not too hot, or if it isn't raining.
With Assuming that it is a beautiful day with 0.7 degree,it is hot with 0.3 degree and it is raining with 0.6 degree . We have:
$(0.7degree \wedge \neg 0.3degree) \vee \neg 0.6degree$
$(0.7degree \wedge 0.7degree) \vee 0.4degree$
$0.7degree \vee 0.4degree$
$0.7degree$
So Dimitris and Fany will go to park with 0.7 degree