

AND KAPISI

0 0 ----------- 0

0 1 ----------- 0

1 0 ----------- 0

1 1 ----------- 1

AGIRLIK DEGERLERI

* Girdi katmanı ile ara katman arasındaki başlangıç ağırlıkları = Ai

Ai = [ [0.131 0.443 0.624]

[-0.347 -0.445 -0.127] ]

Ai =  [[ A11i A12i A13i  ]

[ A21i A22i A23i  ]]

GK\_A1 = [0.131 0.443 0.624; -0.347 -0.445 -0.127 ]

A1\_A2 = [0.978 0.956; 0.724 -0.932; -0.946 0.923 ]

A2\_C = [0.987 0.966]

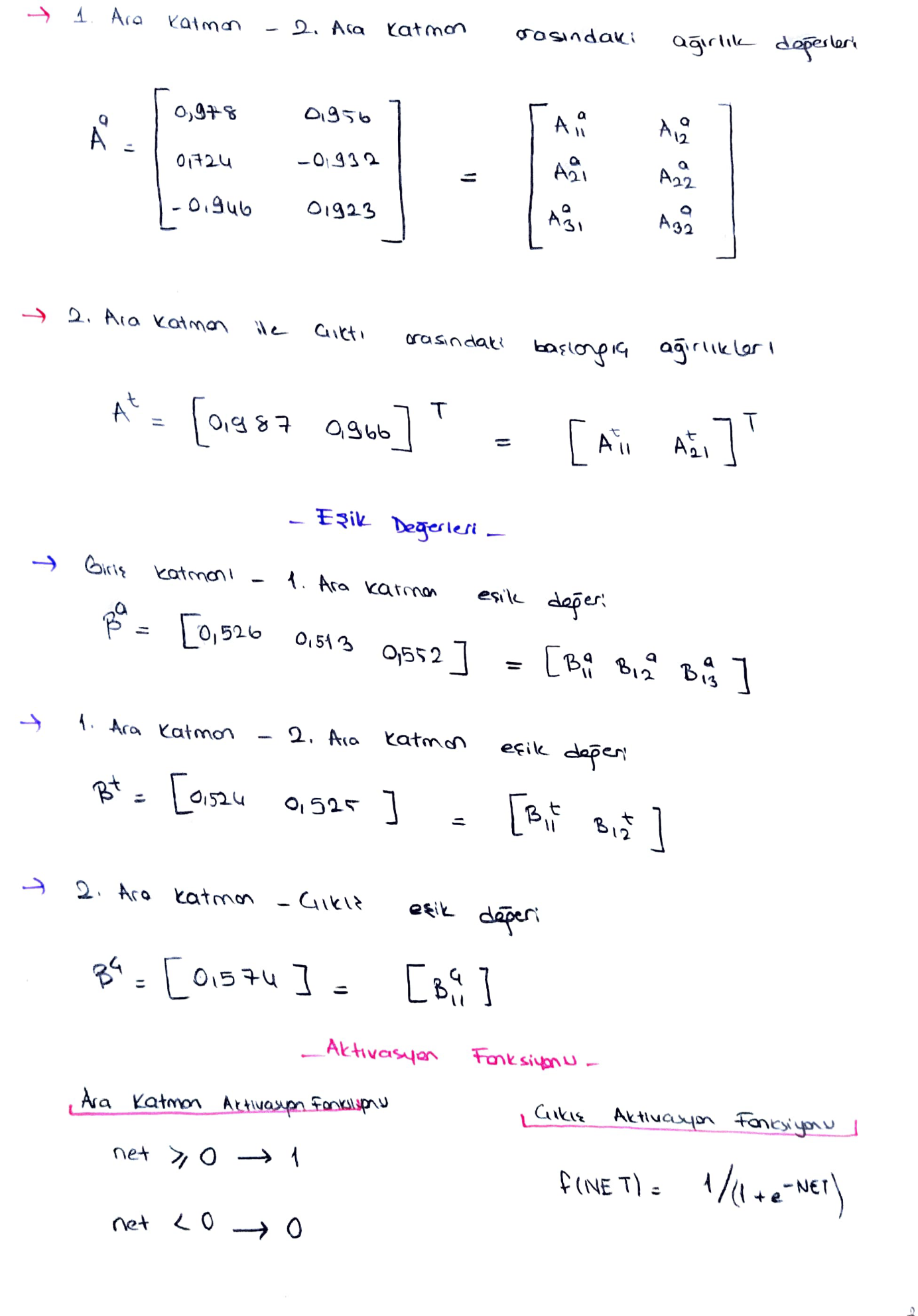
B1 = [0.526 0.513 0.552]

B2 = [0.524 0.525

B3 = [0.574]

Fi = 0.9

Mem = 0.1



* 1. Ara katman ile 2. Katman arasındaki ağırlık değerleri:

Aa = [ [0.978 0.956]

[0.724 -0.932]

[-0.946 0.923] ]

Aa = [ [A11a A12a]

[A21a A22a]

[A31a A32a] ]

* 2.Ara katman ile çıktı arasindaki başlangıç agırlıkları

At = [0.987 0.966]T = [A11t A12t]T

EŞİK DEGERLERİ

* Giriş katmanı ile 1. katman arasındaki eşik değeri:

Ba = [0.526 0.513 0.552]= [B11a B12a  B13a]

* 1. Katman ile 2. katman arasındaki eşik değeri

,

Bt = [0.524 0.525]= [B11t B12t]

* 2. Katman ile çıkış katmanı arasındaki eşik değeri

Bç = [0.574]= [B11ç]

AKTİVASYON FONSKSİYONU

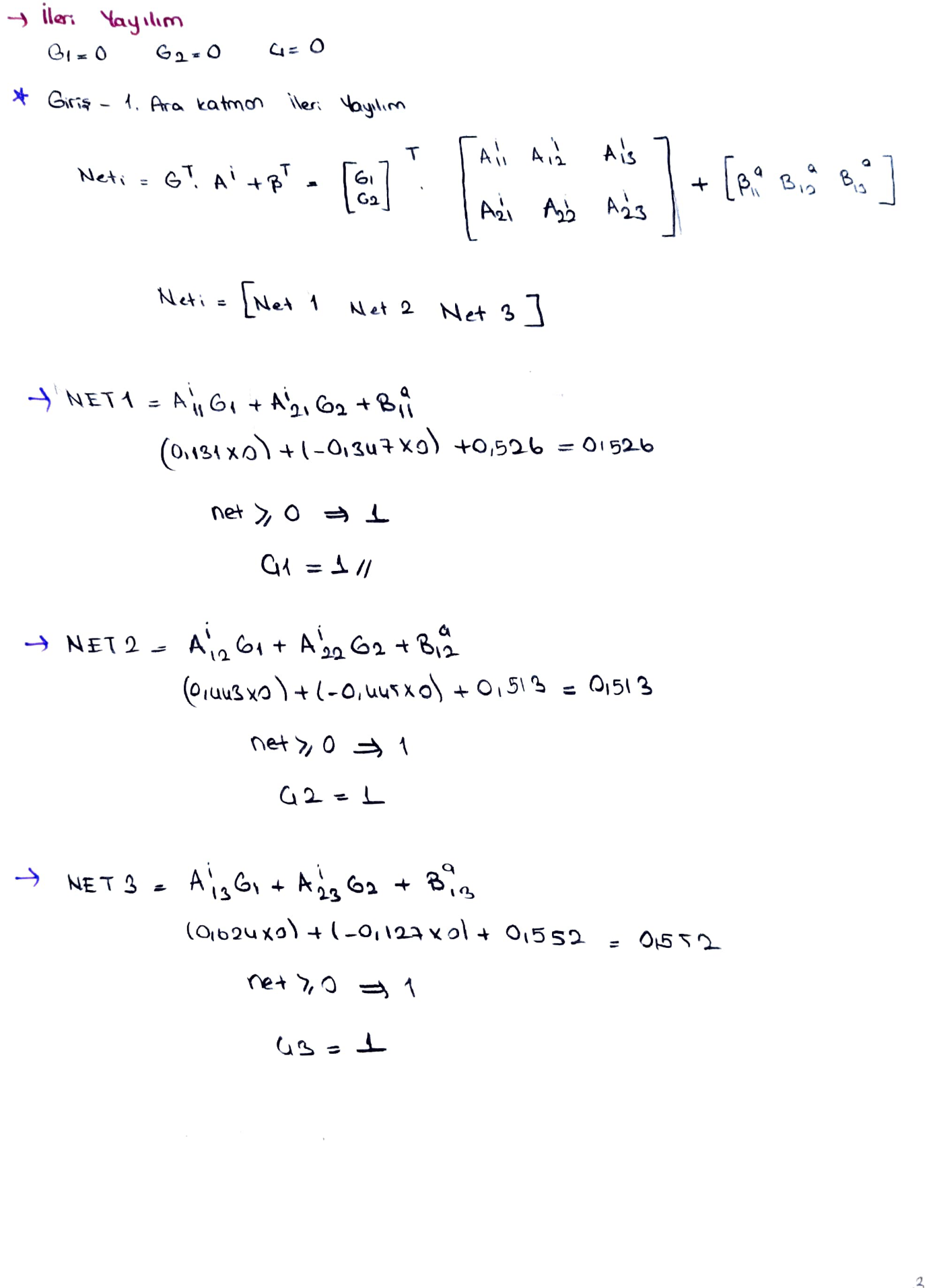
* Ara katman aktivasyon fonksiyonu

Net >= 0 ise 1

Net < 0 ise 0

* Çıkış aktivasyon fonksiyonu

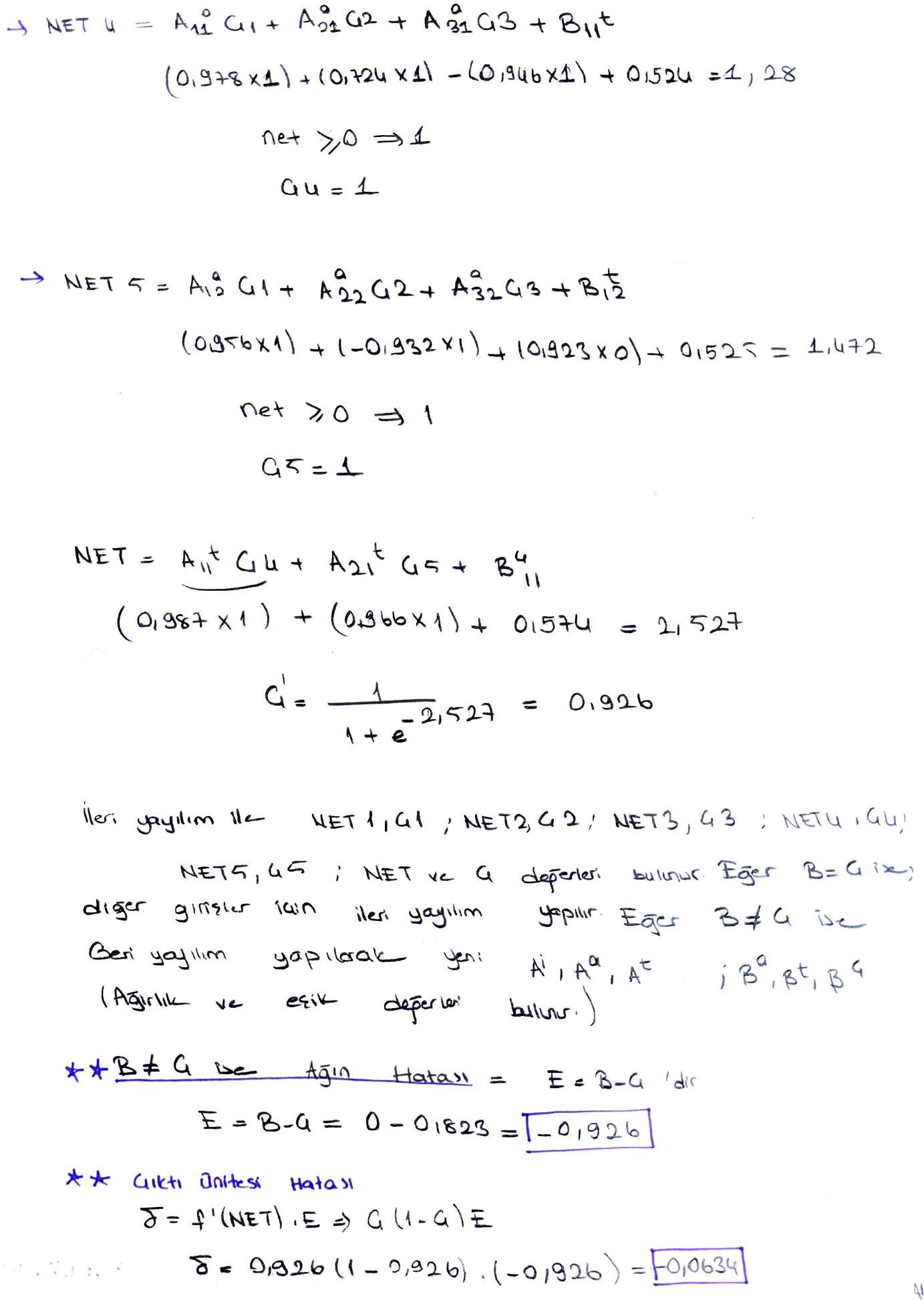
F(NET) = 1 / (1 + e--NET)

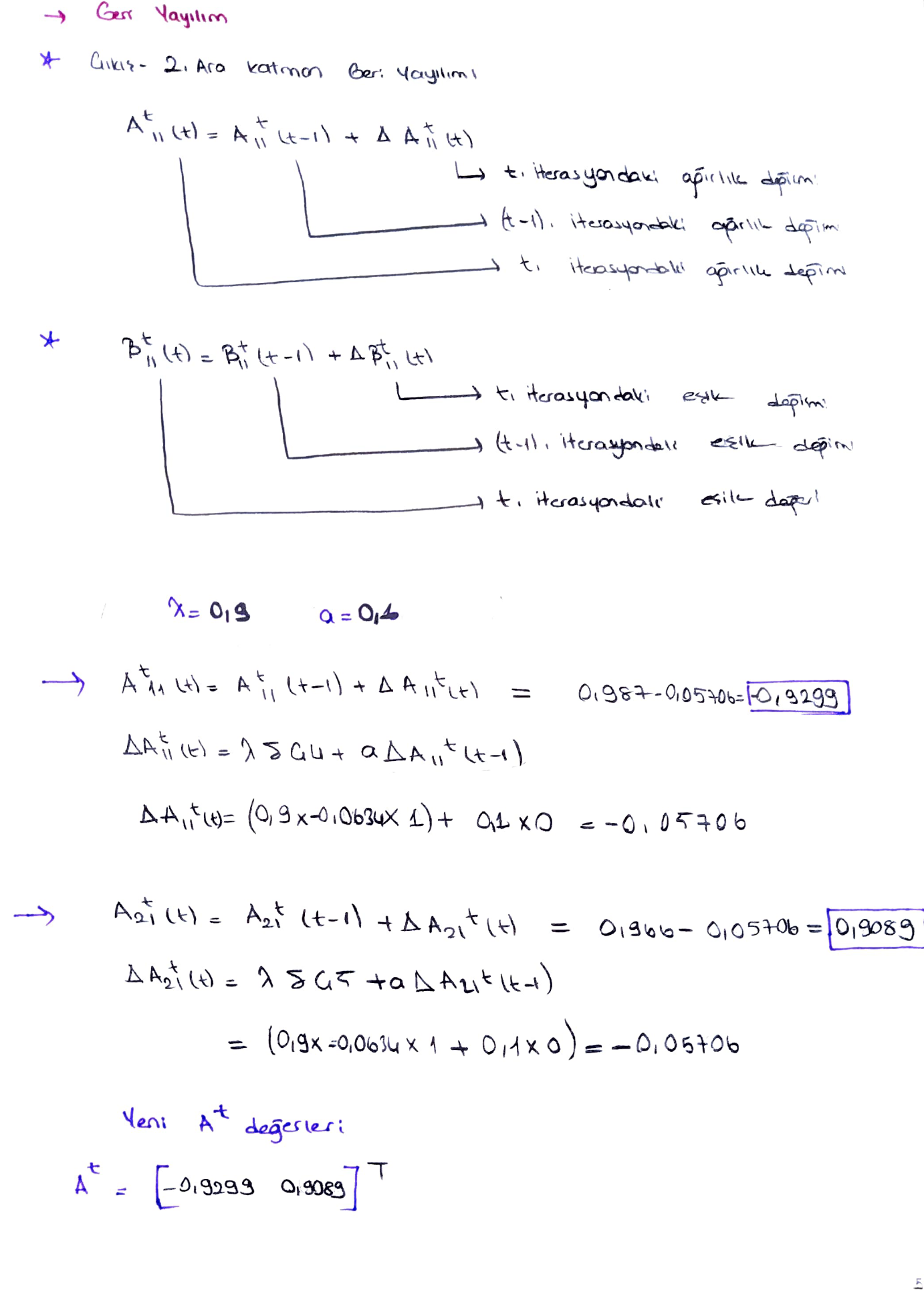


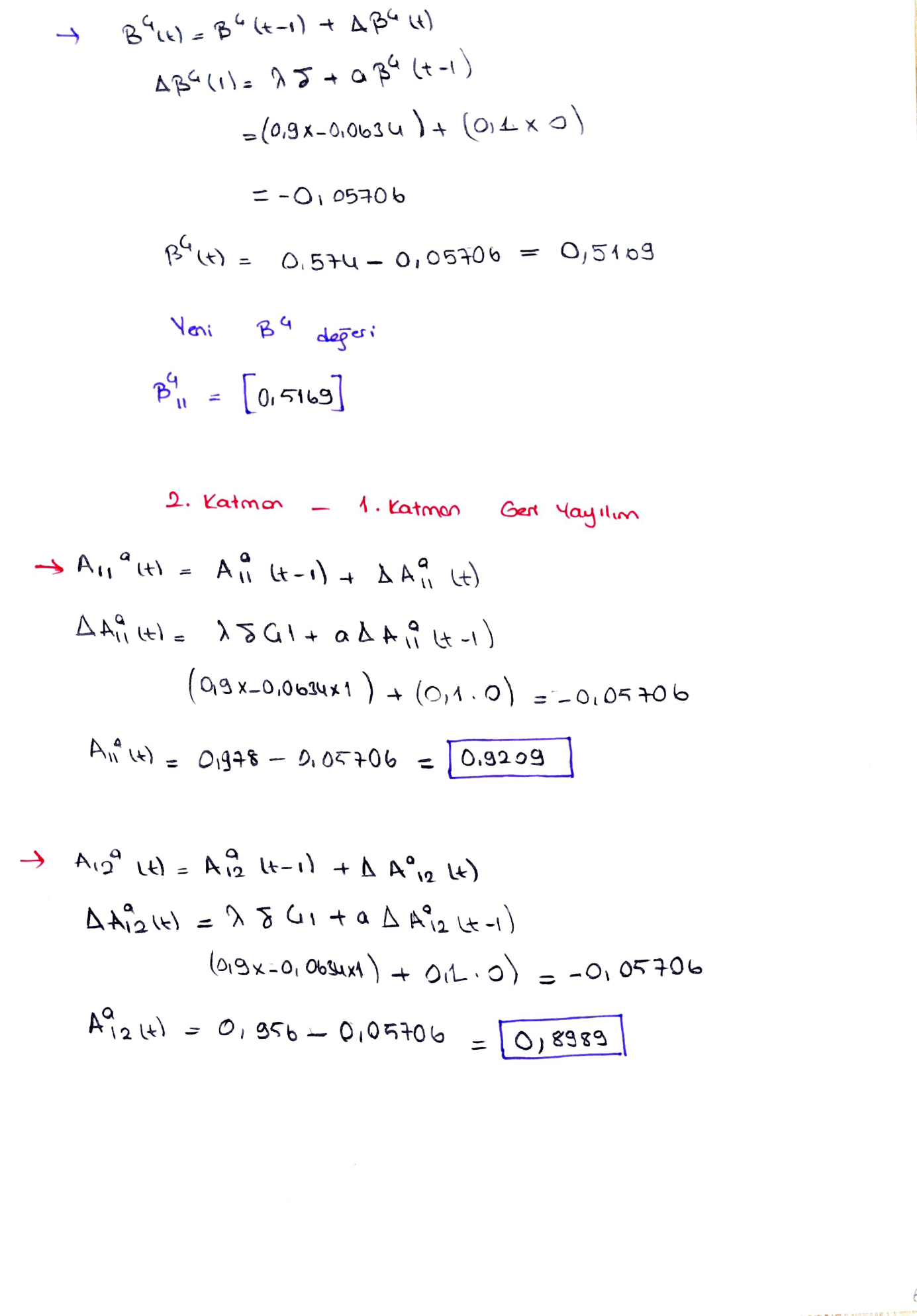
İLERİ YAYILIM

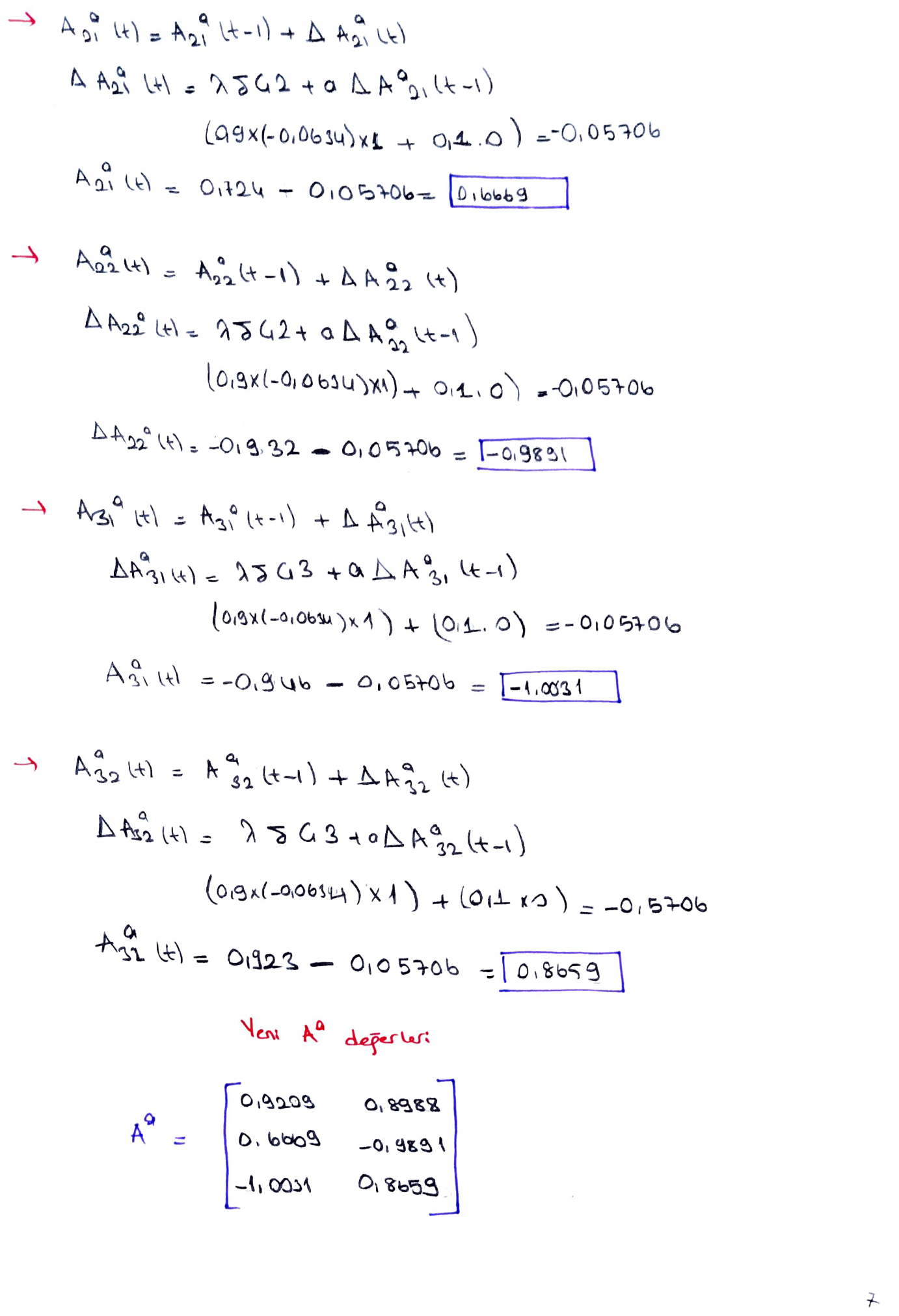
G1 = 0, G2 = 0, Ç = 0

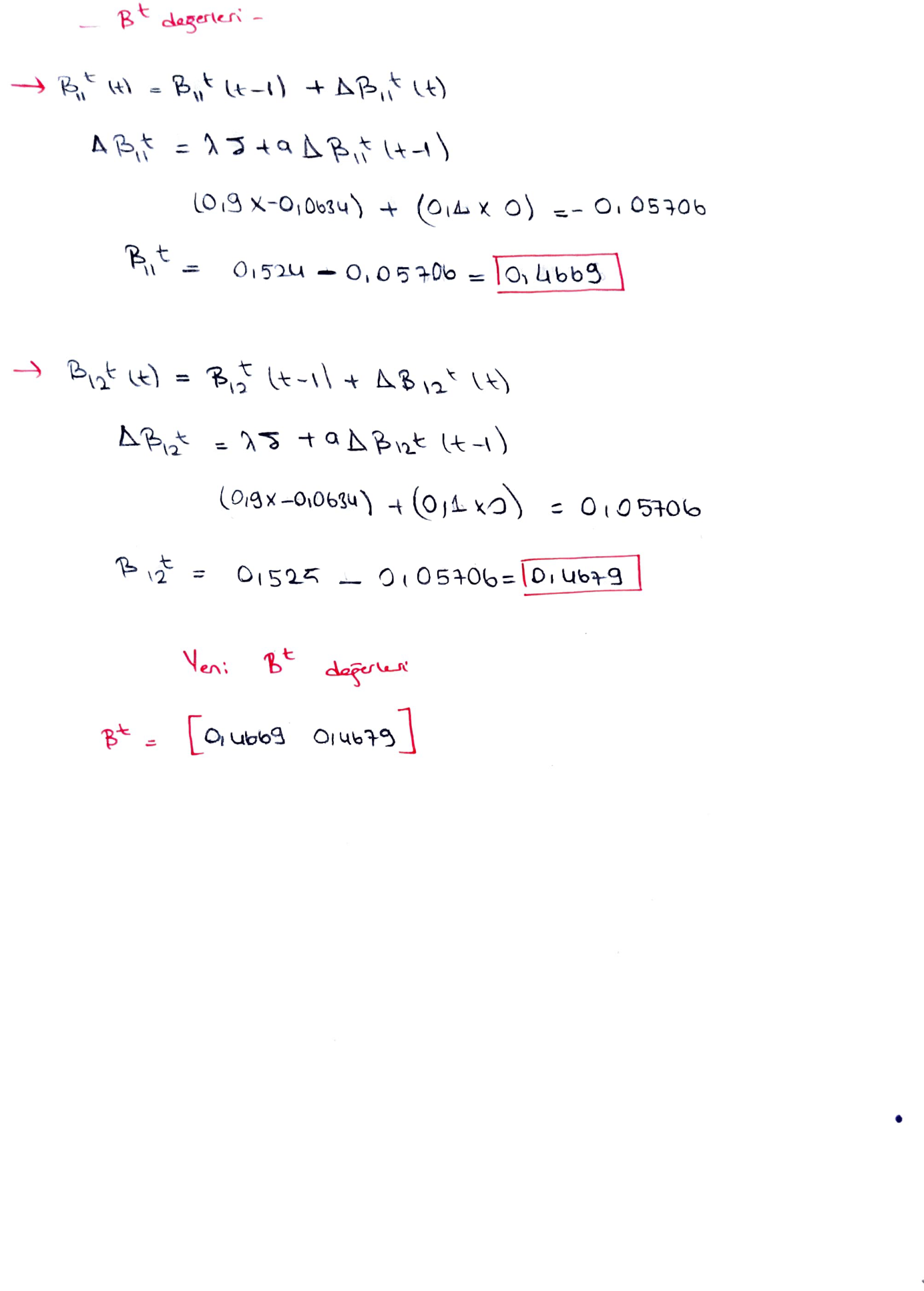
* NETi = GT \* Ai + Bt + T \* T +











Matlab Kod

**GD = [ 0 0; 0 1; 1 0; 1 1]; %durumların tanitilmasi**

**GK\_A1 = [0.131 0.443 0.624; -0.347 -0.445 -0.127 ]; %Giriş katmanı ile ara katman arasındaki ağırlıklar**

**A1\_A2 = [0.978 0.956; 0.724 -0.932; -0.946 0.923 ]; %Birinci ara katman ile ikinci ara katman arasındaki ağırlıklar**

**A2\_C = [0.987 0.966] ; %İkinci ara katman ile çıkış katmanı arasındaki ağırlıklar**

**B1 = [0.526 0.513 0.552]; % I. esik degeri**

**B2 = [0.524 0.525]; % II. esik degeri**

**B3 = [0.574]; % III. esik degeri**

**bk = [0 0 0 1]; %beklenen durumlar**

**A2\_C\_O = [0 0] ; %A2\_C(t - 1)**

**B3\_C\_O = [0]; %B3\_C(t - 1)**

**A1\_A2\_O = [0 0;0 0;0 0] ; %A1\_A2(t - 1)**

**B2\_C\_O = [0 0]; %B2\_C(t - 1)**

**GK\_A1\_O = [0 0 0; 0 0 0] ; %A1\_C(t - 1)**

**B1\_C\_O = [0 0 0]; %B1\_C(t - 1)**

**fi = 0.9 ; %ogrenme katsayisi**

**mem = 0.1; %momentum**

**a = 1;**

**iterasyon = 0;**

**for sayici = 0 : 2**

**while a < 5**

**iterasyon = iterasyon + 1;**

**%ileri yayilim**

**net\_1 = GD(a, 1 : 2) \* GK\_A1 +B1**

**net\_1 = step\_fonk(net\_1)**

**net\_2 = (net\_1 \* A1\_A2) + B2**

**net\_2 = step\_fonk(net\_2)**

**net\_3 = (net\_2 \* (A2\_C)') + B3**

**net\_3 = sgm\_fonk(net\_3)**

**%hata kontrolu**

**E = bk(a) - net\_3**

**if(abs(E) < abs(0.01))**

**fprintf('hata yok \n');**

**a = a + 1;**

**else if(abs(E) >= abs(0.01) )**

**z = sgm\_trv\_fonk(net\_3);**

**sigma = z \* E**

**%geri yayilim ara\_katman\_2 - cikis**

**A2\_C\_Y = geri\_yayilim\_c(A2\_C,sigma,net\_2,A2\_C\_O)**

**A2\_C\_O = A2\_C;**

**A2\_C = A2\_C\_Y;**

**B3\_Y = geri\_yayilim\_bc(B3 ,sigma , B3\_C\_O)**

**B3\_C\_O = B3;**

**B3 = B3\_Y;**

**% ara\_katman\_1 - ara\_katman\_2**

**A1\_A2\_Y = geri\_yayilim\_c(A1\_A2, sigma, (net\_1)', A1\_A2\_O)**

**A1\_A2\_O = A1\_A2;**

**A1\_A2 = A1\_A2\_Y;**

**B2\_Y = geri\_yayilim\_bc(B2 ,sigma , B2\_C\_O)**

**B2\_C\_O = B2;**

**B2 = B2\_Y;**

**% ara\_katman\_1 - ara\_katman\_2**

**GK\_A1\_Y = geri\_yayilim\_c(GK\_A1, sigma, (GD(a, 1 : 2))',GK\_A1\_O)**

**GK\_A1\_O = GK\_A1;**

**GK\_A1 = GK\_A1\_Y;**

**B1\_Y = geri\_yayilim\_bc(B1 ,sigma , B1\_C\_O)**

**B1\_C\_O = B1;**

**B1 = B1\_Y;**

**a = 1;**

**end**

**end**

**end**

**a = 4;**

**GD(a, 1 : 2)**

**iterasyon**

**break**

**end**

Matlab Fonksiyonlar

**%\***

**%"Bt\_0 ------- Esik degerlerinin (t - 1)**

**%"Bt\_1 ------- Esik degerlerinin (t)**

**% "Bt\_2 ------- Esik degerlerinin (t + 1)**

**%\***

**function [y] = geri\_yayilim\_bc(Bt\_1 ,sigma , Bt\_0)**

**fi = 0.9 ; %ogrenme katsayisi**

**mem = 0.1; %momentum**

**Bt\_2 = Bt\_1 + (fi \* sigma + mem \* Bt\_0);**

**y = Bt\_2 ;**

**end**

**%\***

**%At\_0 ------- Agirlik degerlerinin (t - 1)**

**%At\_1 ------- Agirlik degerlerinin (t)**

**%At\_2 ------- Agirlik degerlerinin (t + 1)**

**%\***

**function [y] = geri\_yayilim\_c(At\_1,sigma,net,At\_0)**

**fi = 0.9 ; %ogrenme katsayisi**

**mem = 0.1; %momentum**

**At\_2 = At\_1 + (fi .\* sigma .\* net + mem .\* At\_0);**

**y = At\_2;**

**end**

**%\***

**%matris gelen matris degeri"**

**%\***

**function [y] = sgm\_fonk(matris)**

**[m,n] = size(matris);**

**%cikan degerler aktivasyon fonksiyonuna girer**

**for i = 1 : m**

**for j = 1 : n**

**matris(i,j) = 1 / (1 + exp (-matris(i,j)));**

**j = j + 1;**

**end**

**i = + 1;**

**end**

**y = matris;**

**end**

**%\***

**%matris gelen matris degeri"**

**%\***

**function [y] = sgm\_trv\_fonk(matris)**

**[m,n] = size(matris);**

**%cikan degerler aktivasyon fonksiyonuna girer**

**for i = 1 : m**

**for j = 1 : n**

**matris(i,j) = matris(i,j) \* (1 - matris(i,j));**

**j = j + 1;**

**end**

**i = + 1;**

**end**

**y = matris;**

**end**

**%\***

**%matris gelen matris degeri"**

**%\***

**function [y] = step\_fonk(matris)**

**[m,n] = size(matris);**

**%cikan degerler aktivasyon fonksiyonuna girer**

**for i = 1 : m**

**for j = 1 : n**

**if(matris(i,j) >= 0)**

**matris(i,j) = 1;**

**else**

**matris(i ,j) = 0;**

**end**

**j = j + 1;**

**end**

**i = + 1;**

**end**

**y = matris;**

**end**

Python Kod

"""GiriÅŸ katmanÄ± ile ara katman arasÄ±ndaki aÄŸÄ±rlÄ±klar

Birinci ara katman ile ikinci ara katman arasÄ±ndaki aÄŸÄ±rlÄ±klar

Ä°kinci ara katman ile Ã§Ä±kÄ±ÅŸ katmanÄ± arasÄ±ndaki aÄŸÄ±rlÄ±klar

I. esik degeri

II. esik degeri

III. esik degeri

beklenen durumlar

A2\_C(t - 1)

B3\_C(t - 1)

A1\_A2(t - 1)

B2\_C(t - 1)

A1\_C(t - 1)

B1\_C(t - 1)

ogrenme katsayisi"""

import numpy as np

import math as mt

"fonksiyon tanimlamalari"

"\*"

"matris gelen matris degeri"

"\*"

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

def step\_fonk(matris):

(m,n) = matris.shape;

for i in range(m):

for j in range(n):

if(matris[i,j] >= 0):

matris[i,j] = 1;

else:

matris[i,j] = 0;

j = j + 1;

i = i +1;

return matris;

"""\*

"matris gelen matris degeri"

"\*

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

def sgm\_fonk(matris):

m = len(matris);

for i in range(m):

matris[i] = 1 / (1 + mt.exp(-matris[i]));

i = i + 1;

return matris;

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

def sgm\_trv\_fonk(matris):

m = len(matris);

for i in range(m):

matris[i] = matris[i] \* (1 - matris[i]);

i = i + 1;

return matris;

"""\*

"At\_0 ------- Agirlik degerlerinin (t - 1)

"At\_1 ------- Agirlik degerlerinin (t)

"At\_2 ------- Agirlik degerlerinin (t + 1)

"\*

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

def geri\_yayilim\_c(At\_1,sigma,net,At\_0):

fi = 0.9;

mem = 0.1;

At\_2 = At\_1 + (fi \* sigma \* net + mem \* At\_0);

y = At\_2;

return y;

"""\*

"Bt\_0 ------- Esik degerlerinin (t - 1)

"Bt\_1 ------- Esik degerlerinin (t)

"Bt\_2 ------- Esik degerlerinin (t + 1)

"\*

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"""

def geri\_yayilim\_bc(Bt\_1 ,sigma , Bt\_0):

fi = 0.9;

mem = 0.1;

Bt\_2 = Bt\_1 + (fi \* sigma + mem \* Bt\_0);

y = Bt\_2;

return y;

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

"degisken tanimlamalari"

GD = np.array([[0.0, 0.0],

[0.0, 1.0],

[1.0 ,0.0],

[1.0 ,1.0]]);

GK\_A1 = np.array([[0.131 ,0.443, 0.624],

[-0.347 ,-0.445 ,-0.127]]);

A1\_A2 = np.array([[0.978, 0.956],

[0.724, -0.932],

[-0.946, 0.923 ]]);

A2\_C = np.array([0.987 ,0.966]);

B1 = np.array([0.526 ,0.513 ,0.552]);

B2 = np.array([0.524 ,0.525]);

B3 = np.array([0.574]);

bk = np.array([0.0 ,0.0, 0.0 ,1.0]);

A2\_C\_O = np.array([0 ,0]) ;

B3\_C\_O = np.array([0.0]);

A1\_A2\_O = np.array([[0,0],[0,0],[0,0]] );

B2\_C\_O = np.array([0 ,0]);

GK\_A1\_O = np.array([[0.0, 0.0, 0.0], [0.0 ,0.0 ,0.0]]) ;

B1\_C\_O = np.array([0.0, 0.0, 0.0]);

fi = 0.9 ;

mem = 0.1;

E = 0.0;

iterasyon = 0;

a = 0;

x = 0;

for x in range(1):

while a < 4 :

"ileri yayilim"

print("iterasyon", iterasyon);

iterasyon = iterasyon + 1;

net\_1 = np.dot(GD[a : a + 1, 0:2], GK\_A1[0 : 3,0: 4 ]) + B1;

print("net\_1", net\_1);

net\_1 = step\_fonk(net\_1);

print("net\_af", net\_1);

net\_2 = np.dot(net\_1, A1\_A2) + B2;

print("net\_2", net\_2);

net\_2 = step\_fonk(net\_2);

print("net\_2\_af", net\_2);

net\_3 = np.dot(net\_2, A2\_C.transpose()) + B3;

print("net\_3", net\_3);

net\_3 = sgm\_fonk(net\_3);

print("net\_3\_af", net\_3);

E = bk[a] - net\_3;

if(abs(E) < abs(0.01)):

print("hata yok \n");

a = a + 1;

elif(abs(E) >= abs(0.01)):

z = sgm\_trv\_fonk(net\_3);

sigma = z \* E;

A2\_C\_Y = geri\_yayilim\_c(A2\_C,sigma,net\_2,A2\_C\_O);

A2\_C\_O = A2\_C;

A2\_C = A2\_C\_Y;

print("a2\_c\_y \n");

print(A2\_C\_Y , A2\_C\_O ,A2\_C ,"\n");

B3\_Y = geri\_yayilim\_bc(B3 ,sigma , B3\_C\_O);

B3\_C\_O = B3;

B3 = B3\_Y;

print("b3\_y \n");

print(B3\_Y,B3\_C\_O, B3 ,"\n");

A1\_A2\_Y = geri\_yayilim\_c(A1\_A2,sigma, net\_1.transpose(),A1\_A2\_O );

A1\_A2\_O = A1\_A2;

A1\_A2 = A1\_A2\_Y;

print("GK\_A1\_Y \n");

print(A1\_A2\_Y ,"\n" , A1\_A2\_O ,"\n" ,A1\_A2 ,"\n" );

B2\_Y = geri\_yayilim\_bc(B2 ,sigma , B2\_C\_O);

B2\_C\_O = B2;

B2 = B2\_Y;

print("b2y \n");

print(B2\_Y, "\n" ,B2\_C\_O, "\n", B2,"\n");

GK\_A1\_Y = geri\_yayilim\_c(GK\_A1,sigma,GD[a : a + 1, ::].transpose(),GK\_A1\_O);

GK\_A1\_O = GK\_A1;

GK\_A1 = GK\_A1\_Y;

print("GK\_A1\_Y \n");

print(GK\_A1\_Y ,"\n" , GK\_A1\_O ,"\n" ,GK\_A1 ,"\n" );

B1\_Y = geri\_yayilim\_bc(B1 ,sigma , B1\_C\_O);

B1\_C\_O = B1;

B1 = B1\_Y;

print("b1y \n");

print(B1\_Y, "\n" ,B1\_C\_O, "\n", B1 ,"\n");

a = 0;

print("iterasyon =" , iterasyon);

Feyzanur Boztepe 171110015

Can Ahmet Acar 171110001

Tayfun Bacak 171110009

Ömer Faruk Çelik 161110020