

Lab 6

Systemy CAD/CAE

Adrian Madej 25.11.2024

1. Zmodyfikowany fragment kodu

```
1. %ustawiamy seed
2. rng(0)
3. .
4. .
5. .
6. for j=1:ndataset
7.     %i=floor(r(j));
8.     %if(i==0)
9.         %i=1;
10.    %end

11.    % chcemy trenować bez losowości odkomentujemy 4 powyższe linie
12.    i=j;

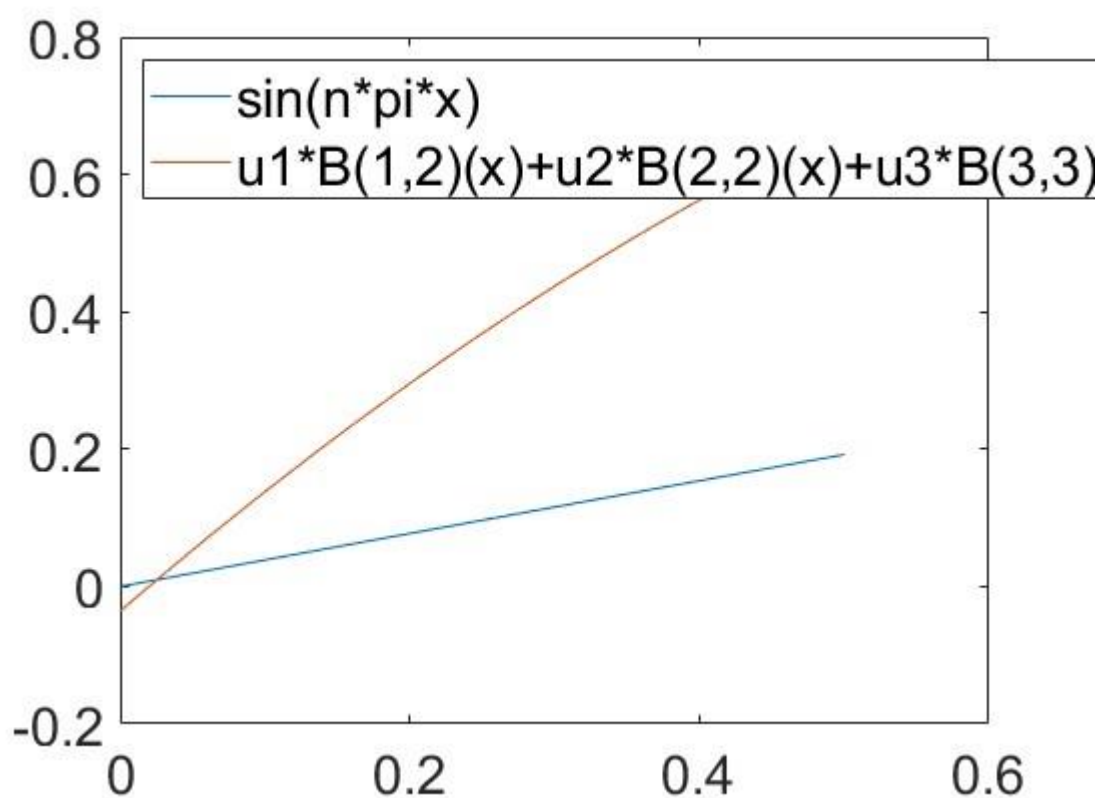
13.    for epoch=1:1000
14.        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15.        %1
16.        a1_=a1; b1_=b1; c1_=c1; d1_=d1;
17.    .
18.    .
19.    .
20.        d3=d3-eta3* derrordd;
21.    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
22.    end
23.    .
24.    .
25.    .
26. fprintf('a1= %f;\n', a1)
27. fprintf("b1= %f;\n", b1)
28. fprintf("c1= %f;\n", c1)
29. fprintf("d1= %f;\n", d1)
30. disp("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%")
31. fprintf("a2= %f;\n", a2)
32. fprintf("b2= %f;\n", b2)
33. fprintf("c2= %f;\n", c2)
34. fprintf("d2= %f;\n", d2)
35. disp("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%")
36. fprintf("a3= %f;\n", a3)
37. fprintf("b3= %f;\n", b3)
38. fprintf("c3= %f;\n", c3)
39. fprintf("d3= %f;\n", d3)
```

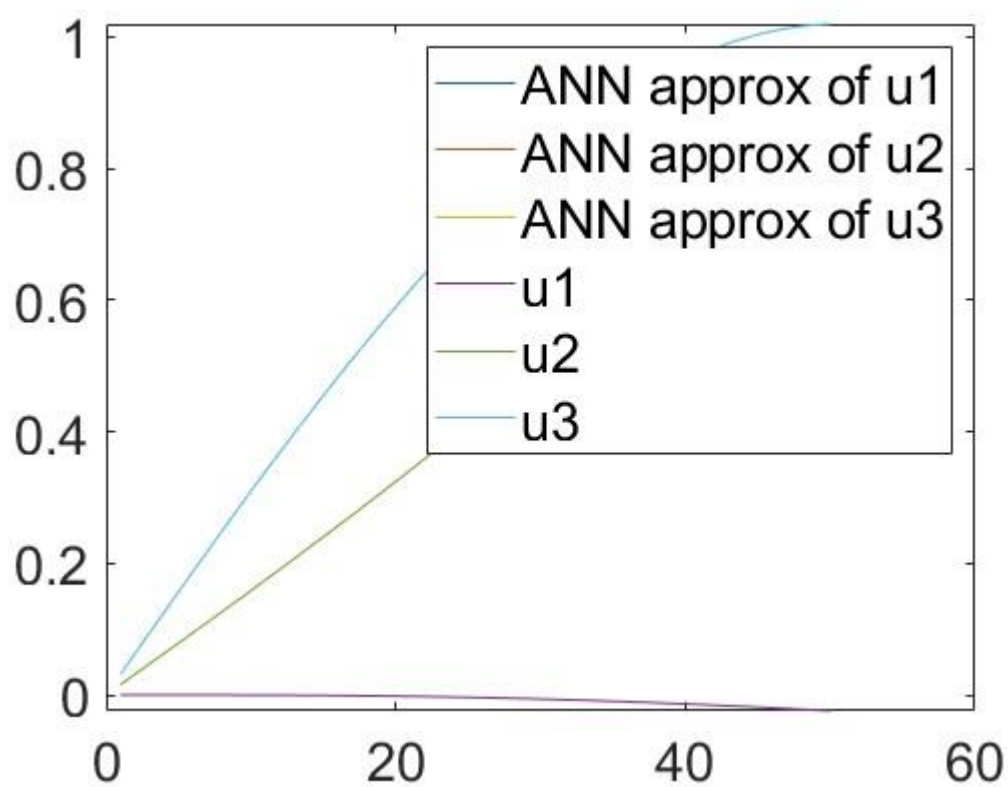
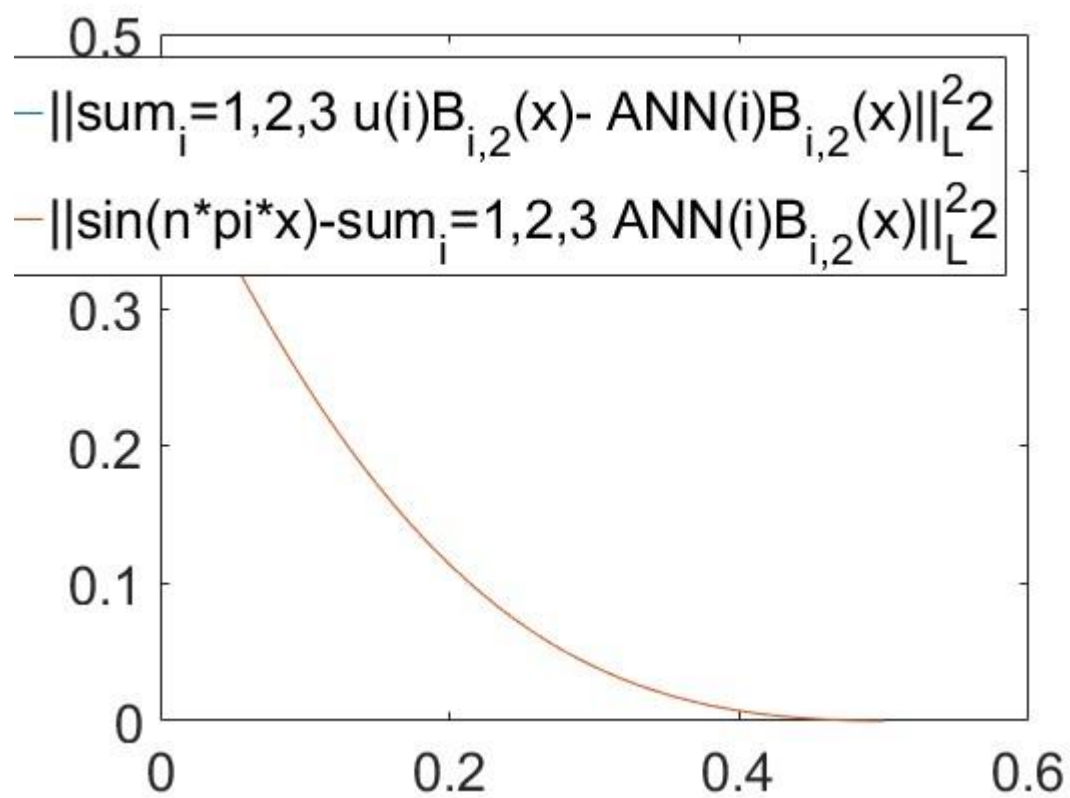
2. Otrzymane rezultaty

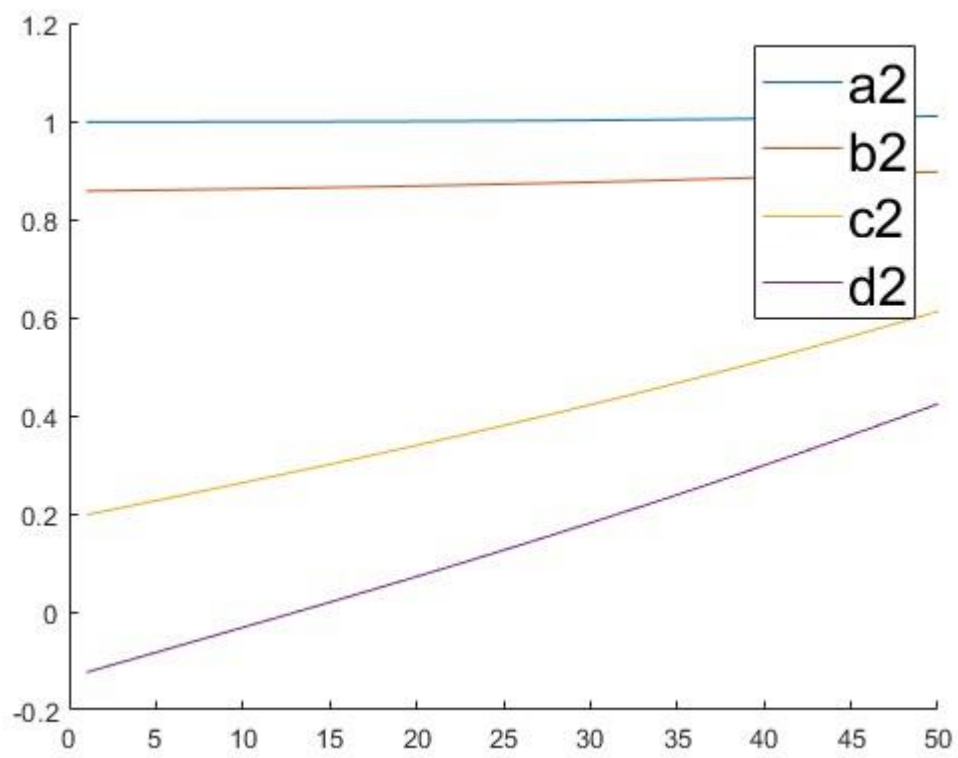
Dla trenowanego modelu bez losowości (linia 12) otrzymujemy

Parametr	Wartość
a_1	0.998317
b_1	0.856309
c_1	0.170493
d_1	-0.159871
a_2	1.010407
b_2	0.896798
c_2	0.612143
d_2	0.423566
a_3	1.008945
b_3	0.906363
c_3	0.655185
d_3	0.492672

Wówczas proces uczenia zbiega się



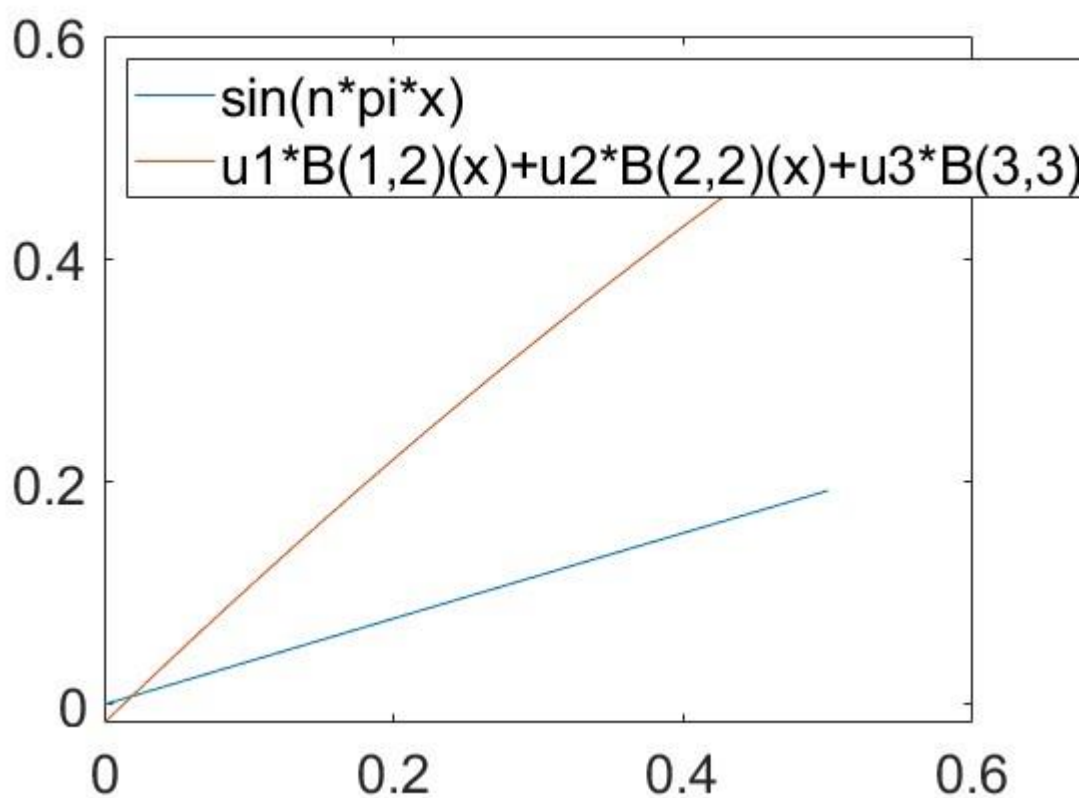


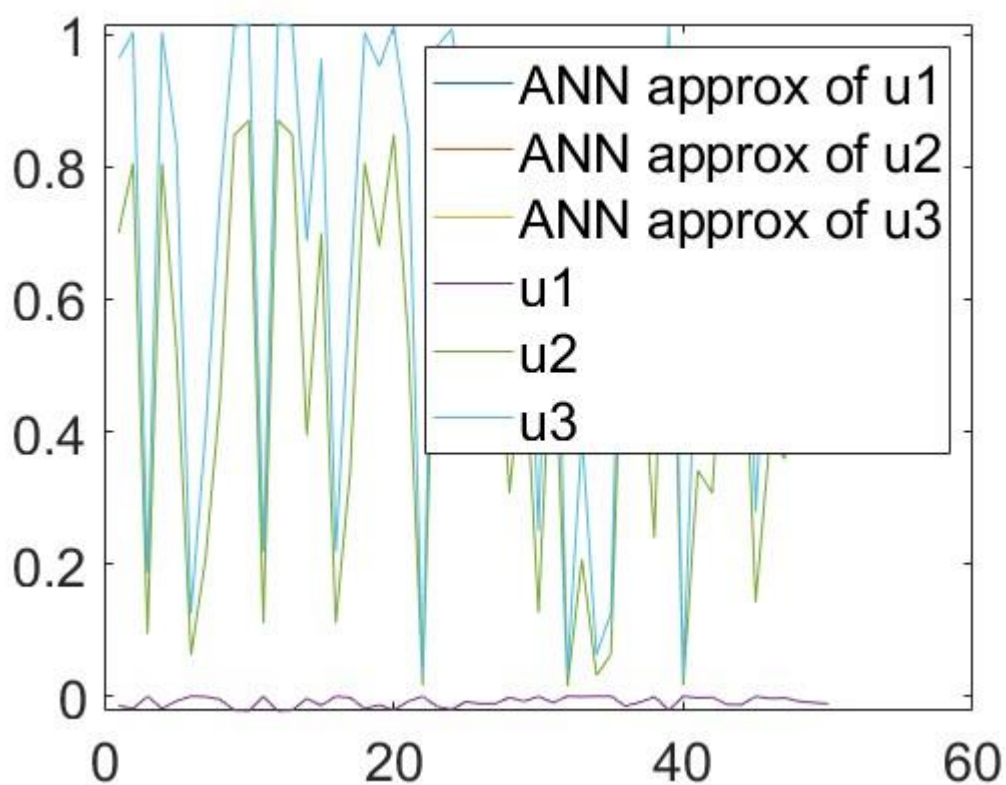
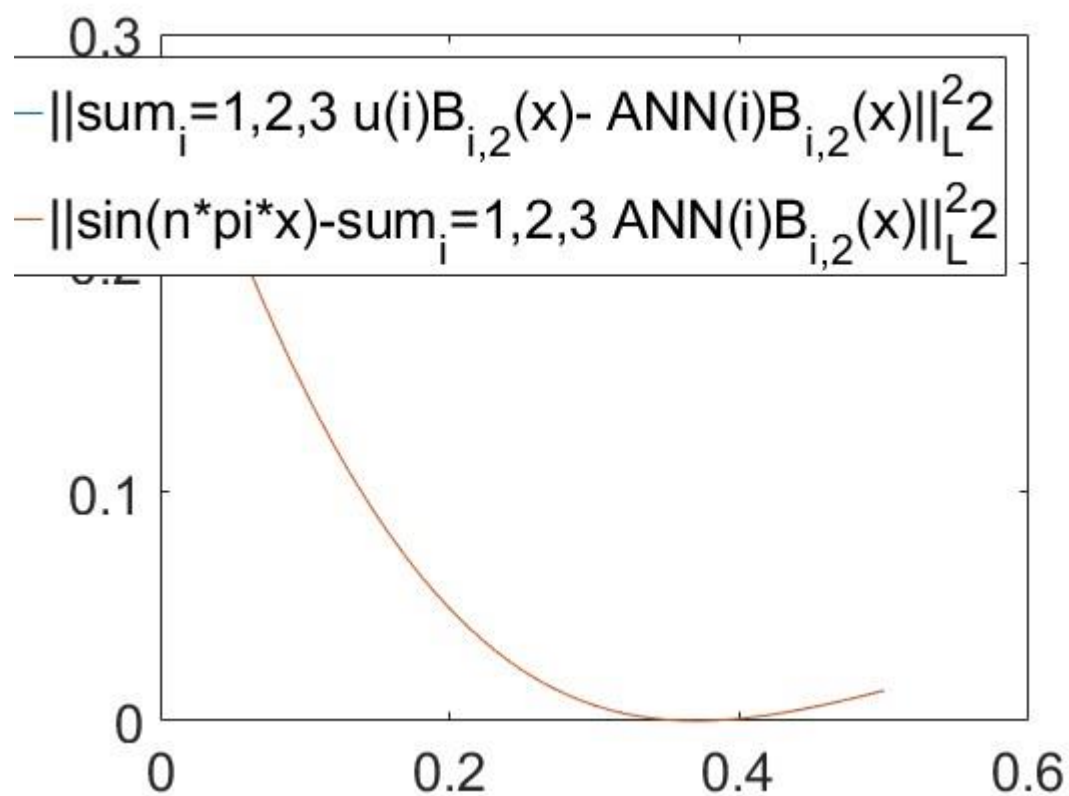


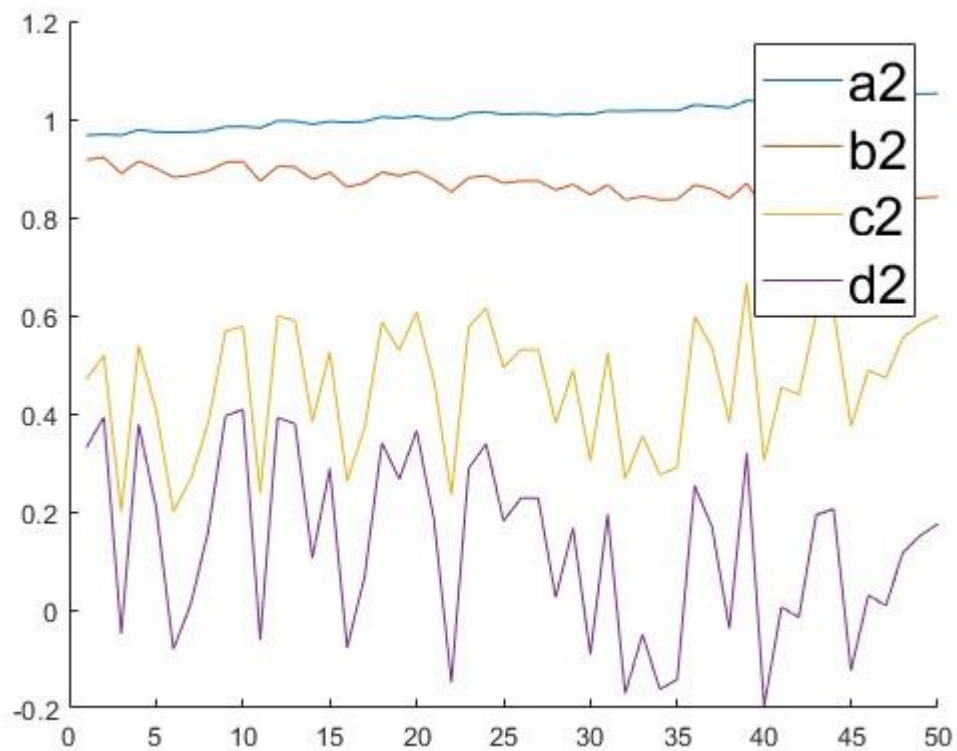
Dla trenowanego modelu z losowością (linie 7-10) otrzymujemy

Parametr	Wartość
a_1	0.955345
b_1	0.891653
c_1	0.120090
d_1	-0.104508
a_2	1.052252
b_2	0.841757
c_2	0.599959
d_2	0.174938
a_3	1.097997
b_3	0.830174
c_3	0.772666
d_3	0.329075

Wówczas proces uczenia także zbiega się







3. Opis znalezienia punktu

Do znalezienia odpowiednich parametrów wykorzystano kod **NNtrain.m** z modyfikacjami opisanymi w rozdziale 1.

Program był uruchamiany dla różnych wartości liczby epok. Po przekroczeniu pewnej liczby epok, rezultaty pozostawały niezmiennie, co wskazuje na osiągnięcie stabilnej zbieżności modelu.