

# Grafičko sučelje u prikazu generalizacijskog svojstva unaprijedne neuronske mreže

---

Špoljarić, Andreja

Master's thesis / Diplomski rad

2016

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

*Permanent link / Trajna poveznica:* <https://urn.nsk.hr/urn:nbn:hr:235:258671>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-16**

*Repository / Repozitorij:*

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# **DIPLOMSKI RAD**

**Andreja Špoljarić**

Zagreb, 2016.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# DIPLOMSKI RAD

Mentor:

Prof. dr. sc. Dubravko Majetić

Student:

Andreja Špoljarić

Zagreb, 2016.

*Izjavljujem da sam ovaj rad izradila samostalno koristeći stečena znanja tijekom studija i navedenu literaturu.*

*Posebno se zahvaljujem voditelju rada prof.dr.sc. Dubravku Majetiću na prihvaćanju mentorstva, pružanju korisnih savjeta te stručne pomoći pri izradi rada.*

*Također zahvaljujem baki, roditeljima te kolegama na pruženoj podršci i potpori.*

*Andreja Špoljarić*



SVEUČILIŠTE U ZAGREBU  
**FAKULTET STROJARSTVA I BRODOGRADNJE**



Središnje povjerenstvo za završne i diplomske ispite  
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:  
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje
Datum 07-07-2016 Prilog
Klasa: 602-04/16-6/3
Ur.broj: 15-1703-16-261

## DIPLOMSKI ZADATAK

Student: **Andreja Špoljarić** Mat. br.: 0035188655

Naslov rada na hrvatskom jeziku: **Grafičko sučelje u prikazu generalizacijskog svojstva unaprijedne neuronske mreže**

Naslov rada na engleskom jeziku: **Neural network graphical interface in demonstration of feed forward neural network generalisation property**

Opis zadatka:

U radu je potrebno načiniti interaktivno grafičko sučelje za komunikaciju između korisnika i algoritama umjetne neuronske mreže. Grafičko sučelje treba pomoći korisniku u radu s neuronskom mrežom. U prvom redu treba pomoći u kreiranju same datoteke učenja. Potom treba omogućiti odabir strukture i način učenja neuronske mreže te osigurati jednostavan postupak testiranja naučene neuronske mreže. Konačno, grafičko sučelje treba pomoći korisniku u razumijevanju generalizacijskog svojstva neuronske mreže. To svojstvo treba pokazati i analizirati na različitim zadanim primjerima.

U radu treba načiniti slijedeće:

1. Izvesti matematički model učenja povratnog prostiranja pogreške za statičku neuronsku mrežu s jednim sakrivenim slojem neurona, pri čemu treba predvidjeti odabir raznih aktivacijskih funkcija neurona skrivenog sloja.
2. Radi ubrzanja procesa učenja matematički model učenja proširiti metodom zamaha prvog i drugog reda.
3. U postupku analize generalizacijskog svojstva naučene mreže odabrati primjere s više ulaznih i izlaznih podataka.
4. Usporediti učenje, testiranje i generalizaciju naučenih neuronskih mreža s različitim aktivacijskim funkcijama skrivenog sloja mreže.
5. Za zadane probleme izraditi interaktivno grafičko sučelje.
6. Programsku podršku načiniti u nekom od dostupnih matematičkih programskih paketa.
7. Izvesti zaključke rada.

Zadatak zadan:

5. svibnja 2016.

Rok predaje rada:

7. srpnja 2016.


Predvideni datum obrane:

13., 14. i 15. srpnja 2016.

Zadatak zadao:

  
Prof. dr. sc. Dubravko Majetić

Predsjednik Povjerenstva:

  
Prof. dr. sc. Franjo Cajner

**SADRŽAJ**

SADRŽAJ	I
POPIS SLIKA	III
POPIS TABLICA	V
POPIS OZNAKA	VIII
SAŽETAK	IX
SUMMARY	X
1. UVOD	1
1.1 Umjetna neuronska mreža	1
1.1.1 Podjela umjetnih neuronskih mreža	2
1.1.2. Učenje neuronske mreže	2
1.1.2.1 Algoritmi učenja neuronske mreže	3
1.2 Biološki neuron	4
1.3 Umjetni neuron	5
2. UNAPRIJEDNA STATIČKA JEDNOSLOJNA NEURONSKA MREŽA	6
2.1 Perceptron	6
2.2 Delta pravilo	7
2.3 Višeslojne neuronske mreže	8
2.3.1 Statički neuron	11
2.3.2 Statička neuronska mreža	12
2.4 Učenje povratnim rasprostiranjem pogreške	12
2.4.1 Unaprijedna faza učenja	12
2.4.1.1. Skriveni sloj	13
2.4.1.2. Izlazni sloj	13
2.4.2 Povratna faza učenja	14
2.4.2.1 Promjena težina izlaznog sloja	16
2.4.2.2 Promjena težina sakrivenog sloja	18
2.4.3 Točnost algoritma učenja	22
3. GRAFIČKO KORISNIČKO SUČELJE	24
3.1 Programski paket MATLAB	24
3.2 GUI (eng. Grahic User Interface)	24
3.3 Rad sa grafičkim sučeljem	24
3.4 Primjer rada mreže	36
4. ISPITIVANJE RADA JEDNOSLOJNE NEURONSKE MREŽE	40
4.1 Ispitivanje mreže s dva ulaza i jednim izlazom	40
4.1.1 Broj neurona skrivenog sloja	40
4.1.2 Aktivacijska funkcija	42
4.1.3 Normalizirani korijen srednje kvadratne pogreške	44
4.1.4 Brzina učenja	45
4.1.5 Momentum prvog i drugog reda	45
4.1.6 Testiranje mreže s dva ulaza i jednim izlazom	58
4.2 Ispitivanje mreže s dva ulaza i tri izlaza	65

4.2.1 Testiranje mreže s dva ulaza i tri izlaza	65
4.3 Ispitivanje mreže s dva ulaza i četiri izlaza	69
4.3.1 Promjena širine i centra Gaussove funkcije	71
5. UNAPRIJEDNA STATIČKA DVOSLOJNA NEURONSKA MREŽA	72
5.1 Unaprijedna faza učenja dvoslojne neuronske mreže	73
5.1.1 Prvi sakriveni sloj	73
5.1.2 Drugi sakriveni sloj	74
5.1.3 Izlazni sloj	75
5.2 Povratna faza učenja dvoslojne neuronske mreže	75
5.2.1 Promjena težina izlaznog sloja	75
5.2.2 Promjena težina drugog izlaznog sloja	76
5.2.3 Promjena težina prvog izlaznog sloja	80
6. ISPITIVANJE RADA DVOSLOJNE NEURONSKE MREŽE	84
6.1 Testiranje dvoslojne mreže sa dva ulaza i jednim izlazom	85
6.2 Testiranje dvoslojne mreže sa dva ulaza i četiri izlaza	92
7. ZAKLJUČAK	96
8. LITERATURA	98
9. PRILOG	99

**POPIS SLIKA**

Slika 1.1 Umjetni neuron	5
Slika 2.1 Model statičkog neurona	8
Slika 2.2 Model statičke unaprijedne neuroske mreže	11
Slika 3.1 Početak rada programa	25
Slika 3.2 Izbor željenih kombinacija brojeva	25
Slika 3.3 Opis rada programa	26
Slika 3.4 Redoslijed spremanja odabranih kombinacija	28
Slika 3.5 Izbor već spremljenih kombinacija brojeva	29
Slika 3.6 Definiranje parametara učenja	29
Slika 3.7 Moguće opcije prilikom odabira parametara	30
Slika 3.8 Definicija parametara Gaussianove funkcije	30
Slika 3.9 Odabir proizvoljnih težina	31
Slika 3.10 Odabir postojećih težina	31
Slika 3.11 Odabir momentuma	32
Slika 3.12 Grafički prikaz promijene NRMS-a u svakom koraku učenja	33
Slika 3.13 Grafički prikaz ovisnosti NRMS-a o koraku nakon učenja	33
Slika 3.14 Nastavak učenja mreže	34
Slika 3.15 Test mreže s jednim izlazom	35
Slika 3.16 Test mreže s tri izlaza	35
Slika 3.17 Test mreže s četiri izlaza	36
Slika 3.18 Početak rada mreže s dva ulaza i tri izlaza	36
Slika 3.19 Primjer definiranja parametara	37
Slika 3.20 Grafički prikaz promjene NRMS-a za 2000 koraku	37
Slika 3.21 Grafički prikaz promjene NRMS-a za 42000 koraka	38
Slika 3.22 Testiranje mreže nakon 40251 koraka	39
Slika 4.1 Grafički prikaz promjene NRMS-a po koraku za različite vrijednosti matrica težina $W$ i $V$	41
Slika 4.2 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija i 5 neurona skrivenog sloja bez momentuma	43
Slika 4.3 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija dijelovanjem momentuma prvog reda 0.3	46
Slika 4.4 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija dijelovanjem momentuma prvog reda 0.5	47
Slika 4.5 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija dijelovanjem momentuma prvog reda 0.7	47
Slika 4.6 Grafički prikaz promjene NRMS-a po koraku za bipolarnu i unipolarnu sigmoidalnu funkciju primjenom momentuma prvog reda 0.3, 0.5 i 0.7	48
Slika 4.7 Grafički prikaz promjene NRMS-a po koraku za sinusnu i Gaussianovu funkciju primjenom momentuma prvog reda 0.3, 0.5 i 0.7	48
Slika 4.8 Grafički prikaz promjene NRMS-a po koraku za bipolarnu i unipolarnu sigmoidalnu funkciju primjenom momentuma drugog reda 0.3, 0.5 i 0.7	49



Slika 4.9 Grafički prikaz promjene NRMS-a po koraku za sinusnu i gaussianovu funkciju primjenom momentuma drugog reda 0.3, 0.5 i 0.7	50
Slika 4.10 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3	51
Slika 4.11 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5	51
Slika 4.12 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7	52
Slika 4.13 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3	53
Slika 4.14 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5	53
Slika 4.15 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7	54
Slika 4.16 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3	55
Slika 4.17 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5	55
Slika 4.18 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7	56
Slika 4.19 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3	57
Slika 4.20 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5	57
Slika 4.21 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7	58
Slika 5.1 Prikaz strukture dvoslojne neuronske mreže	72
Slika 6.1 Grafičko sučelje za definiranje parametara učenja dvoslojne neuronske mreže sa dva ulazna i jednim izlaznim neuronom	84

**POPIS TABLICA**

Tablica 2.1 Aktivacijske funkcije korištene u diplomskom radu	9
Tablica 3.2 Prikaz rezultata učenja mreže nakon 40251 koraka	38
Tablica 4.1 Parametri zadani u procesu učenja	40
Tablica 4.2 Vrijednost koraka učenja i vrijeme izračuna za parametre zadane u tablici 4.1	40
Tablica 4.3 Utjecaj težina na korake i vrijeme učenja	40
Tablica 4.4. Stvarna vrijednost odziva mreže	42
Tablica 4.5. Zadani parametri prilikom ispitivanja aktivacijskih funkcija	42
Tablica 4.6. Konačni koraci i vrijeme prilikom ispitivanja aktivacijskih funkcija	43
Tablica 4.7 Stvarni odziv mreže za različite aktivacijske funkcije i jednake vrijednosti težina	43
Tablica 4.8 Zadani parametri prilikom ispitivanja NRMS-a	44
Tablica 4.9 Prikaz rezultata testiranja za vrijednosti NRMS-a 0.01 i 0.03	44
Tablica 4.10. Prikaz rezultata testiranja za vrijednosti $\eta$ 0.1, 0.05 i 0.5	45
Tablica 4.11. Zadani parametri prilikom ispitivanja momentuma prvog reda	45
Tablica 4.12. Prikaz rezultata testiranja za vrijednosti momentuma prvog reda 0.3, 0.5 i 0.7	46
Tablica 4.13 Zadani parametri prilikom ispitivanja momentuma drugog reda	48
Tablica 4.14 Prikaz rezultata testiranja za vrijednosti momentuma drugog reda 0.3, 0.5 i 0.7	49
Tablica 4.15 Prikaz rezultata testiranja različitih momentuma za bipolarnu sigmoidalnu funkciju	50
Tablica 4.16 Prikaz rezultata testiranja različitih momentuma za unipolarnu sigmoidalnu funkciju	52
Tablica 4.17 Prikaz rezultata testiranja različitih momentuma za sinusnu funkciju	54
Tablica 4.18 Prikaz rezultata testiranja različitih momentuma za Gaussianovu funkciju	56
Tablica 4.19 Prikaz 25 mogućih kombinacija potrebnih za učenje mreže	58
Tablica 4.20 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom	59
Tablica 4.21 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 iz procesa učenja mreže	59
Tablica 4.22 Prikaz rezultata testiranja za 9 ne učenih kombinacija	59
Tablica 4.22 Prikaz rezultata testiranja za 9 ne učenih kombinacija	60
Tablica 4.24 Prikaz rezultata testiranja za 16 ne naučenih kombinacija	60
Tablica 4.25 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 1 i 2 radi testiranja Gaussianovom funkcijom	61
Tablica 4.26 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1	61
Tablica 4.27 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 2	62
Tablica 4.28 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže prvi broj 3 i drugi broj 4	62

Tablica 4.29 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 3 i drugi broj 4	62
Tablica 4.30 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 i prvi broj 2	63
Tablica 4.31 Prikaz rezultata testiranja za 14 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 2	63
Tablica 4.32 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže brojeve 2 i 4	64
Tablica 4.33 Prikaz rezultata testiranja za 16 ne naučenih kombinacija koje u sebi sadrže brojeve 2 i 4	64
Tablica 4.34 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3	65
Tablica 4.35 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3	65
Tablica 4.36 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 i prvi broj 4	66
Tablica 4.37 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 4	66
Tablica 4.38 Zadani parametri prilikom novog ispitivanja mreže	67
Tablica 4.39 Prikaz rezultata testiranja za 14 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 4 nakon promjene NRMS-a sa 0.03 na 0.01	67
Tablica 4.40. Zadani parametri prilikom ispitivanja sinusne funkcije	68
Tablica 4.41 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže prvi broj 2 i drugi broj 4	68
Tablica 4.42 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 2 i drugi broj 4	68
Tablica 4.43 Zadani parametri prilikom ispitivanja sinusne funkcije s manjim NRMS-om	69
Tablica 4.44 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 2 i drugi broj 4 nakon smanjenja NRMS -a s 0.03 na 0.01	69
Tablica 4.45 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3	70
Tablica 4.46 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 za NRMS 0.03	70
Tablica 4.46 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 za NRMS 0.01	71
Tablica 4.47 Tablica promjena centra i širine gaussianove funkcije	71
Tablica 6.1 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 iz procesa učenja mreže	85
Tablica 6.2 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom	85
Tablica 6.3 Prikaz rezultata testiranja za 9 ne učenih kombinacija	85
Tablica 6.4 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom	86
Tablica 6.5 Prikaz rezultata testiranja za 9 ne učenih kombinacija	86
Tablica 6.6 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom	87
Tablica 6.7 Prikaz rezultata testiranja za 9 ne učenih kombinacija	87
Tablica 6.8 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 1 iz procesa učenja mreže	88

Tablica 6.9 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1	88
Tablica 6.10 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 5 iz procesa učenja mreže	89
Tablica 6.11 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1	89
Tablica 6.12 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 2 i 4 iz procesa učenja mreže	90
Tablica 6.13 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom	90
Tablica 6.14 Prikaz rezultata testiranja za 16 ne učenih kombinacija	90
Tablica 6.15 Parametri učenja dvoslojne neuronske mreže s dva ulaza i jednim izlazom	91
Tablica 6.16 Prikaz rezultata testiranja za 16 ne učenih kombinacija sa različitim vrijednostima NRMS-a	91
Tablica 6.17 Prikaz utjecaja promjene neurona u sakrivenim slojevima na korake i vrijeme učenja	92
Tablica 6.18 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 2 i 4 iz procesa učenja mreže	92
Tablica 6.19 Parametri učenja neuronske mreže s dva ulaza i četiri izlaza	93
Tablica 6.20 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Gaussianove funkcije	93
Tablica 6.21 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Bipolarne sigmoidalne funkcije	93
Tablica 6.22 Parametri učenja neuronske mreže s dva ulaza i četiri izlaza	94
Tablica 6.23 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Bipolarne sigmoidalne funkcije	94
Tablica 9.1 Prikaz grafičkog sučelja	99

## POPIS OZNAKA

<i>BIAS</i>	neuron bez ulaza s konstantnom izlaznom vrijednosti jednakom 1
<i>c</i>	centar Gaussove funkcije
$d_k$	željena vrijednost k-og izlaza neuronske mreže
<i>E</i>	funkcija cilja
<i>I</i>	broj neurona ulaznog sloja mreže
<i>J</i>	broj neurona sakrivenog sloja mreže
<i>K</i>	broj neurona izlaznog sloja mreže
$K_p$	pojačanje sustava
<i>N</i>	broj parova ulazno-izlaznih vrijednosti seta učenja
<i>n</i>	broj značajki, broj neurona ulaznog sloja
<i>NRMS</i>	normalizirani korijen srednje kvadratne pogreške
<i>net</i>	vrijednost funkcije sume
$net_j$	vrijednost funkcije sume j-og neurona
$net_{Hj}$	vrijednost funkcije sume j-og neurona sakrivenog sloja
$net_{Ok}$	vrijednost funkcije sume j-og neurona izlaznog sloja
$O_k$	k-ti izlaz neuronske mreže
<i>RMS</i>	korijen srednje kvadratne pogreške
<i>T</i>	vremenska konstanta
$v_{ji}$	težinski koeficijent veze između j-tog neurona sakrivenog sloja i i-tog neurona ulaznog sloja
$w_{kj}$	težinski koeficijent veze između k-tog neurona izlaznog sloja i j-tog neurona sakrivenog sloja
<i>y</i>	vrijednost izlaza neurona
$y_j$	vrijednost izlaza j-og neurona sakrivenog sloja
$Z_i$	i-ti ulaz neuronske mreže
$\vartheta$	parametar učenja (težinski koeficijent)
$\vartheta(n)$	trenutna vrijednost parametra učenja
$\vartheta(n + 1)$	nova vrijednost parametra učenja
$\Delta\vartheta(n)$	tekuća promjena parametra učenja
$\alpha$	vrijednost momentuma
$\Delta$	iznos promjene težine u jednom koraku
$\delta$	parametar algoritma povratnog prostiranja pogreške
$\delta_{Ok}$	parametar algoritma povratnog prostiranja pogreške izlaznog sustava
$\gamma$	aktivacijska funkcija neurona
$\gamma'$	derivacija aktivacijske funkcije neurona
$\eta$	koeficijent brzine učenja
$\sigma$	standardno odstupanje, širina Gaussove funkcije
$\Delta E$	promjena pogreške
$\nabla E$	gradijent pogreške

## SAŽETAK

U ovom diplomskom radu detaljno je prikazan rad jednoslojne neuronske mreže sa povratnim rasprostiranjem pogreške. Prilikom učenja neuronske mreže korisnik može izabrati između četiri aktivacijske funkcije i dva momentuma. Također, korisnik ima mogućnost upisa željenog broja neurona skrivenog sloja, iznosa brzine učenja, iznosa NRMS-a te broja koraka učenja. Broj ulaznih i izlaznih neurona definirani su samim zadatkom učenja. Rezultat učenja korisnik dobiva u tabličnom obliku te pomoću grafičkog prikaza. Korisniku je omogućeno testiranje neuronske mreže u kojem korisnik osim zadanih brojeva može testirati i svoje vrijednosti brojeva koji nisu definirani u procesu učenja. Rad mreže korisnik može provjeriti izborom strukture s dva ulaza i s jednim, tri ili četiri izlaza. Ulazi u proces učenja su brojevi od jedan do pet te njihove kombinacije korisnik odabire na samom početku rada mreže prije odabira parametara učenja. Izlazi iz neuronske mreže su rezultati različitih matematičkih operacija za zadane ulazne kombinacije brojeva. Princip rada i učenja neuronske mreže prikazan je u prvom dijelu diplomskog rada, dok su primjeri rada i testiranje mreže prikazani u drugom dijelu rada. Na kraju rada prikazan je i objašnjen rad dvoslojne neuronske mreže sa povratnim rasprostiranjem pogreške, također sa dva ulazna neurona te jednim i četiri izlazna neurona. Grafičko sučelje za svaki primjer izrađeno je u programskom paketu MATLAB.

## Ključne riječi

- neuronska mreža
- aktivacijska funkcija
- neuroni skrivenog sloja
- momentum
- brzina učenja
- NRMS
- koraci učenja
- grafičko sučelje

## **SUMMARY**

In this thesis is in detail explained work with one-layer neural network with backpropagation. During the neural network learning process user can choose between four activation functions and two momentums. Also, user have the opportunity to write the wanted number of hidden layer neurons, the amount of learning speed, the amount of NRMS and number of learning steps. The number of input and output neurons are defined by single task learning. The learning results user can get in a tabular format or with graphic display. The user can test the neural networks with given numbers and with numbers that are not defined in the learning process. User can check neural networks operations if he use structure with two inputs and one, three or four outputs. Inputs in the learning process are numbers between one and five, and their combinations user selects on the start, and after that user can select neural network learning parameters. Outputs from neural network are results of different mathematical operations for the default input combinations of numbers. Work principle and neural network learning process is explained in the first part of this thesis, while examples of work and testing of the neural networks are shown in the second part. At the end of paper is shown and explained the work of two-layer neural network with backpropagation, also with two inputs and with one and four outputs. The graphical interface for each example is made in MATLAB.

## **Keywords**

- neural network
- activation function
- hidden layer neurons
- momentum
- speed of learning
- NRMS
- steps learning
- GUI

# 1. UVOD

## 1.1 Umjetna neuronska mreža

Razvoj umjetnih neuronskih mreža započinje zahvaljujući nizu pokušaja modeliranja biofiziologije mozga čovjeka. Prvobitna namjena ovakvih sustava bila je da omogući bolje razumijevanje i objašnjenje kako ljudski mozak funkcionira. Tako dolazi do kreiranja prvog modela neuronske mreže, kojeg karakterizira sposobnost procesuiranja informacija analogno aktivnostima mozga [1].

Neuronske mreže međusobno se razlikuju po strukturi veza među neuronima i neurona s okruženjem mreže, te po metodologiji određivanja težina mreže, što predstavlja proces učenja mreže. [2] U svijetu se trenutno koriste dva pristupa modeliranju umjetnih neuronskih mreža, a sadrže podosta zajedničkih elemenata. Kod prvog pristupa potrebno je realizirati modele koji dovoljno točno oponašaju aktivnosti mozga čovjeka što u konačnici rezultira sustavom umjetne inteligencije. Drugi pristup odnosi se na razvoj mreža s velikim računskim sposobnostima i sa mogućnošću rješavanja praktičnih zadataka tj. neuroračunala.

Na osnovi navedenog može se zaključiti da je umjetna neuronska mreža u pravilu vrlo složen sustav sastavljen od mnoštva elementarnih jedinica, neurona. Za vrijeme rada mreže neuroni su u međusobnoj interakciji te u interakciji sa okolinom gradeći pri tom funkcionalnu cjelinu. Kao takve, umjetne neuronske mreže potrebno je razmatrati u kontekstu umjetne inteligencije kao jedne od elemenata potrebne za njenu realizaciju, slično kao i ekspertni sustavi, neizrazita logika, metode rezoniranja, zaključivanja, spoznaje i odlučivanja.

Postoji veliki broj mogućih implementacija umjetnih neuronskih mreža, pa se tako u tu svrhu može koristiti analogna tehnika, digitalna tehnika, hibridna tehnika, tehnika impulsne modulacije i mnoge druge. Analogne tehnike implementacije koriste se najviše kod povratnih ili Hopfieldovih mreža, dok se digitalne tehnike uglavnom koriste kod unaprijednih mreža [1].

Umjetne neuronske mreže imaju široko područje primjene, a njihov značaj i razvoj u stalnom su porastu. Tako mreže danas nalaze primjenu u postupcima optimiranja, linearnog programiranja, komuniciranja, donošenja odluka te u mnogim drugim područjima. Također se koriste i kod klasificiranja signala, vođenju sustava i robotici, a svoju primjenu našle su i u medicini gdje se koriste za potrebe dijagnosticiranja bolesti, ultrazvučnog prikazivanja te u mnoge druge svrhe. Dosta je često korištenje neuronskih mreža u kombinaciji s ekspertnim sustavima i neizrazitom logikom, te se tako pomoću mreža mogu riješiti neki do nedavno nerješivi problemi poput prepoznavanja uzoraka, paralelnog procesuiranja te mnogi drugi. Posebno je značajna primjena neuronskih mreža u robotici, gdje se neuronske mreže koriste kod identifikacije dinamike robota, rješavanja inverznog kinematičkog problema, automatsko planiranje trajektorija robota, optimalno i adaptivno vođenje robota u prostoru i vremenu te kod mnogih drugih zadataka i problema. [3]



### **1.1.1 Podjela umjetnih neuronskih mreža**

Postoji više podjela neuronskih mreža, ovisno o postavljenom kriteriju. Prva podjela neuronskih mreža je na jednoslojne neuronske mreže te na višeslojne neuronske mreže. Karakteristika jednoslojnih neuronskih mreža je da uz izlazne i ulazne slojeve imaju samo jedan skriveni sloj. Iz naziva je vidljivo da višeslojne neuronske mreže, osim ulaznih i izlaznih slojeva, sadrže i više skrivenih slojeva. Svaki sloj neuronske mreže sačinjava paralelno složen skup neurona.

Druga podjela neuronskih mreža je na unaprijedne i povratne neuronske mreže. Karakteristika unaprijednih je da signal putuje samo u jednom smjeru od ulaza prema izlazu, dok kod povratnih neuronskih mreža postoji barem jedna povratna petlja što u sam proces rada mreže uključuje i vrijeme. Odabir unaprijedne ili povratne neuronske mreže ovisi o problemu koji mreža treba riješiti. Povratne neuronske mreže, s obzirom da ovisi i o vremenu, mogu biti još i vremensko – kontinuirane te vremensko - diskretne.

Prema glavnom području primjene razlikujemo perceptronske, asocijativne, dvostruko asocijativne, adaptivne te mnoge druge mreže. Što se načina realiziranja tiče razlikujemo softverske, hardverske te optičke neuronske mreže. Neke mreže dobile su naziv po metodama koje koriste za učenje, pa tako imamo povratno propagiranje, suprotno propagirane te statičke neuronske mreže [1].

### **1.1.2. Učenje neuronske mreže**

Razlikujemo supervizorno i nesupervizorno učenje neuronske mreže. Supervizorno učenje je učenje uz nadzor što znači da zahtjeva vanjskog učitelja neuronske mreže koji upravlja mrežom i određuje njeno ponašanje. Kako bi se provelo supervizorno učenje potrebno je usvojiti strukturu mreže, tj. broj ulaza, broj neurona, broj slojeva, broj izlaza, te broj težina mreže. Prilikom učenja mreža skup izlaznih varijabli uspoređuje sa željenim skupom izlaznih varijabli, a razlika između stvarnih i željenih izlaza mreže naziva se pogreška mreže. Dobivena pogreška koristi se za računanje novih težina mreže preko usvojenog algoritma, a cijeli postupak interakcijski se ponavlja sve dok pogreška mreže ne bude manja od unaprijed zadanog iznosa. Po potrebi se broj neurona, broj slojeva i broj težina, koji čine strukturu mreže, mogu mijenjati. Testiranje mreže slijedi nakon učenja mreže, a provodi se pomoću novog skupa ulaza mreže koji nije bio sadržan u ulaznom skupu za vrijeme procesa učenja. Kod testiranja mreža producira nove ulaze mreže i uspoređuje ih sa željenim izlazima, ne mijenjajući težine mreže. I kod procesa testiranja javljaju se pogreške testiranja koje služe za ocjenu generalizacijskih svojstva mreže, odnosno sposobnosti mreže da daje zadovoljavajuće rezultate i za skup ulaza kojim nije bila učenja.

Kod nesupervizornog učenja neuronska mreža se sama organizira, zbog čega se ove mreže nazivaju i samoorganizacijskim neuronskim mrežama. Na ulaz samoorganizacijskih mreža dovodi se skup ulaznih varijabli, a mreža se samoorganizira podešavanjem svojih parametara po definiranom algoritmu. Rezultat učenja kod ovih mreža nije lako predvidiv s obzirom da

željeni izlaz mreže nije specificiran za vrijeme učenja. Od takvih neuronskih mreža može se zahtijevati da slični skupovi ulaza aktiviraju izlaz jednog te istog neurona. Mreža organizirana na taj način može se koristiti za prepoznavanje uzoraka. I kod ovakvih mreža nakon procesa učenja slijedi testiranje.

Bitna karakteristika umjetnih neuronskih mreža jest sposobnost učenja novih pojmova, uz istovremeno pamćenje već naučenih pojmova. U nekim slučajevima zahtjeva se i izuzetno brzo učenje neuronskih mreža pa je za takve potrebe razvijen cijeli niz algoritama učenja, poput neiterativnih postupaka učenja tj. učenja u jednom koraku. Dosta su popularne mreže sa radijalnim baznim funkcijama [5, 6], kod kojih se težine izlaznog sloja računaju neiterativno, dok se težine skrivenog sloja računaju iterativno nekim od gradijentnih postupaka. Također se pored konvencionalnih izlaznih aktivacijskih funkcija uvode ulazne i interakcijske aktivacijske funkcije te ulazne i izlazne interakcijske matrice težina mreže [7]. Posebno je popularno korištenje unaprijedne neuronske mreže sa povratnim rasprostiranjem pogreške. Ova metoda koristi se i kod dinamičkih neuronskih mreža koje u skrivenom sloju imaju ugrađen dinamički neuron [4].

#### *1.1.2.1 Algoritmi učenja neuronske mreže*

Postoje različiti algoritmi učenja neuronskih mreža, od kojih su najpoznatiji: generalizirano delta pravilo, povratno propagiranje, Grossbergovo (Hebbianovo) učenje, primjena Lyapunovih funkcija, optimalno učenje uz zadani kriterij optimalnosti, te statičko i adaptivno učenje [1].

U neuronskoj mreži prikazanoj u ovom radu koristit će se generalizirano delta pravilo te povratno propagiranje. Delta predstavlja razliku stvarnih i željenih vrijednosti, a tijekom generalizacije delta pravila, starim vrijednostima parametara mreže dodaju se promjene tih parametara koji se računaju u funkciji pogreške mreže. Generirano delta pravilo često se proširuje dodatkom momentuma, tj. zamaha, koji služi za izbjegavanje lokalnih minimuma, pa pogreška mreže konvergira globalnom minimumu.

Povratno propagiranje koristi se kod mreža koje imaju najmanje tri sloja, a to su ulazni, skriveni i izlazni sloj. Moguće je i povećavanje skrivenih slojeva, ovisno o potrebi. Na početku učenja težine učenja postavljene su tako da su njihovi iznosi mali slučajni brojevi, a potom slijedi supervizorno učenje. Nakon prvog prolaza signala s ulaza na izlaz na osnovi pogreške mreže najprije se podešavaju parametri, odnosno težine izlaznog sloja, koristeći se generalnim delta pravilom. Potom slijedi podešavanje skrivenih slojeva mreže redom od izlaza prema ulazu, tj. povratno propagiranje. Algoritmi povratnog propagiranja razvijeni su i u početku korišteni samo kod unaprijednih neuronskih mreža, ali u međuvremenu su dodatno modificirani tako da se danas koriste i kod dinamičkih neuronskih mreža [4].

## 1.2 Biološki neuron

Procjenjuje se da mozak čovjeka ima preko 100 milijardi neurona [8], koji međusobno komuniciraju preko neuronskih vlakana, sinapsa, gradeći tako 100 trilijuna veza. Jasno je na temelju tog podatka da se radi o vrlo složenoj biološkoj mreži. Upravo ona je odgovorna za naše razmišljanje, učenje, emociju, percepciju i spoznaju te izvođenje motorne sensorike u autonomnih funkcija. Međutim, način rada biološke mreže nije u potpunosti razumljiv, zato što je većina informacija u vezi funkcioniranja biološkog neurona još uvijek nepoznata.

Poznato je da se biološki neuron sastoji od tijela neurona, aksona i dendrita, te da je veličine 5 do 100 mikrona u promjeru. Mali razmak između završetaka aksona (oblik cjevčice) prethodnog neurona i dendrita ili tijela sljedećeg neurona naziva se sinapsa. Jedan kraj aksona povezan je na tijelo neurona, drugi je u obliku niza razgranatih grana, a pomoću sinapsa stvaraju se sinaptičke veze s mnoštvom drugih neurona. Impulsi tj. izlazi neurona putuju kroz akson do sinapse te se odatle signali različitog intenziteta šalju kroz dendrite ili direktno na tijelo drugih neurona, prilikom čega intenzitet prijenosa ovisi efikasnosti svakog pojedinog sinaptičkog prijenosa.

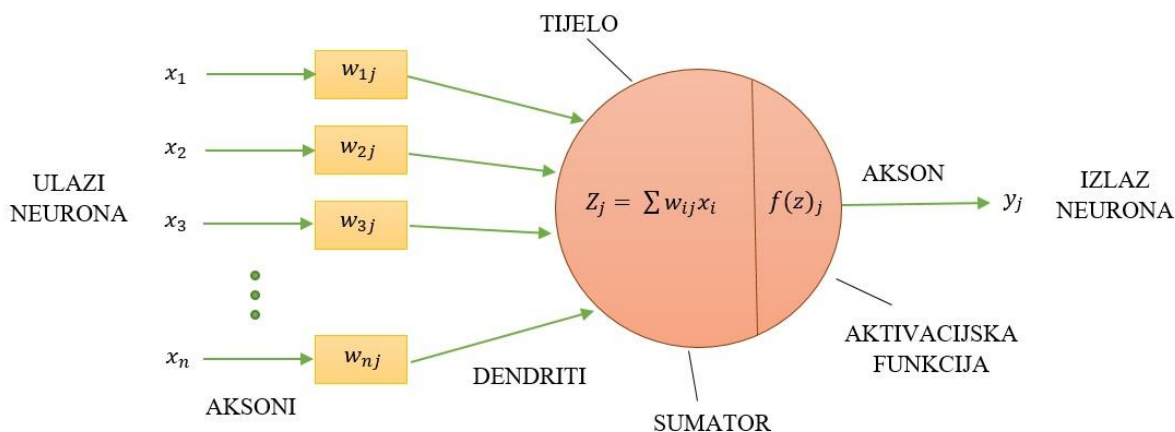
Pojedini neuron ima mogućnost generiranja impulsa koji unutar neuronske mreže može aktivirati stotine i tisuće drugih neurona, a ujedno na svaki neuron može istovremeno djelovati stotine i tisuće drugih neurona. Iz navedenog je vidljivo da su neuroni izuzetno dobro povezani unutar neuronske mreže, što dovodi do zaključka da se kompleksne funkcije neuronske mreže ostvaruju kompletnošću odnosnu težinama među neuronima, prije nego isključivo kompleksnošću svakog neurona posebno.

Važno je primijetiti da postoji određena veza između mozga čovjeka i računala [1], jer oba koriste električne signale, sadrže velik broj jednostavnih elemenata te izvode funkcije koje se općenito nazivaju računskim funkcijama. Međutim, postoji bitna razlika između mozga i računala. Kako bi računalo pravilno obavilo određen zadatak potrebno ga je programirati, dok čovjek ukoliko želi pravilno izvršiti neki zadatak treba učiti. Zbog toga je brzina prijenosa neuronskih impulsa u mozgu znatno niža u odnosu na računalo i to zahvaljujući velikom broju paralelnih računskih jedinica, neurona. Još jedna bitna razlika između mozga i računala je kvaliteta pogreške. Naime, računalo neće pogriješiti sve dok su mu ulazi, softver i hardver ispravni. S druge strane, mozak daje bolje zaključke i aproksimacije čak i za djelomično nekompletne i nekorektne ulaze. Razlika je i u potrošnji energije, računalo je daleko veći potrošač energije u odnosu na ljudski mozak.

Pretpostavlja se da će umjetne neuronske mreže u perspektivi moći oponašati funkcije mozga čovjeka, ali to će biti moguće tek kad će ljudi bolje razumjeti rad ljudskog mozga. Dosad izrađene neuronske mreže građene su na osnovi izuzetno pojednostavljenog modela koji ne uzima u obzir većinu poznatih informacija o ljudskom mozga, jer još uvijek ne postoji matematički aparat kojim bi se kompleksne poznate funkcije mozga preslikale u odgovarajući model.

### 1.3 Umjetni neuron

Dizajn umjetnog neurona osmišljen je tako da što više sliči biološkom neuronu kako bi oponašao njegove funkcije. Izgled umjetnog neurona prikazan je na slici 1.1.



Slika 1.1 Umjetni neuron

Sumator predstavlja tijelo biološkog neurona, ulazi u sumator predstavljaju dendrite, dok su izlazi iz sumatora jednaki aksonima biološkog neurona. Prag osjetljivosti bioloških sustava preslikan je pomoću aktivacijske funkcije. Pomoću težinskih faktora  $w$  ostvaruju se veze neurona s njegovom okolinom. Ovi težinski faktori mogu biti pozitivni i negativni brojevi, a kod modernih umjetnih neuronski mreža težinski faktor može biti funkcija. U slučaju da je težinski faktor nula, mreža nema vezu s okolinom. Intenzitet veze ovisi o iznosu, a karakter veze o predznaku težinskog faktora.

Izlaz sumatora povezan je na ulaz aktivacijske funkcije, koja na svom izlazu da je odgovarajući izlaz umjetnog neurona. Aktivacijska funkcija može biti linearna i nelinearna. Linearne aktivacijske funkcije karakterizira što je izlaz sumatora množen s nekim faktorom pojačanja te tako tvori izlaz neurona. Nelinearna aktivacijska funkcija može poprimiti različite oblike, a radi tako provodi izlaz sumatora na izlaz neurona preko nelinearnog pojačanja. Funkcija praga osjetljivosti daje na izlazu neurona jedinicu što odgovara ispaljivanju impulsa kod biološkog neurona, dok je u suprotnom iznos neurona nula. Najčešće su korištene nelinearne aktivacijske funkcije: funkcija praga osjetljivosti, sigmoidalne, hiperboličke te harmoničke. Svaka nabrojena aktivacijska funkcija ima svoje područje primjene, gdje posjeduje određene prednosti.

Umjetni neuron često nosi naziv po svojim autorima kao npr. McCulloch-Pittsov neuron [2]. U slučaju da je odabrana aktivacijska funkcija oblika praga osjetljivosti, takav se neuron naziva perceptron [9]. Više o perceptronu bit će rečeno u nastavku.

## 2. UNAPRIJEDNA STATIČKA JEDNOSLOJNA NEURONSKA MREŽA

### 2.1 Perceptron

Binarni neuron čiji perceptron može poprimiti jednu od dvije vrijednosti, 0 ili 1, naziva se perceptron. Model perceptrona prikazan je u poglavlju 1.3 na slici 1.1. Radi se o umjetnom neuronu kod kojeg aktivacijska funkcija koristi binarnu funkciju praga osjetljivosti definiranu izrazom:

$$y_j = f(z_j) = \begin{cases} 1 & \text{ako } z_j > \text{prag} \\ 0 & \text{ako } z_j \leq \text{prag} \end{cases} \quad (2.1)$$

Ukoliko je dobivena vrijednost jedan, znači da je promatrani perceptron neurona aktivan i šalje signal susjednim neuronima. U slučaju vrijednosti nula, promatrani perceptron je neaktivan.

Perceptroni se međusobno mogu povezati i tako tvoriti mrežu. Ako se takva mreža sastoji od više ulaznih i jednog izlaznog sloja govori se o jednoslojnoj perceptronskoj mreži, kakva se obično primjenjuje kod problema raspoznavanja uzoraka [10]. Broj ulaza takve mreže određuje se na temelju značajki koje opisuju problem, a broj izlaza određen je na temelju skupina za koje se provodi klasifikacija. Dakle, svaki neuron ulaznog sloja predstavlja jednu značajku, a izlazni neuron jednu skupinu klasifikacijskog problema. Bitno je naglasiti da su samo neuroni ulaznog sloja povezani s neuronima izlaznog sloja, dok neuroni unutar istog sloja nisu povezani.

Signal se prenosi preko ulaznog sloja, do izlaznog sloja. Ulazne vrijednosti množe s težinama i sumiraju. Težine su podesivi parametri perceptronskih mreža i predmet su učenja.

Perceptronske mreže uče se iterativnim postupkom. Prvo je potrebno na ulazni vektor  $x$  uzoraka iz skupa za učenje odrediti izlazni vektor  $y$  prema formuli  $y = xw$ .  $W$  je matrica težina, a elementi unutar nje obično nakon prve iteracije poprimaju vrijednost u intervalu između 0 i 1. U slučaju točnog izlaza, mreža se vraća na vrijednost 1. Ako je iznos pogrešan i 0, vrijednosti izlaznog vektora dodaju se pripadajućim težinama. Treća mogućnost je pogrešan iznos i vrijednost 1, te se u tom slučaju vrijednosti ulaznog vektora oduzmu od pripadajućih težina. Potom se proces učenja ponavlja za sve uzorke skupa za učenje sve dok svi nisu pravilno razvrstani u pripadajuće skupove. Dakle, cilj učenja jest podesiti nepoznate težine tako da binarni izlazni neuron koji pripada istoj skupini kao i promatrani uzorak skupa za učenje poprimi vrijednost 1, a svi ostali neuroni vrijednost 0.

Kod uspješno provedenog učenja ovakve mreže mogu se koristiti kao klasifikatori, prilikom čega se ispitivani uzorak razvrstava u skupinu čiji neuron izlaznog sloja poprimi najveću vrijednost.

Međutim, mrežom perceptora nije moguće riješiti problem ako se radi o linearno neseparabilnim uzorcima, tj. o XOR problemu. U tom slučaju potrebno je dodati skrivene slojeve, pod uvjetom da se u njima koristi nelinearna aktivacijska funkcija. Zadatak uvedenih linearnih slojeva jest rastavljanje linearno neseparabilnih uzoraka, a detaljnije o višeslojnim neuronskim mrežama slijedi u nastavku.

## 2.2 Delta pravilo

Generalizirano delta pravilo već je spomenuto i objašnjeno u dijelu 1.1.2.1., te predstavlja jednu od osnova učenja većine modela neuronskih mreža. Dakle, delta pravilo koristi se za neurone s kontinuiranim ulaznim i izlaznim vrijednostima kod kojih se kao aktivacijska funkcija ne koristi binarna funkcija praga već neka nelinearna funkcija [1].

Delta pravilo definirano je izrazom:

$$\delta_j = T_j - A_j, \quad (2.2)$$

gdje je:

- $\delta_j$  - odstupanje (delta),
- $T_j$  - željena izlazna vrijednost izlaznog neurona  $j$ ,
- $A_j$  - izlazna vrijednost izlaznog neurona  $j$ ,
- $j$  - indeks neurona izlaznog sloja

Podesive težine  $w$  dobivaju se iz izraza:

$$\Delta_{ij} = -\eta \delta_j x_i, \quad (2.3)$$

$$w_{ij}(n+1) = w_{ij}(n) + \Delta_{ij}, \quad (2.4)$$

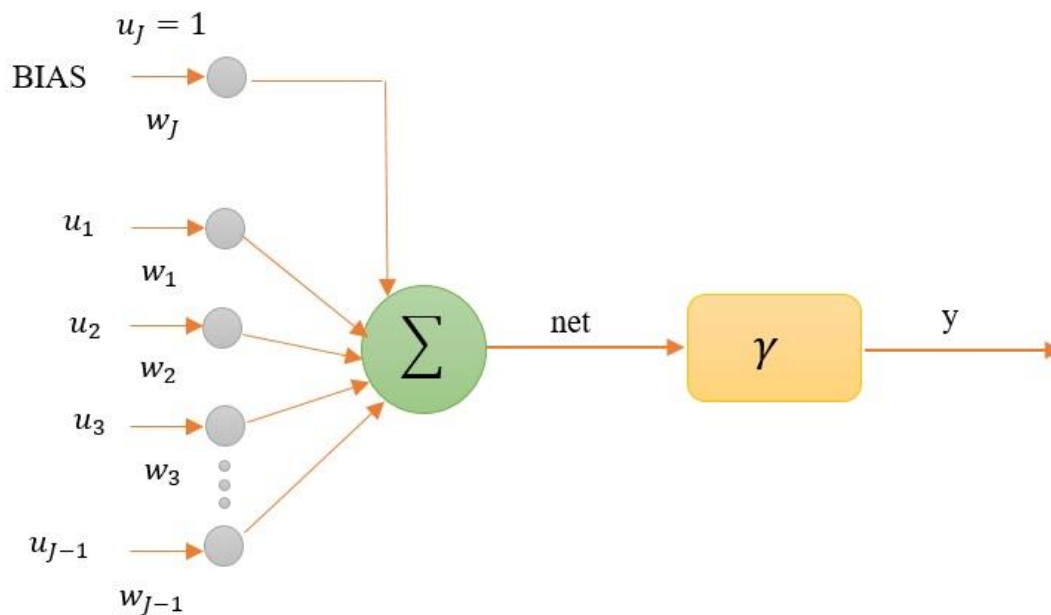
gdje je:

- $w_{ij}$  - težine izlaznog sloja,
- $\Delta_{ij}$  - faktor promjene težina,
- $\delta_j$  - odstupanje (delta),
- $\eta$  - koeficijent brzine učenja,
- $j$  - indeks neurona izlaznog sloja,
- $i$  - indeks neurona ulaznog sloja,

- $w_{ij}(n)$  - težina prije podešavanja,
- $w_{ij}(n+1)$  - težina veze između neurona  $i$  ulaznog sloja i neurona  $j$  izlaznog sloja poslije podešavanja

## 2.3 Višeslojne neuronske mreže

### 2.3.1 Statički neuron



Slika 2.1 Model statičkog neurona

Na slici 2.1 prikazane su dvije temeljne podfunkcije statičkog neurona [11, 12, 13, 14], funkcija sume  $\Sigma$  i aktivacijska funkcija  $\gamma$ . Svaki neuron koji sudjeluje u procesu učenja posjeduje poseban ulaz jedinične vrijednosti koji je u strukturi neuronske mreže realiziran vezom sa zasebnim neuronom, BIAS-om [11, 12], konstantnog iznosa u vrijednosti 1.

Funkcija sume predstavlja sumu umnožaka ulaza neurona i pripadnih težinskih faktora, što u konačnici rezultira vrijednošću  $net$ , pa vrijedi formula:

$$net = \sum_{j=1}^J w_j u_j \quad (2.5)$$

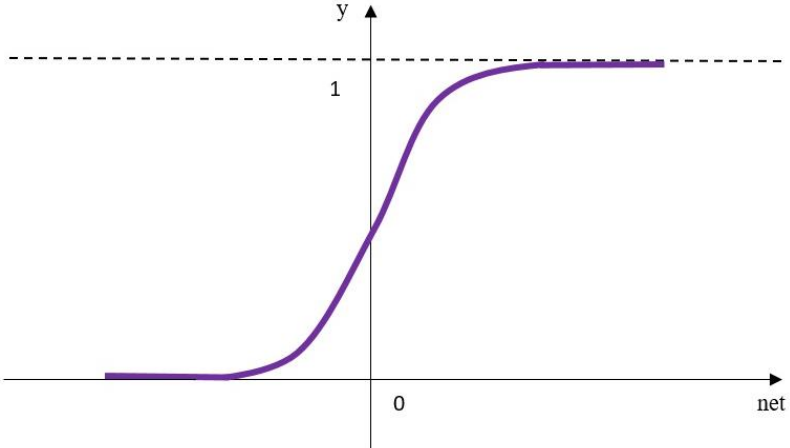
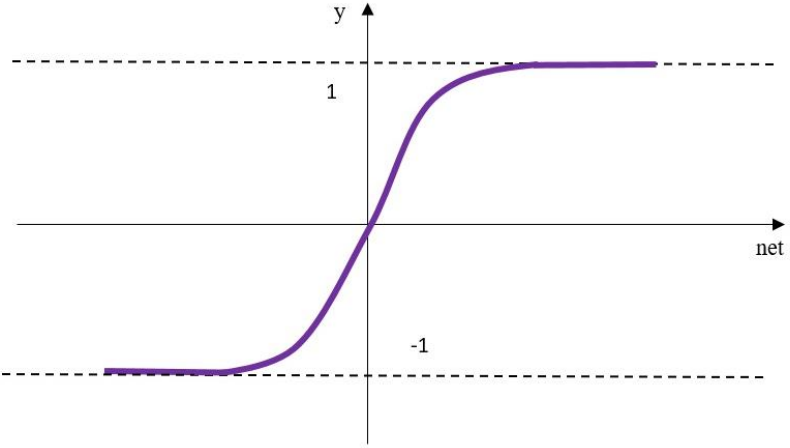
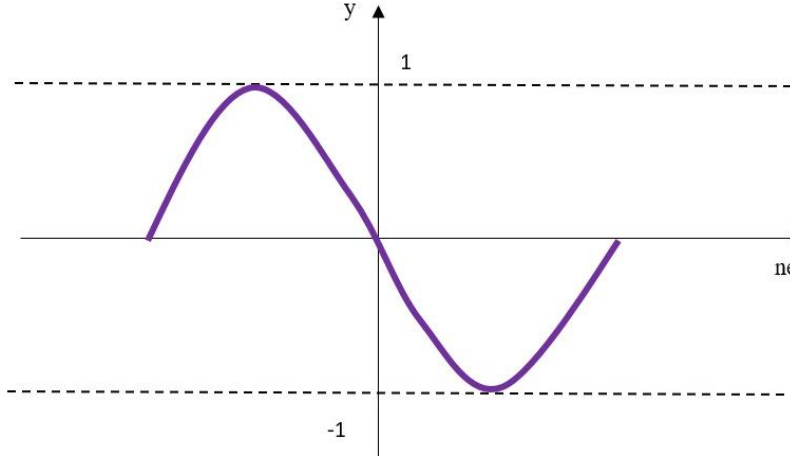
Aktivacijska funkcija služi za preslikavanje vrijednosti  $net$ -a u izlaznu vrijednost neurona  $y$ , što je opisano formulom:

$$y = \gamma(net) \quad (2.6)$$

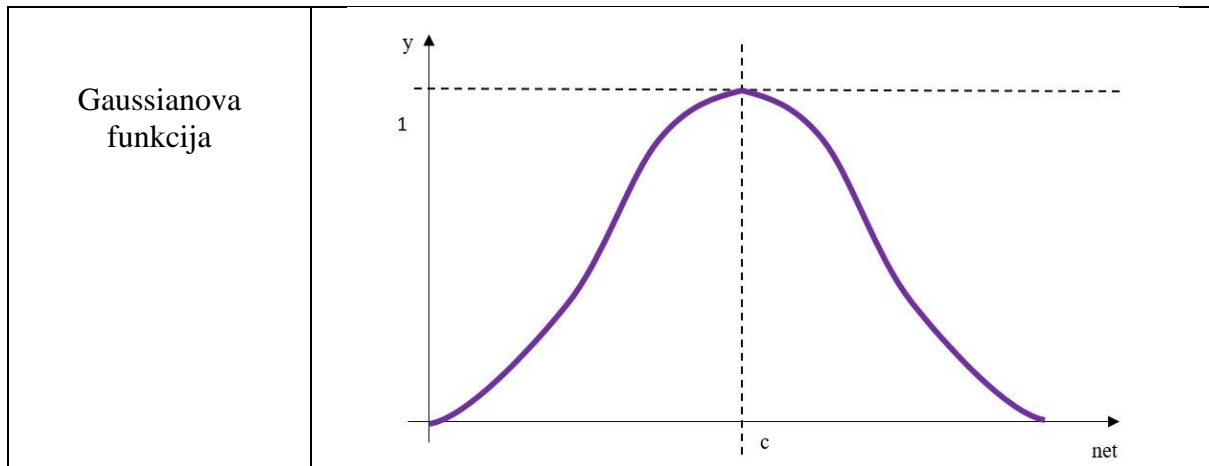
Obično se kao aktivacijske funkcije biraju monotone rastuće funkcije sa zasićenjem. Odabir aktivacijske funkcije prije svega ovisi o skupu uzoraka za učenje i o skupu uzoraka za testiranje neuronske mreže.

U sklopu ovog rada bit će prikazane četiri aktivacijske funkcije, prikazane u tablici 2.1.

Tablica 2.1 Aktivacijske funkcije korištene u diplomskom radu

Naziv aktivacijske funkcije	Grafički prikaz aktivacijske funkcije
Unipolarna sigmoidalna funkcija	
Bipolarna sigmoidalna funkcija	
Sinusna funkcija	





Kao što je vidljivo u tablici 2.1, odabirom unipolarne sigmoidalne funkcije izlazne veličine neurona ne mogu poprimiti negativnu vrijednost. Najčešće je korištena bipolarna sigmoidalna funkcija koja je globalnog karaktera te pokazuje dobre rezultate i u slučaju korištenja kod učenja dinamičkih sustava. Bipolarna sigmoidalna funkcija normirana je vrijednostima izlaza neutrona od 1 do -1. Isto normiranje vrijedi i za sinusnu funkciju, dok je Gaussianova funkcija [15] ograničena vrijednostima 0 i 1. Problem implementacije Gaussianove funkcije leži u činjenici da je potrebno poznavati širinu i centar funkcije.

U slučaju odabira unipolarne sigmoidalne funkcije, izraz (2.6) poprima oblik:

$$y = \frac{1}{1 + e^{-net}} \quad (2.7)$$

Ako odaberemo bipolarnu sigmoidalnu funkciju, za izlaz mreže vrijedi izraz:

$$y = \frac{1}{1 + e^{-net}} - 1 \quad (2.8)$$

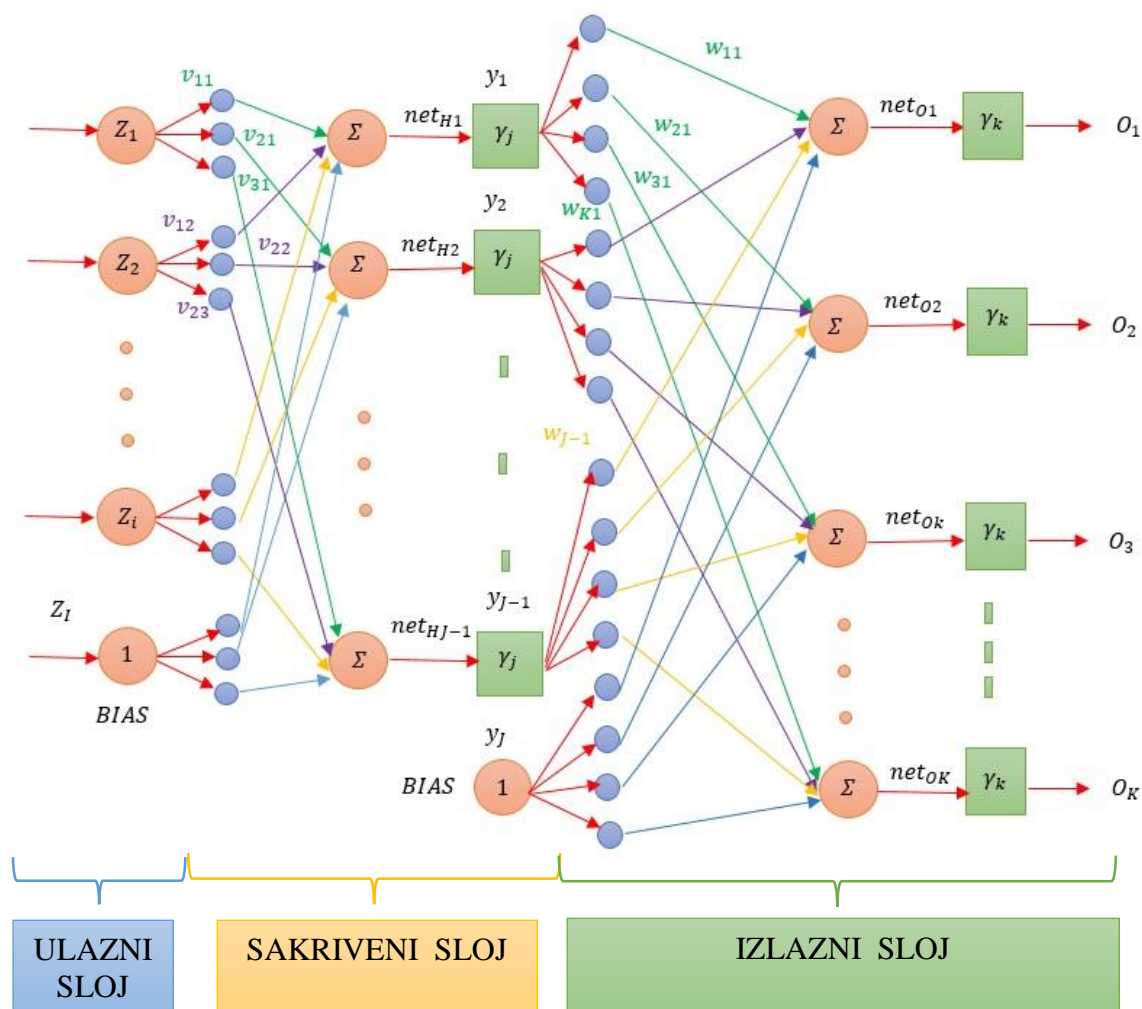
Kod sinusne funkcije vrijednost izlaza dobiva se iz:

$$y = \sin(net) \quad (2.9)$$

Vrijednost izlaza Gaussianove funkcije dobiva se iz izraza:

$$y = e^{-\frac{1}{2}\left(\frac{net-c}{\sigma}\right)^2} \quad (2.10)$$

## 2.3.2 Statička neuronska mreža



Slika 2.2 Model statičke unaprijedne neuronske mreže

$Z_i$  - neuroni ulaznog sloja, tj. ulazi u mrežu,

$i = 1, 2, \dots, I$

$v_{ji}$  - težine skrivenog sloja,

$i = 1, 2, \dots, I, j = 1, 2, \dots, J-1$

$w_{kj}$  - težine izlaznog sloja,

$k = 1, 2, \dots, K, j = 1, 2, \dots, J-1$

$O_k$  - izlaz neuronske mreže,

$k = 1, 2, \dots, K$

Struktura neuronske mreže dobiva se tako da se neurone organizira u slojeve, te se slojeve poveže vezama opterećenima težinskim faktorima. Kako je već ranije spomenuto, neuronska mreža sastoji se od najmanje triju slojeva: ulazni, izlazni te sakriveni sloj. Sakriveni sloj jedini nije u komunikaciji s okolinom, te se zato i naziva sakrivenim. Najčešće se koristi statička unaprijedna neuronska mreža čiji je model prikazan na slici 2.2.

Oznaka J predstavlja broj neurona u sakrivenom sloju uvećan za jedan, bias. Broj ulaza u mrežu predstavlja oznaka I, također uvećana za jedan, bias. Izlaz iz mreže označen je slovom K, a njegov se iznos ne povećava dodatkom biasa.

Vidljivo je da su svi slojevi mreže potpuno umreženi tj. svaki neuron promatranog sloja povezan je sa svakim neuronom prethodnog sloja. Izuzetak su jedino neuroni konstantne vrijednosti jednake jedinici, bias neuroni.

## 2.4 Učenje povratnim rasprostiranjem pogreške

U poglavlju 1.1.2 već je bilo govora o procesima učenja neuronske mreže. Kako neuronska mreža u sklopu diplomskog rada uči pomoću učitelja, potrebno je cijeli postupak detaljnije objasniti.

Kao što je već spomenuto, učenje se temelji na podešavanju težinskih koeficijenata veza među slojevima mreže, s ciljem da se izlazi mreže za odgovarajuće ulaze skupa za učenje što više približe željenim iznosima izlaza mreže. Prema tome, učenjem se nikad ne dobiva sto posto točan rezultat, već se uvijek radi o aproksimaciji nekog željenog izlaza. Kvaliteta aproksimacije ovisi o samom zadatku učenja, odabranoj topologiji mreže te algoritmu učenja mreže (vidi poglavlje 1.1.2.1). Dakle, učenje s učiteljem temelji se na iterativnom postupku adaptacije težinskih koeficijenata veza, pri čemu se mreži uzastopno prikazuju ulazne veličine i za njih odgovarajuće željene izlazne veličine.

Kod učenja statičke mreže, učenje se odvija u unaprijednoj i povratnoj fazi, a detaljnije o svakoj fazi bit će u nastavku. Što se tiče promjene parametara učenja, parametri se mogu mijenjati jednom nakon prolaska čitavog ulaznog skupa datoteke učenja kroz mrežu, a promjena se odvija nekom srednjom pogreškom proizvedenom tim uzlaznim skupom. Takva promjena parametara naziva se "batch" promjena parametara. Druga promjena parametara naziva se "pattern" ili "stochastic" promjena parametara, koja će biti korištena u sklopu ovog diplomskog rada. Kod ovog učenja, parametri učenja se mijenjaju za svaki ulazno izlazni par skupa učenja. Pri tome je jedan korak učenja jedan prolaz kroz zapisnik učenja, uz onoliko promjena vrijednosti težinskih faktora koliko parova vrijednosti ima i sam zapisnik.

### 2.4.1 Unaprijedna faza učenja

Prva faza učenja naziva se unaprijedna faza. Kod ove faze iz trening zapisnika uzimaju se vrijednosti svih ulaza mreže, te se njima izračunava izlaz mreže, pomoću težinskih faktora. Vrijednosti težinskih faktora obično se dobivaju generatorom slučajnih brojeva, u istom rasponu vrijednosti u kojem su normirani ulazi i izlazi. U slučaju neuronske mreže u ovom diplomskom radu, vrijednosti težina kreću se između 0.1 i -0.1.

## 2.4.1.1. Sakriveni sloj

$$net_{Hj} = \sum_{i=1}^I v_{ji} Z_i \quad j = 1, 2, \dots, J - 1, \quad i = 1, 2, \dots, I \quad (2.11)$$

$net_{Hj}$	-	funkcija sume neurona skrivenog sloja H
$H$	-	sakriveni sloj
$v$	-	težine sakrivenog sloja
$Z$	-	ulazni sloj
$J$	-	broj neurona u sakrivenom sloju sa biasom
$I$	-	ulazni neuroni +1

Funkcija sume  $net_{Hj}$  neurona sakrivenog sloja  $H$  dobiva se iz izraza (2.11), a prvi prvi indeks predstavlja oznaku sloja, te za svaki  $j$ -ti neuron dobiva drugi indeks.

Izlazni neuroni sakrivenih slojeva za aktivacijske funkcije korištene u ovom radu dobivaju se iz sljedećih izraza:

- Unipolarna sigmoidalna funkcija: 
$$y_j = \frac{1}{1 + e^{-net_{Hj}}} \quad (2.12)$$

- Bipolarna sigmoidalna funkcija: 
$$y_j = \frac{2}{1 + e^{-net_{Hj}}} - 1 \quad (2.13)$$

- Sinusna funkcija: 
$$y_j = \sin(net_{Hj}) \quad (2.14)$$

- Gaussianova funkcija: 
$$y_j = e^{-\frac{1}{2} \left( \frac{net_{Hj} - c_j}{\sigma_j} \right)^2} \quad (2.15)$$

Za funkcije (2.12), (2.13), (2.14) i (2.15) vrijedi  $j = 1, 2, \dots, J - 1$ , gdje je  $J$  broj neurona sakrivenog sloja uvećan za bias. Što se tiče biasa, u sva četiri slučaja vrijedi:

$$y_j = 1 \quad (2.16)$$

## 2.4.1.2. Izlazni sloj

Vrijednosti izračunatih izlaza neurona sakrivenog sloja spojene su s izlazom mreže preko težina sakrivenog sloja  $w$ . Funkcija sume  $net$  neurona izlaznog sloja  $O$  dobiva se iz izraza:

$$net_{Ok} = \sum_{j=1}^J w_{kj} y_j \quad j = 1, 2, \dots, J-1, \quad k = 1, 2, \dots, K \quad (2.17)$$

Prvi indeks sume  $net$  neurona sakrivenog sloja odnosi se na prvi indeks pripadnog izlaznog sloja, dok je drugi pripadni indeks svakom neuronu izlaznog sloja.

$Net$	-	funkcija sume neurona izlaznog sloja $O$
$O$	-	izlazni sloj
$W$	-	težine izlaznog sloja
$J$	-	broj neurona u sakrivenom sloju sa biasom
$K$	-	broj izlaznih neurona

Ako se za aktivacijsku funkciju izlaznog sloja odabere linearna funkcija, moguće je ostvariti vrijednost izlaza mreže veću od 1. Aproksimacija izraza iz mreže tada se dobiva iz izraza (2.18), gdje je  $K_p$  nagib aktivacijske funkcije. Kod izrade mreže korišteno je  $K_p = 1$ .

$$O_k = K_p net_{Ok} \quad k = 1, 2, \dots, K \quad (2.18)$$

#### 2.4.2 Povratna faza učenja

Tek nakon što je unaprijedne faze učenja završila, započinje povratna faza učenja. U ovaj fazi računa se pogreška učenja na temelju ostvarenog i željenog izlaza mreže iz unaprijedne faze učenja. Na osnovi pogreške učenja vrši se korekcija vrijednosti težinskih koeficijenata veza među slojevima. Postupak se ponavlja za svaki ulazno-izlazni par sve dok se ne postigne pogreška manja ili jednaka dozvoljenoj pogrešci koju zadaje učitelj.

Najčešće se kao suma kvadrata pogreške kao mjere odstupanja izlaza mreže od željene vrijednosti izlaza koristi sljedeća funkcija cilja:

$$E = \frac{1}{2} \sum_{n=1}^N (d_n - O_n)^2 \quad (2.19)$$

gdje je  $N$  broj elemenata u skupu za učenje. Postupak podešavanja težinskih faktora je takvo podešavanje težinskih koeficijenata veza koje minimizira funkcije cilja  $E$ .

Promjena koeficijenata težina izvršava se primjenom nekog algoritma za nelinearno optimiranje, prema formuli:

$$\vartheta(n+1) = \vartheta(n) + \Delta\vartheta(n) \quad (2.20)$$

- $\vartheta$  - težinski koeficijenti
- $n$  - trenutni korak učenja
- $\Delta\vartheta(n)$  - veličina promjene parametara učenja
- $\vartheta(n + 1)$  - nova vrijednost parametara učenja

Za izlazni sloj vrijedi jednakost  $\vartheta = w$ , a za sakriveni  $\vartheta = v$ .

Ako se suma kvadrata pogreške  $E$  uvrsti u aproksimativni izraz (2.20) i aproksimira s prva dva člana Taylorovog reda dobivamo sljedeće izraze:

$$E(\vartheta + \Delta\vartheta) \approx E(\vartheta) + \Delta E(\vartheta) \quad (2.21)$$

$$\Delta E(\vartheta) = \Delta\vartheta^T \nabla E(\vartheta) \quad (2.22)$$

$$\Delta E(\vartheta) = - \frac{\partial E(\vartheta)}{\partial \vartheta} \quad (2.23)$$

Izraz (2.23) naziva se gradijentom pogreške, a kako bi se pogreška smanjivala najvećim mogućim iznosom potrebno je odrediti  $\Delta\vartheta$  za koji promjena pogreške učenja  $\Delta E(\vartheta)$  poprima najveći mogući iznos. Potrebno je postaviti uvjet:

$$\Delta\vartheta = - \eta \nabla E(\vartheta) \quad (2.24)$$

gdje je:

$\eta$  – koeficijent brzine učenja

Koeficijent brzine učenja [11, 12] također određuje učitelj, a pri tom odabire neku vrijednost u rasponu od  $10^{-3}$  do 10. Upotrebom izraza (2.24) osigurava se promjena parametara učenja u smjeru najstrmijeg tj. najveće mogućeg pada ukupne pogreške.

Ukoliko izraz (2.24) uvrstimo u (2.20) dobiva se izraz za izračun algoritma najstrmijeg pada ili algoritam povratnog prostiranja pogreške:

$$\vartheta(n + 1) = \vartheta(n) - \eta \nabla E(\vartheta(n)) \quad (2.25)$$

Nedostatak algoritma (2.25) leži u prevelikom broju iteracija, tj. koraka učenja, što rezultira sporijim učenjem. Kako bi se proces učenja ubrzao i za istu dozvoljenu pogrešku učenja smanjio broj koraka učenja, moguća je modifikacija algoritama. Zurada [11] predlaže modifikaciju algoritama promjene parametara učenja danog izrazom (2.24), tako da se uvede zamah ili momentum tj. količina gibanja. Izraz (2.24) poprima sljedeći oblik:

$$\vartheta(n + 1) = \vartheta(n) - \eta \nabla E(\vartheta(n) + \alpha \Delta\vartheta(n - 1)) \quad (2.26)$$

gdje je:

- $n$  - trenutna promjena parametara učenja

- $(n - 1)$  - prethodna promjena parametara učenja  
 $\alpha$  - koeficijent momentuma prvog reda

Vrijednost koeficijenta momentuma određuje učitelj, a pri tome odabire neku vrijednost između 0.1 i 0.9. Uvođenjem modificirane jednadžbe (2.26) brzina algoritma se u odnosu na klasični algoritam prostiranja pogreške povećava i do 10 puta.

Dodatno povećanje brzine učenja moguće je dodatkom zamaha odnosno momentuma drugog reda. Momentum drugog reda rjeđe se koristi jer može usporiti brzinu učenja.

Koeficijent zamaha drugog reda dobiva se iz formule:

$$\beta = \frac{\alpha - 1}{3} \quad (2.27)$$

gdje je:

- $\beta$  - koeficijent momentuma drugog reda  
 $\alpha$  - koeficijent momentuma prvog reda

Umnožak koeficijenta zamaha drugog reda i greške učenja u predzadnjem koraku potrebno je oduzeti u izrazu (2.25):

$$\vartheta(n) = \alpha\Delta\vartheta(n-1) - \beta\Delta\vartheta(n-2) - \eta\nabla E(\vartheta(n)) \quad (2.28)$$

Uvrstimo li izraz (2.28) u izraz (2.25) dobivamo konačan oblik algoritma povratnog rasprostiranja pogreške:

$$\vartheta(n+1) = \vartheta(n) + \alpha\Delta\vartheta(n-1) - \beta\Delta\vartheta(n-2) - \eta\nabla E(\vartheta(n)) \quad (2.29)$$

#### 2.4.2.1 Promjena težina izlaznog sloja

Promjena parametara kod učenja mreže s povratnim rasprostiranjem pogreške odvija se od izlaza mreže prema njenom ulazu. Zato je prvo potrebno razmotriti promjenu težina izlaznog sloja.

Promjena težinskih faktora odvijat će se prema formuli (2.29), u kojoj vrijednost  $\vartheta$  zamijenimo s težinom skrivenog sloja  $w$ . Dobiva se sljedeći izraz:

$$w_{kj}(n+1) = w_{kj}(n) + \alpha\Delta w_{kj}(n-1) - \beta\Delta w_{kj}(n-2) - \eta\nabla E(n) \quad (2.30)$$

Zamijeni li se u izrazu (2.23) vrijednost  $\vartheta$  sa težinom skrivenog sloja  $w$  dobiva se izraz za izračun gradijenta pogreške za težine  $w$ :

$$\Delta E(n) = \frac{\partial E(n)}{\partial w_{kj}} \quad (2.31)$$

Gradijent pogreške izračunava se primjenom uzastopnih parcijalnih derivacija prema izrazu:

$$\frac{\partial E(n)}{\partial w_{kj}} = \frac{\partial E(n)}{\partial O_k} \cdot \frac{\partial O_k}{\partial net_{Ok}} \cdot \frac{\partial net_{Ok}}{\partial w_{kj}} \quad (2.32)$$

Prema Zuradi, karakteristična vrijednost algoritma povratnog prostiranja pogreške računa se prema izrazu:

$$\delta = - \frac{\partial E(n)}{\partial net} \quad (2.33)$$

Vrijednosti pojedinih članova izraza (2.32) dobivaju se nakon parcijalnog deriviranja te glase:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k) \quad (2.34)$$

$$\frac{\partial O_k}{\partial net_{Ok}} = \gamma'_k = 1 \quad (2.35)$$

$$\frac{\partial net_{Ok}}{\partial w_{kj}} = y_j \quad (2.36)$$

Ako izraze (2.34) i (2.35), na osnovu izraza (2.32), uvrstimo i izraz (2.33) dobivamo sljedeću vrijednost neurona izlaznog sloja:

$$\delta_{Ok} = (d_k - O_k) \quad (2.37)$$

Uvrštavanjem izraza (2.37), (2.36), (2.35) i (2.34) u izraz (2.32) dobivamo:

$$\Delta E(n) = \frac{\partial E(n)}{\partial w_{kj}} = -(d_k - O_k)y_j = -\delta_{Ok}y_j \quad (2.38)$$

Konačno, uvrštavanjem vrijednosti (2.38) u (2.30) dobiva se algoritam promjene težina izlaznog sloja:

$$w_{kj}(n+1) = w_{kj}(n) + \alpha \Delta w_{kj}(n-1) - \beta \Delta w_{kj}(n-2) + \eta \delta_{Ok}y_j \quad (2.39)$$



Vrijednost  $k$  u svim navedenim izrazima jednaka je  $k = 1, 2, \dots, K$ , gdje je  $K$  broj izlaza mreže. Također, ni vrijednost  $j$  nije se mijenjala te iznosi  $j = 1, 2, \dots, J - 1$ , gdje je  $J$  broj neurona sakrivenog sloja s biasom.

Kod svih primjera riješenih u sklopu ovog rada vrijede isti izrazi spomenuti u ovom poglavlju, osim za slučaj (2.36), u kojem se vrijednost  $y_j$  određuje s obzirom na odabranu aktivacijsku funkciju prema izrazima (2.12) -(2.15).

#### 2.4.2.2 Promjena težina sakrivenog sloja

Težine sakrivenog sloja mijenjaju se nakon što se promijene sve težine izlaznog sloja, s obzirom na to da mreža računa iz smjera izlaza prema ulazu.

$$v_{ji}(n+1) = v_{ji}(n) + \alpha \Delta v_{ji}(n-1) - \beta \Delta v_{ji}(n-2) - \eta \nabla E(n) \quad (2.40)$$

Analogno izrazu (2.30), umjesto vrijednosti  $\vartheta$  u izraz (2.29) upisujemo težinu sakrivenog sloja  $v$ , te tako dobivamo jednadžbu adaptacije težina sakrivenog sloja.

Gradijent pogreške također se izračunava primjenom uzastopnih parcijalnih derivacija, ovog puta prema izrazu:

$$\frac{\partial E(n)}{\partial v_{ji}} = \frac{\partial E(n)}{\partial y_j} \cdot \frac{\partial y_j}{\partial net_{Hj}} \cdot \frac{\partial net_{Hj}}{\partial v_{ji}} \quad (2.41)$$

Vrijednosti pojedinih članova izraza (2.41) dobivaju se deriviranjem, te nakon sređivanja dobivamo:

$$\begin{aligned} \frac{\partial E(n)}{\partial y_j} &= \frac{\partial E(n)}{\partial O_1} \cdot \frac{\partial O_1}{\partial net_{O1}} \cdot \frac{\partial net_{O1}}{\partial y_j} + \\ &+ \frac{\partial E(n)}{\partial O_2} \cdot \frac{\partial O_2}{\partial net_{O2}} \cdot \frac{\partial net_{O2}}{\partial y_j} + \\ &+ \frac{\partial E(n)}{\partial O_3} \cdot \frac{\partial O_3}{\partial net_{O3}} \cdot \frac{\partial net_{O3}}{\partial y_j} + \\ &\quad \vdots \\ &+ \frac{\partial E(n)}{\partial O_K} \cdot \frac{\partial O_K}{\partial net_{OK}} \cdot \frac{\partial net_{OK}}{\partial y_j} \end{aligned} \quad (2.42)$$

gdje je:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k) \quad k = 1, 2, \dots, K \quad (2.43)$$

$$\frac{\partial O_k}{\partial net_{OK}} = 1 \quad k = 1, 2, \dots, K \quad (2.44)$$

$$\frac{\partial net_{OK}}{\partial y_j} = Z_i \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, J - 1 \quad (2.45)$$

Parametar  $J$  i u ovom slučaju odnosi se na neurone sakrivenog sloja zajedno s vrijednošću biasa. Vrijednost  $K$  odnosi se na broj izlaza mreže.

Uvrštavanjem izraza (2.43)-(2.45) u (2.45) dobiva se:

$$\frac{\partial E(n)}{\partial y_j} = - \sum_{k=1}^K (d_k - O_k) w_{kj} \quad (2.46)$$

Uvrstimo li izraz (2.37) u (2.46) dobiva se konačan oblik rješenja prvoga člana izraza (2.41):

$$\frac{\partial E(n)}{\partial y_j} = - \sum_{k=1}^K \delta_{ok} w_{kj} \quad (2.47)$$

Treći član izraza (2.41) nakon derivacije glasi:

$$\frac{\partial net_{Hj}}{\partial v_{ji}} = Z_i \quad (2.48)$$

Za srednji član izraza (2.41) vrijedi:

$$\frac{\partial y_j}{\partial net_{Hj}} = \gamma_j' \quad (2.49)$$

Jasno je vidljivo iz izraza (2.49) da ovaj član ovisi o aktivacijskoj funkciji, pa će s toga njegov izračun za svaku aktivacijsku funkciju biti drugačiji.

- Unipolarna sigmoidalna funkcija: 
$$v_j' = \frac{e^{net_{Hj}}}{(1 + e^{net_{Hj}})^2} \quad (2.50)$$

- Bipolarna sigmoidalna funkcija: 
$$v_j' = \frac{1}{2}(1 - y_j^2) \quad (2.51)$$

- Sinusna funkcija: 
$$v_j' = \cos(net_{Hj}) \quad (2.52)$$

- Gaussianova funkcija: 
$$v_j' = -y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \quad (2.53)$$

Uvrstimo li izraze (2.50), (2.48) i (2.47) u izraz (2.40) dobivamo algoritam promjene težinskih koeficijenata sakrivenog sloja za slučaj unipolarne sigmoidalne funkcije.

$$v_{ji}(n+1) = v_{ji}(n) + \alpha \Delta v_{ji}(n-1) - \beta \Delta v_{ji}(n-2) + \eta \frac{e^{net_{Hj}}}{(1 + e^{net_{Hj}})^2} Z_i \sum_{k=1}^K \delta_{Ok} w_{kj} \quad (2.54)$$

Potom, uvrstimo li izraze (2.51), (2.48) i (2.47) u izraz (2.40) dobivamo algoritam promjene težinskih koeficijenata sakrivenog sloja za slučaj bipolarne sigmoidalne funkcije.

$$v_{ji}(n+1) = v_{ji}(n) + \alpha \Delta v_{ji}(n-1) - \beta \Delta v_{ji}(n-2) + \eta \frac{1}{2}(1 - y_j^2) Z_i \sum_{k=1}^K \delta_{Ok} w_{kj} \quad (2.55)$$

Ponovimo li postupak s izrazima (2.53), (2.48) i (2.47) dobivamo algoritam promjene težinskih koeficijenata sakrivenog sloja za slučaj sinusne funkcije.

$$v_{ji}(n+1) = v_{ji}(n) + \alpha \Delta v_{ji}(n-1) - \beta \Delta v_{ji}(n-2) + \eta \cos(net_{Hj}) \sum_{k=1}^K \delta_{Ok} w_{kj} \quad (2.56)$$

Na kraju, uvrštavanjem izraza (2.54), (2.48) i (2.47) u (2.40) ) dobivamo algoritam promjene težinskih koeficijenata sakrivenog sloja za slučaj Gaussianove funkcije.

$$v_{ji}(n+1) = v_{ji}(n) + \alpha \Delta v_{ji}(n-1) - \beta \Delta v_{ji}(n-2) - \eta y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \sum_{k=1}^K \delta_{Ok} w_{kj} \quad (2.57)$$

Kod korištenja Gaussianove funkcije u procesu učenja nije dovoljno poznavati samo algoritam promjene težinskih faktora skrivenog sloja, već i algoritam promjene centra Gaussianove funkcije te širine Gaussove funkcije.

Za početak potrebno je na identičan način pomoću parcijalnih derivacija izračunati gradijente pogreške centra i širine funkcije.

Gradijent pogreške centra Gaussianove funkcije dobiva se iz izraza:

$$\frac{\partial E(n)}{\partial c_j} = \frac{\partial E(n)}{\partial y_j} \cdot \frac{\partial y_j}{\partial c_j} \quad (2.58)$$

Prva vrijednost člana analogna je izrazu (2.42), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $c_j$ , koja glasi:

$$\frac{\partial y_j}{\partial c_j} = y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \quad (2.59)$$

Uvrste li se izrazi (2.59) i (2.42) u (2.58) dobiva se gradijent pogreške centra Gaussianove funkcije:

$$\frac{\partial E(n)}{\partial c_j} = - \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \quad (2.60)$$

Analogno izrazu (2.57) dobiva se algoritam promjene težinskih koeficijenata sakrivenog sloja centra Gaussianve funkcije:

$$c_j(n+1) = c_j(n) + \alpha \Delta c_j(n-1) - \beta \Delta c_j(n-2) + \eta \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \quad (2.61)$$

Identičan postupak primjenjuje se i u slučaju promijene širine Gaussianove funkcije, pa se gradijent promjene širine Gaussianove funkcije dobiva iz izraza:

$$\frac{\partial E(n)}{\partial \sigma_j} = \frac{\partial E(n)}{\partial y_j} \cdot \frac{\partial y_j}{\partial \sigma_j} \quad (2.62)$$

Prva vrijednost člana opet je identična izrazu (2.42), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $\sigma_j$ , te glasi:

$$\frac{\partial y_j}{\partial \sigma_j} = y_j \frac{(net_{Hj} - c_j)^2}{\sigma_j^3} \quad (2.63)$$

Uvrštavanjem izraza (2.63) i (2.42) u (2.62) dobiva se gradijent pogreške širine Gaussianove funkcije:

$$\frac{\partial E(n)}{\partial \sigma_j} = - \sum_{k=1}^K \delta_{ok} w_{kj} \cdot y_j \frac{(net_{Hj} - c_j)^2}{\sigma_j^3} \quad (2.64)$$

Konačno se uvrštavanjem (2.64), na identičan način kao i u slučaju aktivacijskih funkcija, dobiva algoritam promjene težinskih koeficijenata sakrivenog sloja širine Gaussianove funkcije:

$$\sigma_j(n+1) = \sigma_j(n) + \alpha \Delta \sigma_j(n-1) - \beta \Delta \sigma_j(n-2) + \eta \sum_{k=1}^K \delta_{ok} w_{kj} \cdot y_j \left( \frac{net_{Hj} - c_j}{\sigma_j^2} \right) \quad (2.65)$$

### 2.4.3 Točnost algoritma učenja

Nakon detaljnog objašnjenja algoritma učenja pomoću učitelja, nužno je još spomenuti par riječi u vidu točnosti algoritma. Kako bi se ocijenila točnost nekog sustava potrebno je definirati mjeru točnosti. Mjere točnosti su srednja kvadratna pogreška, korijen srednje kvadratne pogreške i normalizirani korijen srednje kvadratne pogreške.

Kao mjera točnosti u ovom radu korišten je normalizirani korijen srednje kvadratne pogreške, NRMS. Glavna prednost NRMS-a leži u njegovoj bezdimenzionalnosti koja osigurava neovisnost mjere o dimenzijama učenih veličina i koja omogućuje usporedbu izvedenih algoritama učenja s drugim algoritmima, neovisno o korištenoj sklopovskoj ili programskoj podršci.

NRMS dobivamo iz izraza:

$$NRMS = \frac{\sqrt{\frac{\sum_{n=1}^N (d_n - O_n)^2}{N}}}{\sigma d_n} \quad (2.66)$$

gdje je:

$$\sigma d_n = \sqrt{\frac{\sum_{n=1}^N (d_n - \bar{d})^2}{N}} \quad (2.67)$$

a vrijednost  $\bar{d}$  dobivena je iz izraza:

$$\bar{d} = \frac{1}{N} \sum_{n=1}^N d_n \quad (2.68)$$

NRMS možemo dobiti i preko korijena srednje kvadratne pogreške na temelju izraza:

$$NRMS = \frac{RMS}{\sigma d_n} \quad (2.69)$$

u kojem je korijen srednje kvadratne pogreške jednak iznosu brojnika izraza (2.66).

Korijena srednje kvadratne pogreške dobiva se iz :

$$RMS = \sqrt{MS} \quad (2.70)$$

gdje je MS srednja kvadratna pogreška, koja se inače računa pomoću jednadžbe:

$$MS = \sum_{n=1}^N (d_n - \bar{d})^2 \quad (2.71)$$

## 3. GRAFIČKO KORISNIČKO SUČELJE

### 3.1 Programski paket MatLab

Prva verzija MATLAB-a napisana je krajem 70.-ih na sveučilištima University of New Mexico i Stanford University [16] s ciljem primjene u matricnoj teoriji, linearnoj algebri i numeričkoj analizi. Prvobitna verzija koristi Fortran i dijelove biblioteka LINPACK i EISPACK, dok kasnije prelazi na C programski jezik. Od 1984. MATLAB je dostupan kao komercijalni proizvod tvrtke MathWorks.

Danas se MATLAB koristi za rješavanje vrlo složenih matematičkih problema, a može se koristiti i kod izračuna logaritama, modeliranja, simulacija, razvoja aplikacija i mnogih drugih aktivnosti. Zahvaljujući formi koja je slična našem zapisivanju matematičkih formula, jednim redom u MATLAB-u zamjenjujemo stotine redaka napisanih u programskim jezicima opće namjene, zbog čega ga smatramo jezikom vrlo visoke učinkovitosti tehničkog računanja.

U odnosu na druge programske pakete, MATLAB karakterizira elegancija, preglednost, praktičnost i fleksibilnost, a uz to ima i prigodnu On-line potporu. Također, složeniji programi mogu biti postavljeni u kraćem vremenu, a uz to MATLAB posjeduje i vrlo snažnu grafičku potporu koja nudi brojne mogućnosti. Zbog svih svojih pozitivnih karakteristika MATLAB nalazi primjenu u inženjerskim, privrednim i prirodnim znanostima.

### 3.2 GUI (eng. Grafic User Interface)

U sklopu izrade ovog rada za izradu korisničkog sučelja korišten je jedan od MATLAB-ovih programskih paketa, GUIDE. Svrha ovog alata jest omogućiti korisniku lakše ubacivanje željenih grafičkih objekata u korisničko sučelje.

Za potrebe rada, pomoću navedenog grafičkog korisničkog sučelja izrađena su tri glavna programa s više potprograma koji se pozivaju jedan za drugim. Vrijednost pojedinih parametara učenja koji se unose preko grafičkog sučelja sprema se u *txt* datoteke, te poziva unutar *m.file*-ova.

### 3.3 Rad sa grafičkim sučeljem

Kao što je spomenuto, zbog zadatka diplomskog rada, izrađena su tri različita programa kako bi se što bolje mogao prikazati princip rada mreže.

Svaki program sastoji se od više manjih potprograma, koji su jednaki u svakom od programa. Razlika u pojedinim programima leži u različitom broju izlaznih neurona, zbog čega mreža u svakom programu radi na drugačiji način, što će biti vidljivo iz testa mreže u nastavku.

Svaki od tri programa počinje sa radom ukoliko se pokrene *m.file* pod nazivom *pocetak\_rada*, a primjer prozora koji se potom pokrene prikazan je na slici 3.1

Princip rada i upravljanja grafičkim sučeljem prikazat će se pomoću programa izrađeng za neuronsku mrežu s dva ulaza i jednim izlazom.



Slika 3.1 Početak rada programa

U prvom prozoru navedena je struktura mreže tj. njen broj ulaza i izlaza. Moguće je odabrati želimo li u mreži koristiti nove kombinacije brojeva ili postojeće kombinacije brojeva.

Odabirom prvog izbora otvara nam se prozor prikazan na slici 3.2, u kojem je uz odabir pojedinih kombinacija brojeva od 1 do 5, tj. ulaza u mrežu, moguće pročitati opis rada programa.

Opis programa:

Odaberite brojeve koje želite množiti

Odabir prvog broja:

Odabir drugog broja:

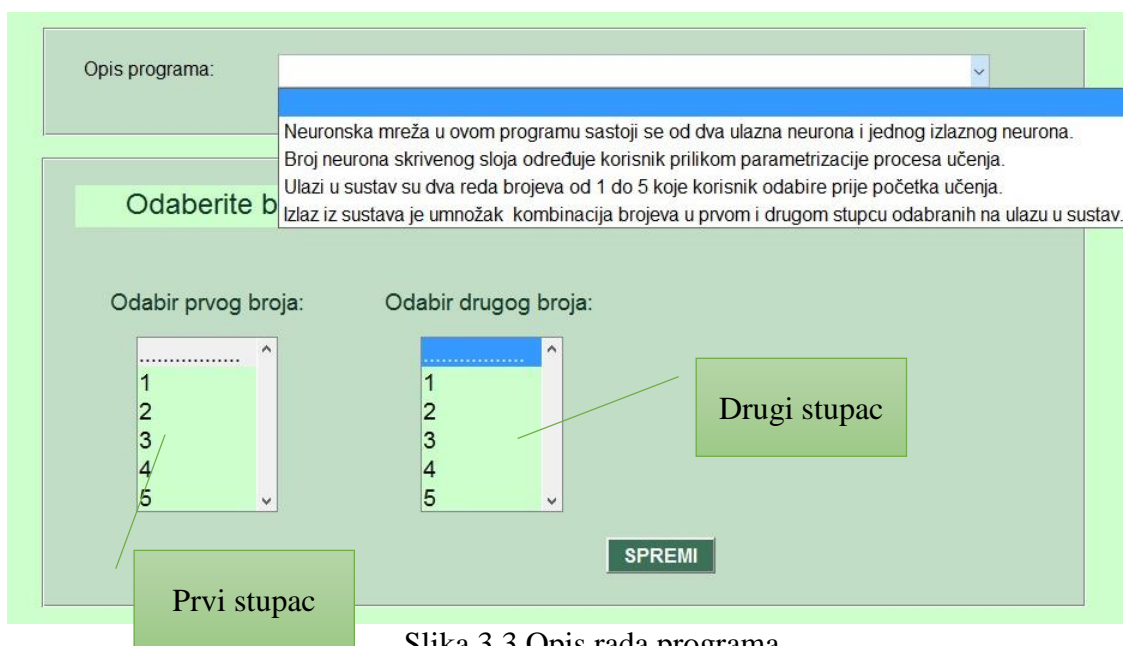
1  
2  
3  
4  
5

1  
2  
3  
4  
5

SPREMI

Slika 3.2 Izbor željenih kombinacija brojeva





Slika 3.3 Opis rada programa

Korisnik, odnosno učitelj mreže, ponajprije označi broj u prvom stupcu, potom u drugom, a potom klikne na tipku *spremi*. U tom trenutku odabrani brojevi spremaju se u *.txt* datoteku. Broj iz prvog stupca i broj iz drugog stupca zajedno tvore jednu kombinaciju. Ukupno je moguće odabrati 25 kombinacija. Postupak odabira kombinacija brojeva prikazan je u tablici 3.1.

Tablica 3.1 Odabir kombinacija za učenje mreže

<p><b>Odabir prve kombinacije</b></p>							
<p><b>Tablični prikaz spremljene prve kombinacije</b></p>	<table border="1" data-bbox="1157 1713 1324 1937"> <thead> <tr> <th></th> <th>množenik</th> <th>množitelj</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>		množenik	množitelj	1	1	1
	množenik	množitelj					
1	1	1					

<p><b>Odabir druge kombinacije</b></p>	<p>Opis programa: <input type="text"/></p> <p><b>Odaberite brojeve koje želite množiti</b></p> <p>Odabir prvog broja: <input type="list" value="1"/></p> <p>Odabir drugog broja: <input type="list" value="2"/></p> <p>Odabrane kombinacije:</p> <table border="1"> <thead> <tr> <th></th> <th>množenik</th> <th>množitelj</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p><input type="button" value="SPREMI"/> <input type="button" value="NASTAVI"/></p>		množenik	množitelj	1	1	1						
	množenik	množitelj											
1	1	1											
<p><b>Tablični prikaz spremljene druge kombinacije</b></p>	<p>Opis programa: <input type="text"/></p> <p><b>Odaberite brojeve koje želite množiti</b></p> <p>Odabir prvog broja: <input type="list" value="1"/></p> <p>Odabir drugog broja: <input type="list" value="1"/></p> <p>Odabrane kombinacije:</p> <table border="1"> <thead> <tr> <th></th> <th>množenik</th> <th>množitelj</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table> <p><input type="button" value="SPREMI"/> <input type="button" value="NASTAVI"/></p>		množenik	množitelj	1	1	1	2	1	2			
	množenik	množitelj											
1	1	1											
2	1	2											
<p><b>Odabir treće kombinacije</b></p>	<p>Opis programa: <input type="text"/></p> <p><b>Odaberite brojeve koje želite množiti</b></p> <p>Odabir prvog broja: <input type="list" value="1"/></p> <p>Odabir drugog broja: <input type="list" value="3"/></p> <p>Odabrane kombinacije:</p> <table border="1"> <thead> <tr> <th></th> <th>množenik</th> <th>množitelj</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table> <p><input type="button" value="SPREMI"/> <input type="button" value="NASTAVI"/></p>		množenik	množitelj	1	1	1	2	1	2			
	množenik	množitelj											
1	1	1											
2	1	2											
<p><b>Tablični prikaz spremljene teće kombinacije</b></p>	<p>Opis programa: <input type="text"/></p> <p><b>Odaberite brojeve koje želite množiti</b></p> <p>Odabir prvog broja: <input type="list" value="1"/></p> <p>Odabir drugog broja: <input type="list" value="1"/></p> <p>Odabrane kombinacije:</p> <table border="1"> <thead> <tr> <th></th> <th>množenik</th> <th>množitelj</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>3</td> <td>2</td> <td>3</td> </tr> </tbody> </table> <p><input type="button" value="SPREMI"/> <input type="button" value="NASTAVI"/></p>		množenik	množitelj	1	1	1	2	1	2	3	2	3
	množenik	množitelj											
1	1	1											
2	1	2											
3	2	3											

<p style="text-align: center;"><b>Odabir četvrte kombinacije</b></p>	
<p style="text-align: center;"><b>Tablični prikaz spremljene četvrte kombinacije</b></p>	

Budići da izlaz mreže treba biti umnožak odabrane kombinacije, broj iz prvog stupca uvijek je množenik, a broj iz drugog množitelj. Vidljivo je iz tablice 3.1 da se kombinacije brojeva spremaju po redoslijedu biranja, od najstarije tj. prve spremljene kombinacije, prema najnovijoj, tj. zadnjoj spremljenoj kombinaciji. Također, učenje ne može početi ako nije odabrana ni jedna kombinacija. Tipka *nastavi* zajedno s tablicom za prikaz kombinacija, pojavljuje se tek kad je spremljena prva kombinacija brojeva. Kako bi se krenulo u učenje, nije potrebno odabrati svih 25 kombinacija, ali nužno je odabrati barem jednu.

Odabrane kombinacije:		
	množenik	množitelj
1	1	1
2	1	2
3	2	3
4	2	5

Prva, najstarija kombinacija

Zadnja, najnovija kombinacija

Slika 3.4 Redoslijed spremanja odabranih kombinacija

Ukoliko se odlučimo za drugi izbor, prikazan na slici 3.1, tada ne moramo birati kombinacije brojeva, već mreža koristi prethodno spremljene kombinacije i prikazuje ih u tablici. U tom slučaju tipka za nastavak rada već je prikazana, ali nema mogućnosti nadodavanja novih kombinacija.

Opis programa:

**Odaberite brojeve koje želite množiti**

Odabir prvog broja:

Odabir drugog broja:

Odabrane kombinacije:

	množenik	množitelj
18	4	3 ^
19	4	4
20	4	5
21	5	1
22	5	2
23	5	3
24	5	4
25	5	5 v

**NASTAVI**

Slika 3.5 Izbor već spremljenih kombinacija brojeva

Klikom na tipku *nastavi*, otvara se prozor prikazan na slici 3.6, jednak i ako odaberemo postojeće kombinacije brojeva ili pak unosimo nove kombinacije.

**Definiranje parametara učenja**

Broj ulaznih neurona:

Broj izlaznih neurona:

Odabir funkcije:

Odabir težine:

Odabir momentuma:

Upišite brzinu učenja:

Upišite željeni NRMS:

Upišite broj koraka učenja:

Učenje mreže:

**UČENJE**

**ISPIS**

Slika 3.6 Definiranje parametara učenja

U prozoru prikazanom na slici 3.6 korisnik, učitelj mreže, zadaje parametre učenja mreže. Nakon definiranja svih parametara potrebno je kliknuti na tipku *učenje* kako bi mreža počela učiti prema zadanim parametrima.

Slika 3.7 Moguće opcije prilikom odabira parametara

Prilikom odabira aktivacijskih funkcija, korisnik može odabrati jednu od četiri ponuđene, a to su: bipolarna sigmoidalna, unipolarna sigmoidalna, sinusna i Gaussianova funkcija. Detalji o navedenim aktivacijskim funkcijama detaljno su opisani u prethodnom poglavlju.

Slika 3.8 Definicija parametara Gaussianove funkcije

Ako odaberemo Gaussianovu funkciju, tada je potrebno definirati centar i širinu Gaussianove funkcije. Oba parametra vezana za Gaussianovu funkciju postavljeni su na 1, a učitelj mreže može ih promijeniti u svrhu ispitivanja njihovog utjecaja na rad i izračun mreže.

Mjesta za unos vrijednosti širine i centra Gaussianove funkcije pojavljuju se kako je prikazano na slici 3.8. Ukoliko odaberemo neku drugu aktivacijsku funkciju, ovaj prozor se neće pojaviti.

Potom slijedi odabir težina. Korisnik može odabrati želi li koristiti nove težine, tj. želi li da mreža prilikom učenja kreira nove matrice  $\mathbf{V}$  i  $\mathbf{W}$  na temelju novog broja neurona skrivenog sloja, ili želi koristiti spremljen broj neurona sakrivenog sloja, tj. već spremljene težine sakrivenog i izlaznog sloja  $\mathbf{W}$  i  $\mathbf{V}$ .

Slika 3.9 Odabir proizvoljnih težina

Ukoliko korisnik mreže odabere proizvoljne težine, s desne strane pojavljuje se prikladan prozor za unos broja neurona sakrivenog sloja. U slučaju odabira postojećih težina, također se prikazuje takav prozor, samo u njemu već piše broj neurona sakrivenog sloja s kojim će mreža učiti, a mreža ga izvlači iz memorije.

Slika 3.10 Odabir postojećih težina

Dakle, broj neurona sakrivenog sloja bitan je za kreiranje matrica težina skrivenog i izlaznog sloja,  $V$  i  $W$ . Dimenzija matrice  $V$  skrivenog sloja ovisi o broju ulaznih neurona te broju skrivenih neurona. Ako dakle uzmemo 5 neurona skrivenog sloja kako je prikazano na slici 3.10, matrica  $V$  u ovom programu imat će dimenziju  $[5 \times 3]$ . Redak matrice  $V$  odgovara broju neurona skrivenog sloja, dok broj stupaca odgovara broju ulaznih neurona zajedno sa biasom. Matrica neurona izlaznog sloja ovisi o broju neurona skrivenog sloja te broju izlaznih neurona. Njena dimenzija za identičan slučaj biti će  $[6 \times 1]$ . Redak matrice  $W$  odgovara broju izlaza iz mreže, dok broj stupaca odgovara broju neurona skrivenog sloja uvećanog za bias.

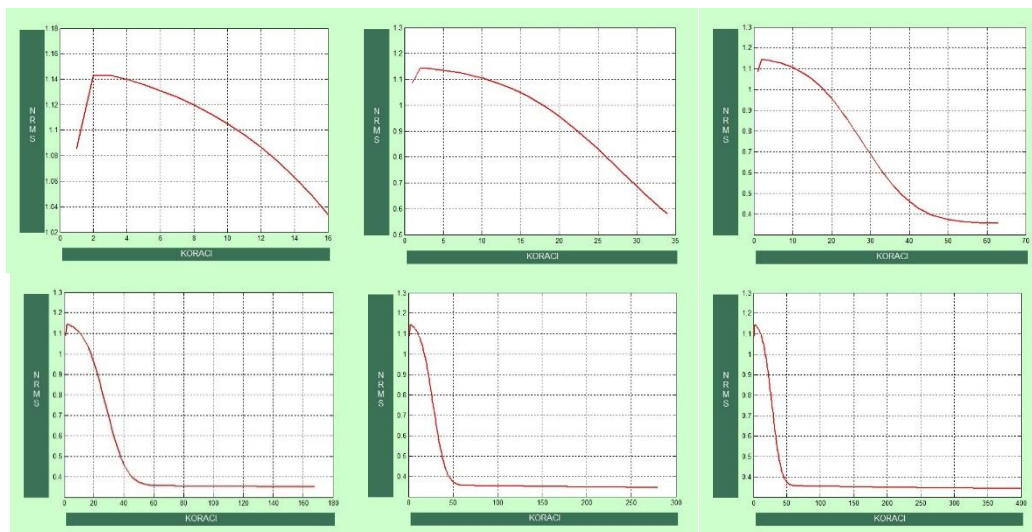
Zatim slijedi odabir momentuma, odnosno zamaha. Učitelj mreže može izabrati želi li u procesu koristiti momentum, te ako želi može odabrati između momentuma prvog i drugog reda. Ukoliko se odabere učenje bez momentuma, pojavit će se sa desne strane prozor u kojem će za vrijednost momentuma biti ispisana vrijednost 0. U slučaju odabira zamaha prvog ili drugog reda otvorit će se identičan prozor, samo ovog puta korisnik može unijeti vrijednost željenog momentuma. Kao što je već spomenuto u prethodnom poglavlju, vrijednost momentuma kreće se od 0.1 do 0.9.

Slika 3.11 Odabir momentuma

Potom slijedi ručni unos koeficijenta brzine učenja, NRMS-a te broja koraka učenja. Kao što je već spomenuto, vrijednost koeficijenta brzine učenja koju zadaje učitelj obično se kreće od  $10^{-3}$  do 10. Vrijednost normaliziranog korijena srednje kvadratne pogreške također je mali broj kojeg određuje korisnik, a karakterističan je po tome što mreža kad ostvari vrijednost zadanog NRMS-a prekida proces učenja. Broj koraka učitelj mreže određuje prema pretpostavci, "od oka". Ukoliko mreža ne ispuni zadatak u zadanom broju koraka, postoji mogućnost vraćanja na proces učenja i ponavljanja postupka uz mogućnost mijenjanja brzine učenja, NRMS-a i koraka učenja.

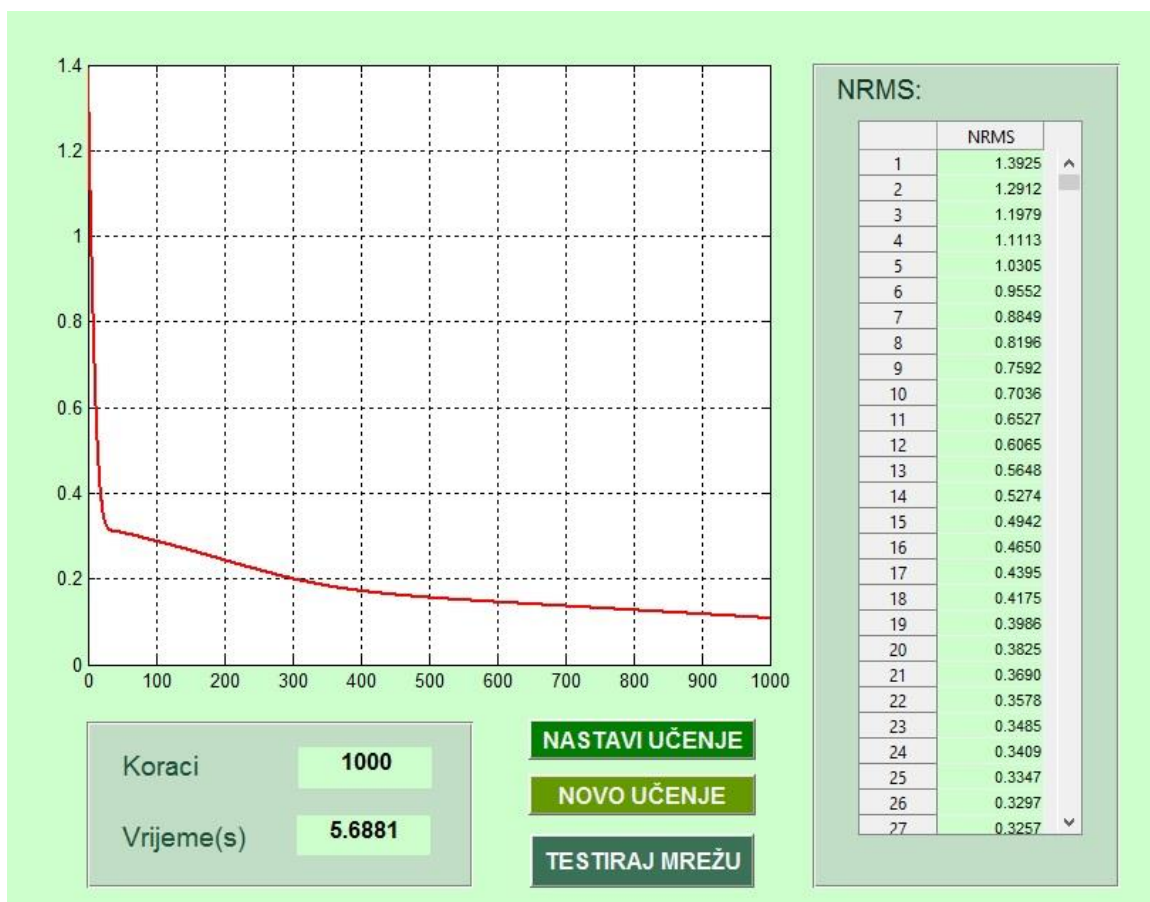
Prije učenja potrebno je još definirati želimo li direktno pratiti učenje preko grafičkog prikaza, ili želimo grafički prikaz tek kad mreža nauči. Ukoliko odaberemo učenje sa grafičkim prikazom, pojavljuje se prozor za ispis grafa prikazan na slici 3.12, u kojem vidimo promjenu

NRMS-a po koraku za vrijeme cijelog procesa učenja mreže. Tek po završetku iscrtavanja grafa možemo kliknuti na tipku *ispis*.



Slika 3.12 Grafički prikaz promijene NRMS-a u svakom koraku učenja

U slučaju da ne želimo grafički prikaz, nakon pritiska tipke *učenje*, možemo odmah pritisnuti *ispis*. Tada nam se grafički prikaz pojavljuje tek na kraju učenja mreže, opet u posebnom prozoru.



Slika 3.13 Grafički prikaz ovisnosti NRMS-a o koraku nakon učenja



Dakle, u ispisu dobivamo grafički prikaz ovisnosti NRMS-a o koraku učenja. Također, na slici 3.13 vidljivo je da uz grafički prikaz postoji i tablični prikaz vrijednosti NRMS-a u svakom koraku. U lijevom donjem uglu nalazi se podsjetnik na broj koraka učenja te vrijeme potrebno da mreža prođe zadane korake.

Sad možemo birati između tri mogućnosti, a to su: nastavak učenja, novo učenje i testiranje. Ukoliko kliknemo na *nastavi učenje* mreža se vraća na definiranje parametara učenja, ali ovog puta može se mijenjati samo brzina učenja, NRMS, korake učenja te na grafički prikaz. Dakle, dobivamo prozor prikazan na slici 3.14.

Slika 3.14 Nastavak učenja mreže

Ukoliko se odlučimo za *novo učenje*, vraćamo se na početak rada, slika 3.1, te možemo opet odabrati želimo li koristiti nove kombinacije ili stare kombinacije brojeva.

Ako smo zadovoljni učenjem i želimo mrežu testirati, kliknemo na *testiraj mrežu*. Otvara nam se prozor prikazan na slici 3.15.

U desnom uglu nalazi se tablica, podsjetnik na odabrane kombinacije brojeva koje smo zadali mreži na početku rada. Testiranje možemo provesti na dva načina. Prvi je da odaberemo najprije broj u prvom stupcu, pa broj u drugom stupcu, te potom kliknemo na tipku *test*. Pritiskom na tipku *test*, dobivamo ispis vrijednosti odziva mreže, stvarne vrijednosti i točne tj. vrijednosti koju mreža treba dobiti.

Druga opcija je ručni unos brojeva za testiranje. Potrebno je kliknuti najprije u polje za upis prvog broja, zatim u polje za upis drugog broja, te opet na tipku *test*. Kod ovog upisa možemo testirati rad mreže i za slučaj brojeva većih od 5.

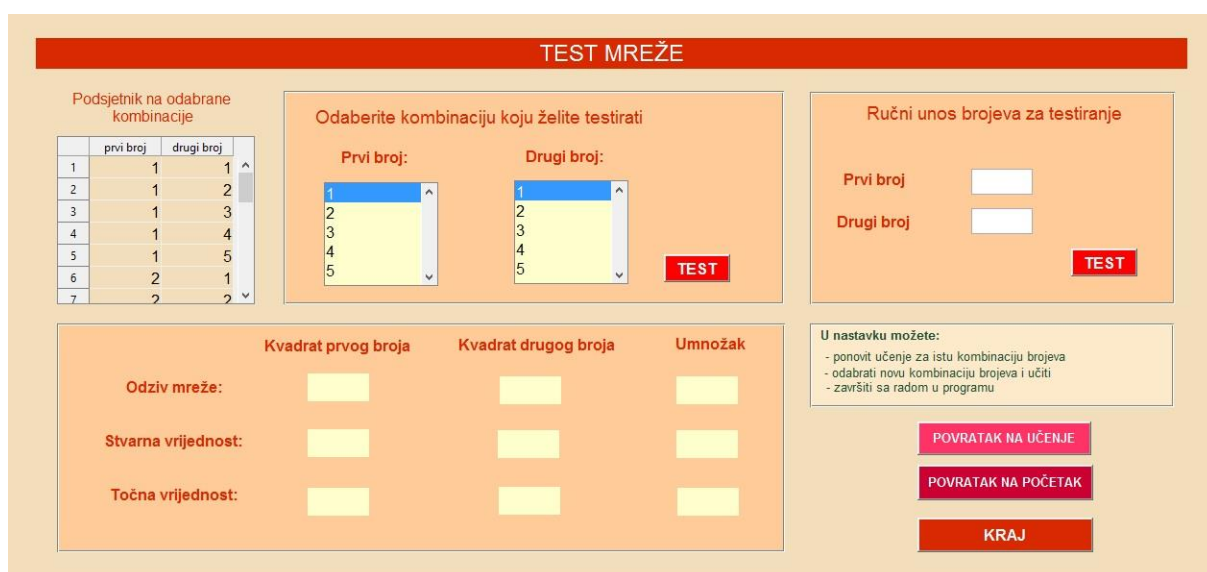
Nakon što provedemo testiranje možemo se vratiti na početak, vratiti na učenje ili završiti s radom programa. Ukoliko kliknemo *vрати na učenje*, otvara nam se prozor prikazan na slici 3.6, što znači mogućnost ponovnog definiranja parametara učenja. Klikom tipke *vрати na početak*,

vraćamo se na prozor prikazan na slici 3.1, početak rada mreže. Nakon toga možemo ponovno provesti učenje po potrebi sa novim kombinacijama. Ukoliko želimo završiti sa radom mreže kliknemo na tipku *kraj*.

Budući da se prozori testiranja za svaki program razlikuju, bit će slikovno prikazani u nastavku.



Slika 3.15 Test mreže s jednim izlazom



Slika 3.16 Test mreže s tri izlaza

Mreža prikazana na slici 3.16 kao jedan izlaz prikazivat će umnožak kombinacije brojeva, kao i mreža na slici 3.15. Druga dva izlaza bit će kvadrati brojeva koji množeni daju taj umnožak, tj. kvadrat prvog odabranog broja i kvadrat drugog odabranog broja. Mreža s četiri izlaza, prikazana na slici 3.17, uz navedene izlaze, kao četvrti izlaz prikazuje sumu svake kombinacije brojeva.

**TEST MREŽE**

Podsjetnik na odabrane kombinacije

	prvi broj	drugi broj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2

Odaberite kombinaciju koju želite testirati

Prvi broj:  Drugi broj:

**TEST**

Ručni unos brojeva za testiranje

Prvi broj:

Drugi broj:

**TEST**

U nastavku možete:

- ponoviti učenje za istu kombinaciju brojeva
- odabrati novu kombinaciju brojeva i učiti
- završiti sa radom u programu

Povratak na učenje

Povratak na početak

KRAJ

Odziv mreže:

Stvarna vrijednost:

Točna vrijednost:

Kvadrat prvog broja:

Kvadrat drugog broja:

Suma:

Umnožak:

Slika 3.17 Test mreže s četiri izlaza

### 3.4 Primjer rada mreže

Način rada mreže prikazati ću na mreži s dva ulaza i tri izlaza.



Slika 3.18 Početak rada mreže s dva ulaza i tri izlaza

Odabrat ću postojeće kombinacije brojeva, njih 25, spremljenih u prijašnjem procesu učenja. Budući da se javlja prozor identičan slici 3.5, samo ovog puta u nijansama crvene, nama ga potrebe slikovno prikazivati. Klikom na *nastavi*, otvara se prozor za definiciju parametara učenja. Odabrat ću unipolarnu sigmoidalnu funkciju. Težine ću postaviti na proizvoljne i

upisati 3 neurona skrivenog sloja. Potom ću odabrati momentum prvog reda, te postaviti njegovu vrijednost na 0.7. Brzinu učenja postaviti ću na 0.1, dok ću vrijednost NRMS-a postaviti na 0.01. Za početak korake učenja postaviti ću na 2000 i neću grafički pratiti proces učenja.

**Definiranje parametara učenja**

Odabir funkcije: Unipolarna sigmoidalna

Odabir težine: Proizvoljne

Odabir momentuma: Prvog reda

Broj ulaznih neurona: 2

Broj izlaznih neurona: 3

Broj skrivenih neurona: 3

Iznos momentuma: 0.7

Upišite brzinu učenja: 0.1

Upišite željeni NRMS: 0.01

Upišite broj koraka učenja: 2000

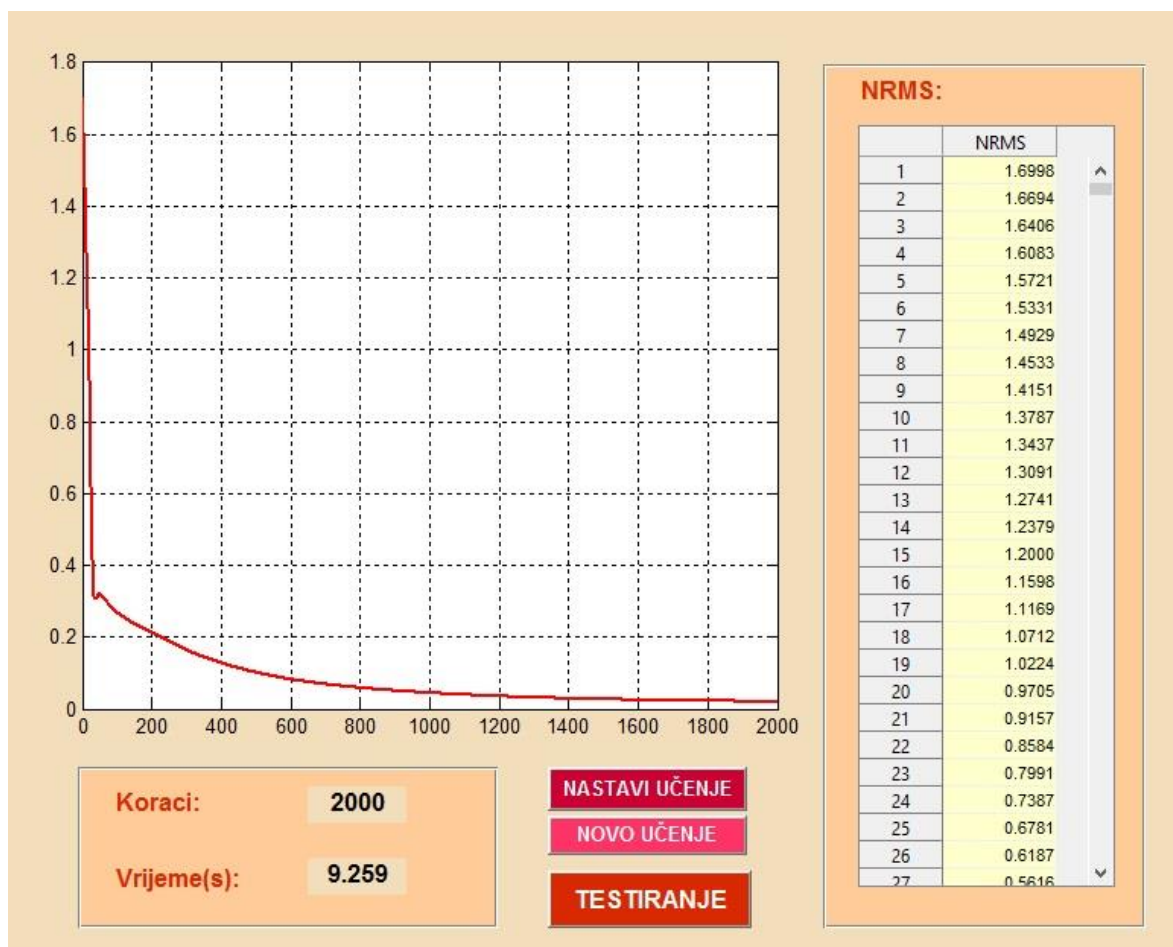
Učenje mreže: Bez grafičkog prikaza

**UČENJE**

**ISPIS**

Slika 3.19 Primjer definiranja parametara

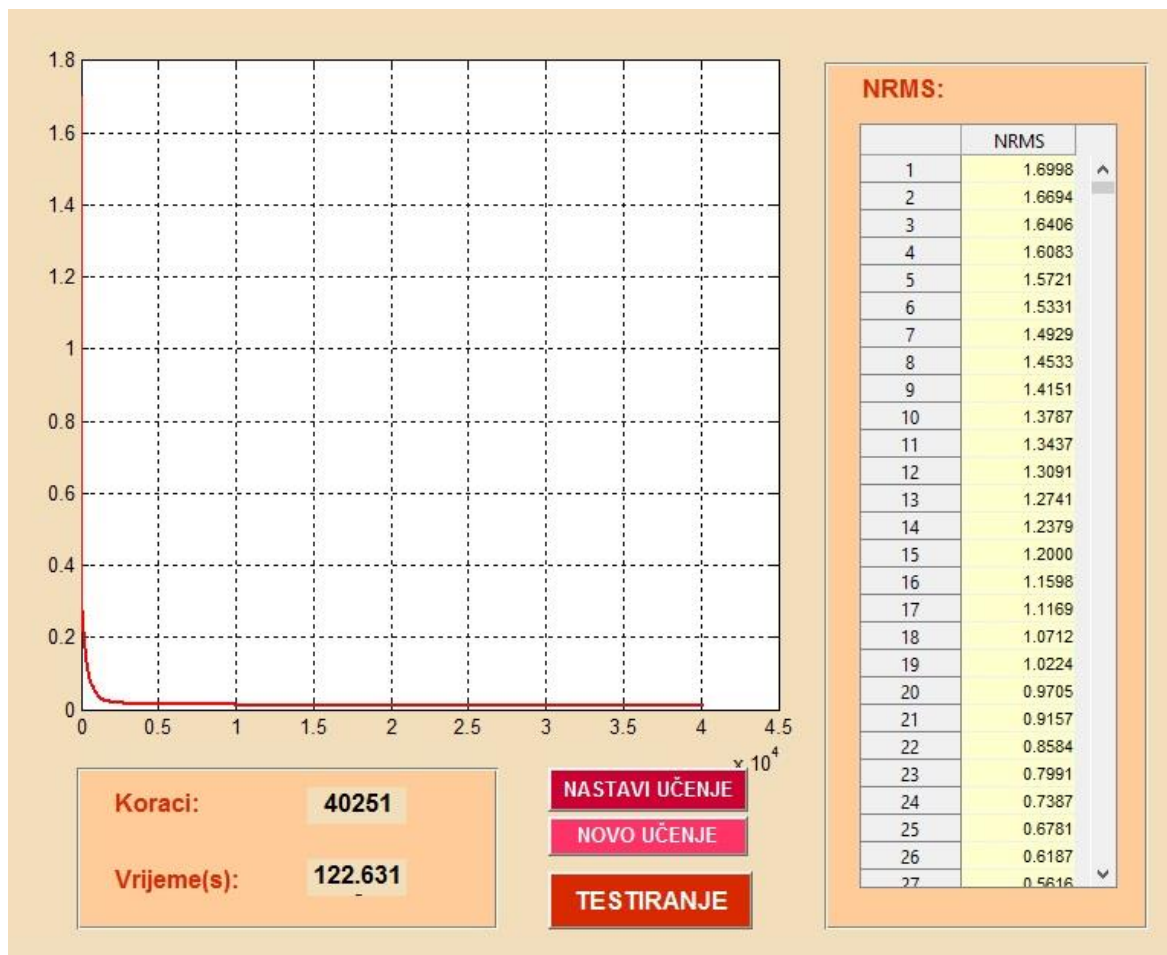
Iz ispisa na slici 3.20 vidi se da mreža nije uspjela naučiti, te da je potrebno povećati broj koraka kako bi u tome uspjela. Kako bi prošla 2000 koraka, mreži je bilo potrebno 7.87 sekundi.



Slika 3.20 Grafički prikaz promjene NRMS-a za 2000 koraka

Dakle, kliknut ću na *nastavi učenje*.

Budući da se radi o jako puno kombinacija, mreži će trebati jako puno koraka kako bi uspjela naučiti. Prema tome u polje za korake upisujem 40000 koraka. Druge parametre neću mijenjati.



Slika 3.21 Grafički prikaz promjene NRMS-a za 42000 koraka

Dakle, za prvobitno zadanih 2000, te u drugom pokušaju zadanih 40000 koraka, dakle ukupno 42000 koraka, mreža je uspjela naučiti za 40251 koraka svih 25 kombinacija. Pri tome, za izračun bilo joj je potrebno 122,631 sekunda. Dakle, vidljivo je da mreža kako bi naučila 25 kombinacija treba velik broj koraka učenja. No, da li je mreža dobro naučila, treba provjeriti u testu. U tablici 3.2 prikazano je nekoliko kombinacija te njihovi rezultati.

Tablica 3.2 Prikaz rezultata učenja mreže nakon 38418 koraka

Prvi broj	Drugi broj		Kvadrat prvog broja	Kvadrat drugog broja	Umnožak
1	2	Odziv mreže	1	4	2
		Stvarna vrijednost	0,93972	4,0575	2,0138
		Točna vrijednost	1	4	2
		Odziv mreže	9	25	15

3	5	Stvarna vrijednost	8,9513	25,0098	14,9264
		Točna vrijednost	9	25	15
5	2	Odziv mreže	25	4	10
		Stvarna vrijednost	24,9875	4,115	9,9788
		Točna vrijednost	25	4	10
7	4	Odziv mreže	43	16	30
		Stvarna vrijednost	43,4396	16,4244	29,8663
		Točna vrijednost	49	16	28
6	8	Odziv mreže	35	51	45
		Stvarna vrijednost	34,7172	50,9809	44,48
		Točna vrijednost	36	64	48

**TEST MREŽE**

**Podsjetnik na odabrane kombinacije**

	prvi broj	drugi broj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2

**Odaberite kombinaciju koju želite testirati**

Prvi broj:

1

2

3

4

5

Drugi broj:

1

2

3

4

5

**TEST**

**Ručni unos brojeva za testiranje**

Prvi broj:

Drugi broj:

**TEST**

	Kvadrat prvog broja	Kvadrat drugog broja	Umnožak
<b>Odziv mreže:</b>	1	4	2
<b>Stvarna vrijednost:</b>	0.93972	4.0575	2.0138
<b>Točna vrijednost:</b>	1	4	2

**U nastavku možete:**

- ponoviti učenje za istu kombinaciju brojeva
- odabrati novu kombinaciju brojeva i učiti
- završiti sa radom u programu

**POVRATAK NA UČENJE**

**POVRATAK NA POČETAK**

**KRAJ**

Slika 3.22 Testiranje mreže nakon 40251 koraka

Vidljivo je da je na temelju tablice 3.2 da je mreža uspjela točno naučiti rezultate željenih kombinacija brojeva. Također, mreža je testirana i za brojeve veće od 5, a iz rezultata je vidljivo da su iznosi i kod tih slučajeva blizu točnim vrijednostima. Međutim, kako se brojevi povećavaju u odnosu na naučene, tako mreža daje sve lošiji rezultat. U nastavku će biti testirano koliko brojeva mogu izbaciti iz procesa učenja, a da mreža daje dobar rezultat. Također, prikazat će utjecaj koeficijenta brzine učenja, NRMS-a, broja neurona i momentuma na sam proces učenja, kao i primjenu pojedinih aktivacijskih funkcija.

## 4. ISPITIVANJE RADA JEDNOSLOJNE NEURONSKE MREŽE

U nastavku slijedi ispitivanje pojedinih parametara koji utječu na proces učenja mreže. Utjecaj pojedinih parametara biti će najprije prikazan na mreži s dva ulaza i jednim izlazom, potom na mreži za dva ulaza i tri izlaza te za kraj na mreži s dva ulaza i četiri izlaza.

### 4.1 Ispitivanje mreže s dva ulaza i jednim izlazom

#### 4.1.1 Broj neurona skrivenog sloja

Kako broj neurona u skrivenom sloju utječe na proces učenja mreže, najprije će prikazati pomoću mreže s dva ulazna i jednim izlaznom slojem. Parametri zadani u procesu učenja prikazani su u tablici 4.1.

Tablica 4.1 Parametri zadani u procesu učenja

Neuronska mreža s dva ulazna sloja i jednim izlaznim slojem	
Aktivacijska funkcija:	Bipolarna sigmoidalna
Broj neurona skrivenog sloja:	3, 5, 7
Iznos momentuma:	0
Brzina učenja:	0.1
Željeni NRMS:	0.03

Tablica 4.2 Vrijednost koraka učenja i vrijeme izračuna za parametre zadane u tablici 4.1

	3 neurona skrivenog sloja	5 neurona skrivenog sloja	7 neurona skrivenog sloja
Broj koraka učenja:	1 634	3 450	3 394
Vrijeme izračuna (s):	6,7203	12,6085	13,5147

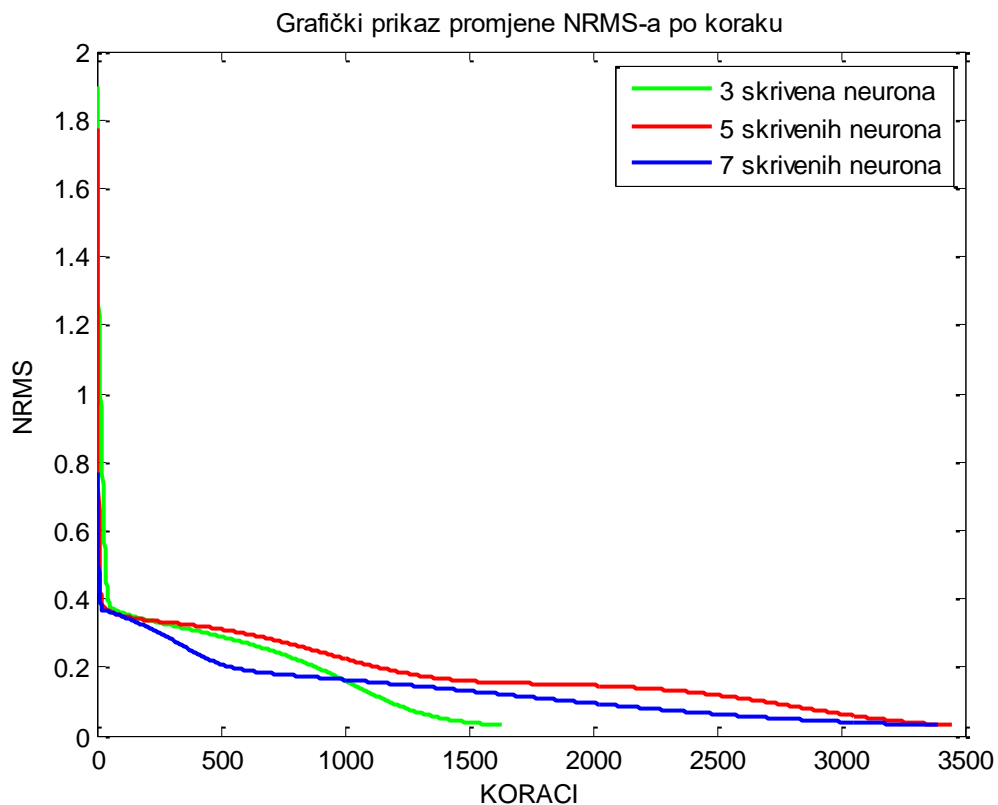
Iz tablice 4.2 vidi se da porast broja neurona skrivenog sloja ne mora nužno utjecati na korake učenja mreže. Tako je za 5 neurona skrivenog sloja bilo potrebno proći 3450 kako bi mreža naučila, dok je za 7 neurona skrivenog sloja bilo potrebno 3394 koraka. Budući da sustav matrice težina  $\mathbf{W}$  i  $\mathbf{V}$  slaže proizvoljno u rasponu od 0.1 do -0.1, dimenzija matrica ovisno o broju ulaznih, izlaznih i neurona skrivenog sloja, vrijednosti  $\mathbf{W}$  i  $\mathbf{V}$  se stalno mijenjaju, pa tako može doći do smanjenja ili povećanja koraka učenja. Kao dokaz te tvrdnje testirala sam mrežu s istim parametrima prikazanim u tablici 4.1, za 3 neurona skrivenog sloja, s različitim vrijednostima matrica  $\mathbf{W}$  i  $\mathbf{V}$ .

Tablica 4.3 Utjecaj težina na korake i vrijeme učenja

	KORACI	VRIJEME
Prvi put	2324	10,657
Drugi put	9612	31,3281
Treći put	1310	6,1614

Dakle, iz tablice 4.3 vidljivo je da matrice težina, iako je zadan isti broj neurona, ukoliko se one generiraju prilikom svakog unosa neurona, mogu imati utjecaj na korake i vrijeme izračuna. Sve ovisi o vrijednostima elemenata unutar matrica  $\mathbf{W}$  i  $\mathbf{V}$ . Tako se kod prvog i trećeg učenja za isti broj neurona skrivenog sloja, ali različite matrice  $\mathbf{W}$  i  $\mathbf{V}$ , koraci učenja bili manji u odnosu na drugo testiranje.

Što se tiče vremena učenja, ono je uvijek različito, ali u pravilu učenje s većim brojem koraka traje duže od učenja s manjim brojem koraka.



Slika 4.1 Grafički prikaz promjene NRMS-a po koraku za različite vrijednosti matrica težina  $\mathbf{W}$  i  $\mathbf{V}$

U nastavku učenja koristiti ću 5 neurona skrivenog sloja i uvijek iste matrice težina  $\mathbf{W}$  i  $\mathbf{V}$ :

$$\mathbf{V} = \begin{bmatrix} -0.7268937253 & 0.5581034465 & 0.3974916647 \\ 0.4424549972 & 0.4300741568 & -0.6043803466 \\ -0.7864762768 & 0.8074411211 & -0.9389181074 \\ 0.3075146973 & 0.7818450087 & 0.4881485207 \\ -0.0116521267 & -0.3316738945 & 0.0000448712 \end{bmatrix} \quad (4.1)$$



$$\mathbf{W} = \begin{bmatrix} 0.8265736553 \\ 0.5923677472 \\ -0.8025754427 \\ -0.4762576323 \\ -0.3292863201 \\ 0.3594559028 \end{bmatrix}^T \quad (4.2)$$

Što se tiče konačnog rezultata učenja, u testiranju mreže vidljivo je da mreža radi jednako dobro za 3, 5 i 7 neurona, s malim razlikama u decimalnim brojevima. Prema tome, može se reći da neuroni skrivenih slojeva utječu jedino na težine  $\mathbf{W}$  i  $\mathbf{V}$ , koje ovisno o vrijednostima koje poprime djeluju na korake učenja. Iako su koraci učenja različiti, manji broj koraka daje zadovoljavajuće rezultate kao veći broj koraka. Na nekoliko proizvoljnih brojeva u tablici 4.4 prikazani su rezultati testiranja koji potvrđuju tvrdnju. Početni parametri postavljeni prije testiranja jednaki su onima prikazanim u tablici 4.1 dok su koraci i vrijeme učenja prikazani tablicom 4.2.

Tablica 4.4. Stvarna vrijednost odziva mreže

Prvi broj	Drugi broj	3 neurona skrivenog sloja	5 neurona skrivenog sloja	7 neurona skrivenog sloja
1	3	3,0482	2,9847	2,7153
2	5	9,9328	10,0038	10,0186
4	2	8,1111	7,8903	8,0246
5	4	20,2342	19,8226	20,1987
6	6	33,89647	36,4327	33,66

Vidljivo je iz tablice 4.4 da neovisno o broju neurona skrivenog sloja mreža nije uspjela točno izračunati umnožak  $6 \times 6$ . Šest se ne nalazi kao jedan od mogućih ulaza u mrežu, tako da je i pretpostavljeno da će mreža pogriješiti budući da nije naučila množiti do šest već do pet.

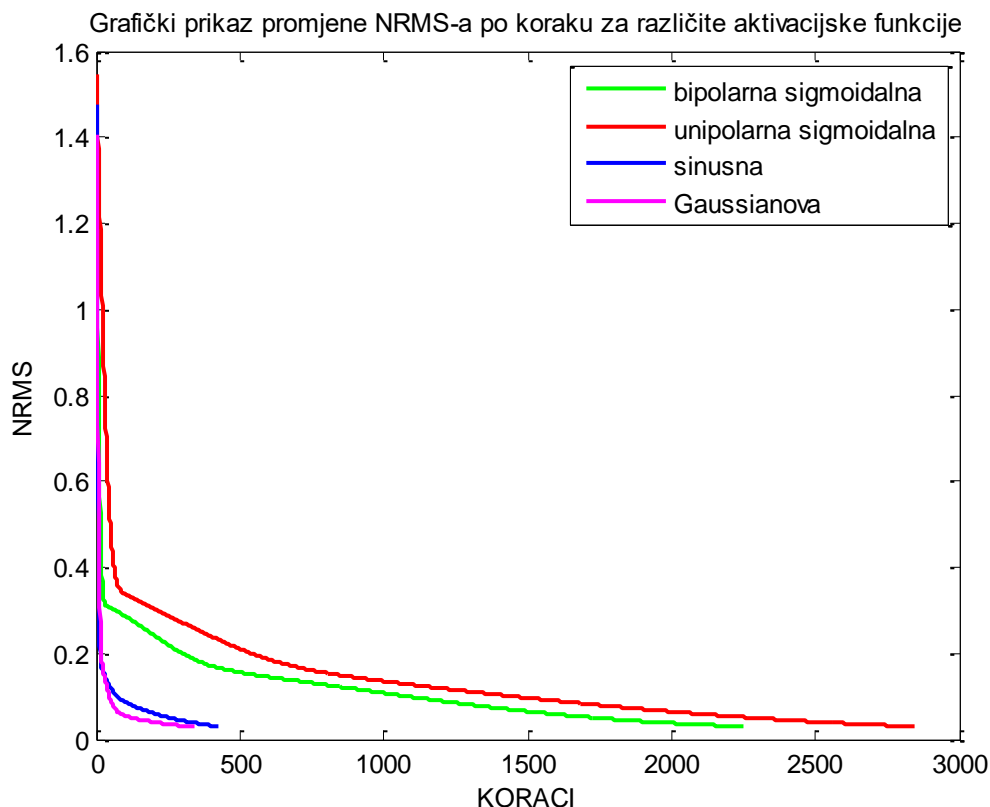
#### 4.1.2 Aktivacijska funkcija

U nastavku slijedi ispitivanje pojedinih aktivacijskih funkcija u neuronskoj mreži s dva ulaza i jednim izlazom. Prilikom zadavanja učenja koriste se parametri prikazani u tablici 4.5. Za sve aktivacijske funkcije korištene su iste matrice težina  $\mathbf{W}$  i  $\mathbf{V}$  prikazane na (4.1) i (4.2).

Tablica 4.5. Zadani parametri prilikom ispitivanja aktivacijskih funkcija

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona skrivenog sloja:	5

Prema rezultatima ispitivanja s jednakim matricama  $\mathbf{V}$  i  $\mathbf{W}$  ispada da bez momentuma i za brzinu učenja od 0.1 najpogodnija je Gaussianova funkcija, što se tiče brzine učenja. Potom slijedi sinusna funkcija, nakon nje bipolarna te na kraju unipolarna sigmoidalna funkcija.



Slika 4.2 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija i 5 neurona skrivenog sloja bez momentuma

Tablica 4.6. Konačni koraci i vrijeme prilikom ispitivanja aktivacijskih funkcija

	Bipolarna sigmoidalna funkcija	Unipolarna sigmoidalna funkcija	Sinusna funkcija	Gussianova funkcija
Koraci	2251	2843	427	344
Vrijeme(s)	11,4869	11,9738	3,4485	2,9623

Dakle, vidljivo je iz tablice 4.6 da je najpogodnija Gaussianova funkcija. Prilikom računanja parametri Gaussianove funkcije, širina i centar, postavljeni su na 1. Ovakvi rezultati za slučaj prikazan u tablici 4.5, kad je momentum postavljen na 0. Rezultati testiranja za iste vrijednosti brojeva koje su prikazane u tablici 4.4, za broj koraka prikazan u tablici 4.6, prikazani su u tablici 4.7.

Tablica 4.7 Stvarni odziv mreže za različite aktivacijske funkcije i jednake vrijednosti težina

Prvi broj	Drugi broj	Bipolarna sigmoidalna funkcija	Unipolarna sigmoidalna funkcija	Sinusna funkcija	Gaussianova funkcija
1	3	2,7165	2,7196	2,5864	2,9107
2	5	9,9924	9,8285	9,9882	9,969
4	2	8,0631	8,0576	8,1846	7,82
5	4	20,1884	20,1385	20,1223	20,2949
6	6	33,622	33,3747	34,7479	33,3528

Iz tablice 4.7 vidljivo je da mreža daje dobre rezultate, neovisno o tome o kojoj se aktivacijskoj funkciji radi. Iako je Gaussianva funkcija najbrža tj. uspjela je naučiti svih 25 kombinacija brojeva u najmanje koraka, njezini rezultati nisu toliko različiti u odnosu na unipolarnu sigmoidalnu funkciju, kojoj je za učenje bilo potrebno najviše koraka i vremena.

#### 4.1.3 Normalizirani korijen srednje kvadratne pogreške

Slijedi provjera rezultata testiranja za iste kombinacije brojeva prikazane u tablici 4.7., ali ovog puta za vrijednost NRMS-a 0.01. Također, korištene su iste težine  $\mathbf{W}$  i  $\mathbf{V}$ , 5 neurona skrivenog sloja, te brzina učenja 0.1. Momentum je i dalje postavljen na 0.

Tablica 4.8 Zadani parametri prilikom ispitivanja NRMS-a

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,01
Broj neurona skrivenog sloja:	5

Tablica 4.9 Prikaz rezultata testiranja za vrijednosti NRMS -a 0.01 i 0.03

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		8452	2251	16327	2843	1209	427	2370	344
Vrijeme (s):		29,3687	11,4869	77,1753	11,9738	4,8292	3,4485	17,8398	2,9623
Prvi broj:	Drugi broj:	nrms 0,01	nrms 0,03	nrms 0,01	nrms 0,03	nrms 0,01	nrms 0,03	nrms 0,01	nrms 0,03
1	3	3,0114	2,7165	3,0175	2,7196	2,9299	2,5864	3,0183	2,9107
2	5	9,9655	9,9924	9,9161	9,8285	9,9735	9,9882	9,9587	9,969
4	2	7,9164	8,0631	7,8977	8,0576	8,0003	8,1846	7,8516	7,82
5	4	20,0967	20,1884	20,0588	20,1385	20,114	20,122	20,0807	20,295
6	6	34,7539	33,622	34,5546	33,3747	35,038	34,748	34,701	33,353

Vidljivo je iz tablice 4.9 da se povećanjem NRMS -a broj koraka učenja smanjio, jer mreži smo dozvolili manji iznos pogreške. To je vidljivo iz rezultata testiranja jer se rezultati stvarnog odziva mreže u slučaju kad NRMS iznosi 0.01 kreću bliže točnom riješenju. Također, prema ovom testiranju, najbrža je sinusna funkcija, a unipolarna sigmoidalna funkcija i dalje je najsporija, tj. potrebno joj je najviše vremena i koraka da bi naučila svih 25 umnožaka mogućih kombinacija brojeva.

Dakle, vrijednost normaliziranog korijena srednje kvadratne pogreške može imati veliki utjecaj kako na rezultat učenja, tako i na brzinu izračuna i korake učenja. Međutim, mreža ne uspijeva naučiti umnožak šest puta šest ni za manju vrijednost NRMS -a.

#### 4.1.4 Brzina učenja

Na ukupan iznos koraka učenja i vrijeme učenja veliki utjecaj može imati brzina učenja,  $\eta$ , koju također zadaje korisnik. U tablici 4.10 prikazani su korišteni parametri te konačni rezultati testiranja za sve četiri aktivacijske funkcije.

Tablica 4.10. Prikaz rezultata testiranja za vrijednosti  $\eta$  0.1, 0.05 i 0.5

Neuroni skrivenog sloja:		5											
Momentum:		0											
Nrms:		0,03											
		Bipolarna sigmoidalna funkcija			Unipolarna sigmoidalna funkcija			Sinusna funkcija			Gaussianva funkcija		
Eta:		0,05	0,1	0,5	0,05	0,1	0,5	0,05	0,1	0,5	0,05	0,1	0,5
Koraci		3647	2251	1612	5159	2843	1566	521	427	1000	443	344	1005
Vrijeme		13,2	11,49	7,76	18,2	11,98	6,27	2,53	3,45	4,18	3,08	2,96	5,65

Dakle, brzina učenja  $\eta$  ima utjecaj na korake učenja. Kod bipolarne i unipolarne sigmoidalne funkcije vidljivo je da za mali iznos  $\eta$ , koraci učenja su znatno veći nego kod većeg iznosa. Kod sinusne i Gaussianove funkcije kod većeg vrijednosti  $\eta$  broj koraka se povećao kao i u slučaju manje vrijednosti NRMS-a. Međutim, i kod malog iznosa  $\eta$  koraci učenja su se povećali, ali za manji iznos u odnosu na veći koeficijent brzine učenja.

#### 4.1.5 Momentum prvog i drugog reda

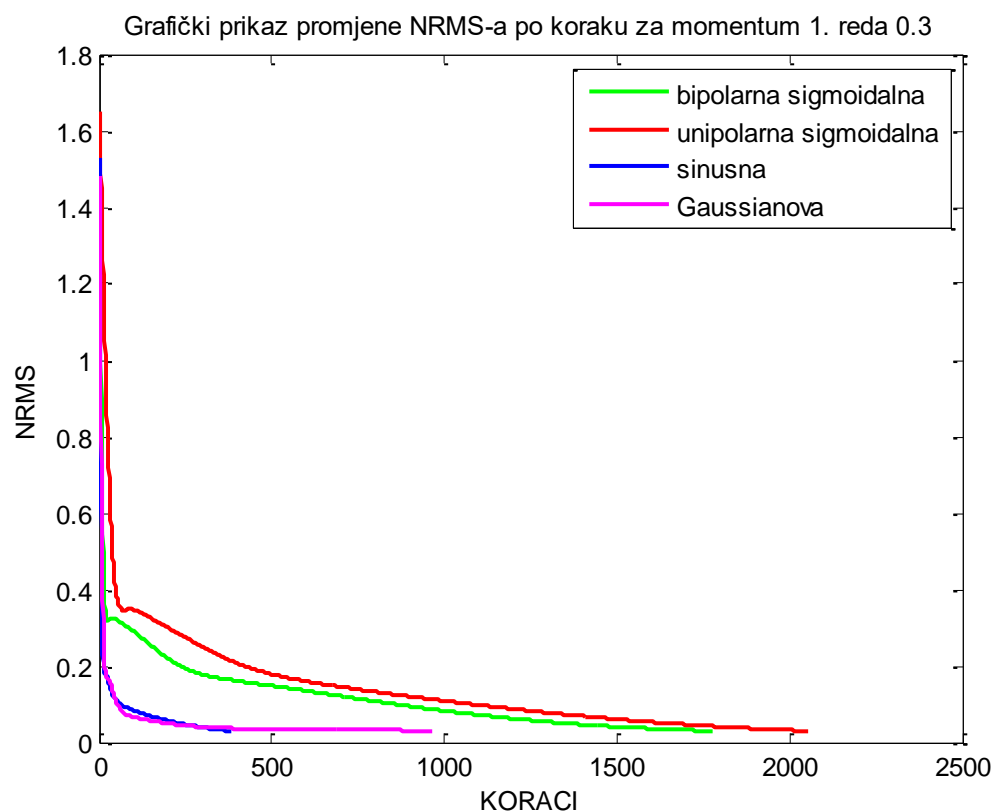
Preostaje još provjeriti utjecaj momentuma prvog i drugog reda na proces učenja. Za početak slijedi provjera utjecaja momentuma prvog reda. Vrijednosti momentuma prvog reda postavljene su na 0.3, 0.5 i 0.7. Ostali parametri prikazani su u tablici 4.11, dok su rezultati ispitivanja prikazani u tablici 4.12. Kod izračuna korištene su težine prikazane u (4.1) i (4.2), kao i u dosadašnjim testiranjima.

Tablica 4.11. Zadani parametri prilikom ispitivanja momentuma prvog reda

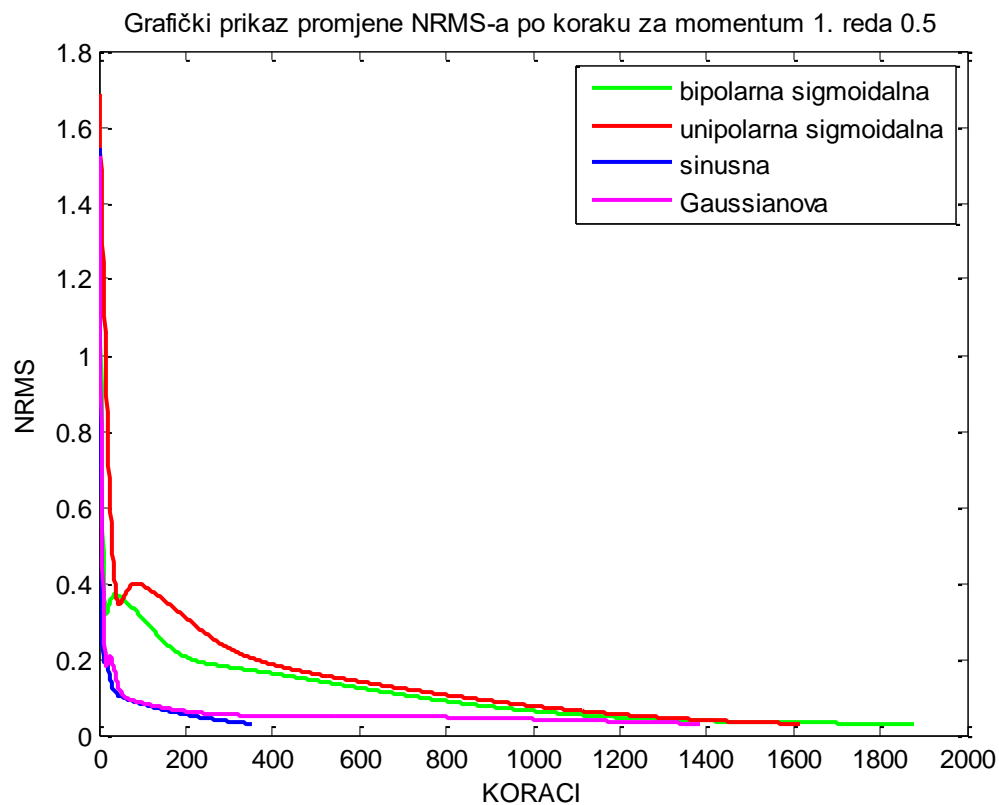
Vrijednost momentuma prvog reda:	0,3 ; 0,5 ; 0,7
Brzina učenja:	0,1
NRMS	0,03
Broj neurona skrivenog sloja:	5

Tablica 4.12. Prikaz rezultata testiranja za vrijednosti momentuma prvog reda 0.3, 0.5 i 0.7

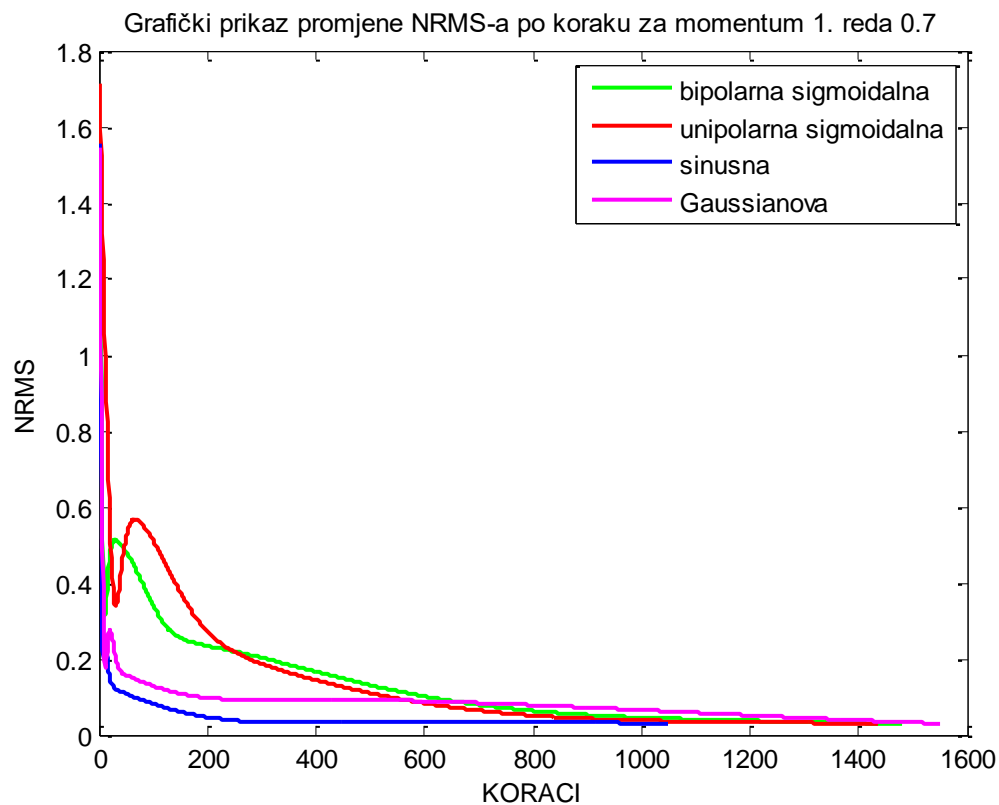
	Momentum 1. reda 0,3		Momentum 1. reda 0,5		Momentum 1. reda 0,7	
	Koraci:	Vrijeme (s):	Koraci:	Vrijeme (s):	Koraci:	Vrijeme (s):
Bipolarna sigmoidalna funkcija	1778	7,6944	1479	7,841	1877	9,2854
Unipolarna sigmoidalna funkcija	2054	9,8412	1435	7,2884	1614	8,2895
Sinusna funkcija	381	4,0287	1050	5,7387	352	2,5834
Gaussianova funkcija	963	9,5505	1549	9,9497	1387	10,5341



Slika 4.3 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija djelovanjem momentuma prvog reda 0.3

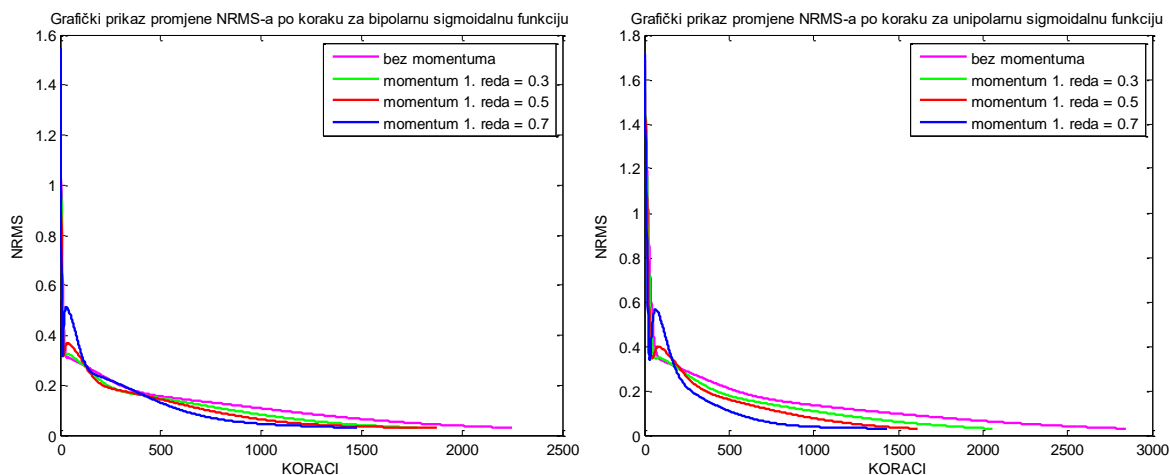


Slika 4.4 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija djelovanjem momentuma prvog reda 0.5



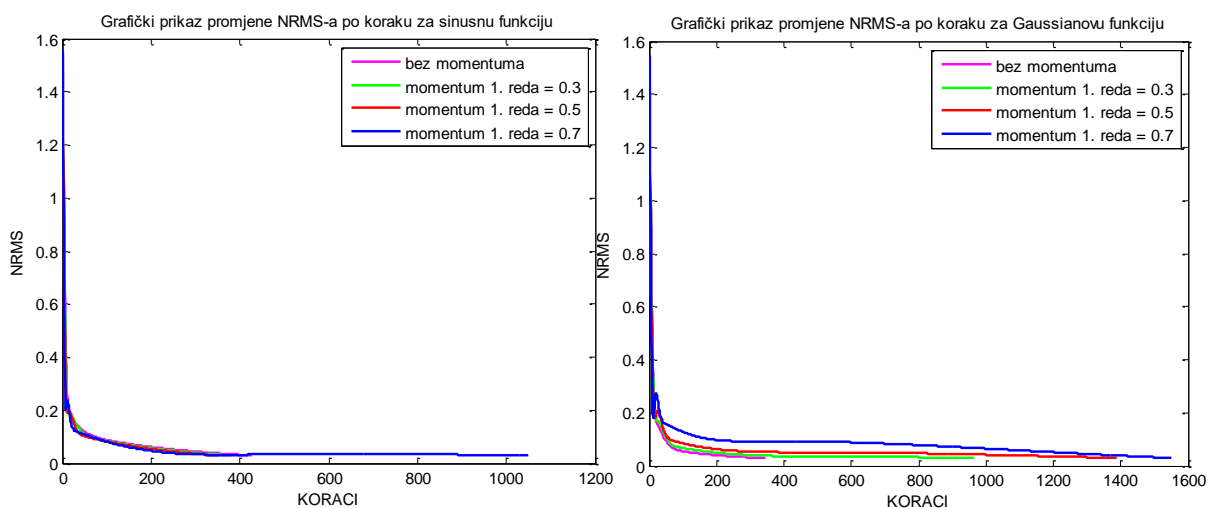
Slika 4.5 Grafički prikaz promjene NRMS-a po koraku za različite aktivacijske funkcija djelovanjem momentuma prvog reda 0.7

Iz tablice 4.12, te grafova prikazanih na slici 4.3, 4.4 i 4.5, vidljiva je slična situacija kao i kod brzine učenja eta. Dakle, postavljanjem momentuma prvog reda na 0.3 vidi se smanjenje koraka učenja kod bipolarne i unipolarne sigmoidalne funkcije. Trend smanjivanja koraka učenja ovih dviju aktivacijskih funkcija nastavlja se i upotrebom momentuma prvog reda iznosa 0.5 i 0.7.



Slika 4.6 Grafički prikaz promjene NRMS-a po koraku za bipolarnu i unipolaru sigmoidalnu funkciju primjenom momentuma prvog reda 0.3, 0.5 i 0.7

Međutim, upotrebom momentuma prvog reda vrijednosti 0.3 dolazi do povećanja koraka učenja sinusne i Gaussianove funkcije. Koraci učenja nastavljaju rasti i upotrebom momentuma prvog reda iznosa 0.5 i 0.7.



Slika 4.7 Grafički prikaz promjene NRMS-a po koraku za sinusnu i Gaussianovu funkciju primjenom momentuma prvog reda 0.3, 0.5 i 0.7

Za provjeru rada mreže upotrebom momentuma drugog reda korišteni su parametri prikazani u tablica 4.13.

Tablica 4.13 Zadani parametri prilikom ispitivanja momentuma drugog reda

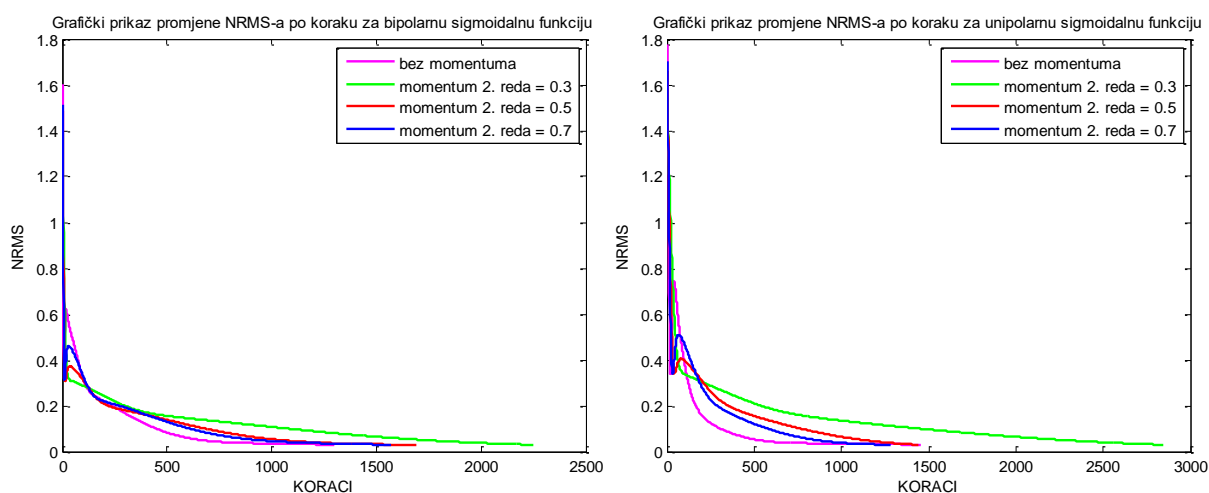
Vrijednost momentuma drugog reda:	0.3, 0.5, 0.7
Brzina učenja:	0,1

NRMS	0,03
Broj neurona skrivenog sloja:	5

Prema rezultatima prikazanim u tablici 4.14 vidljivo je da koraci učenja bipolarne sigmoidalne funkcije nastavljaju padati kao i u slučaju korištenja momentuma prvog reda. Ipak, vidljiv je brži pad, tj. koraci učenja se brže smanjuju nego kod primjene momentuma prvog reda. Što se tiče unipolarne sigmoidalne funkcije, koraci učenja padaju, a potom opet rastu, dok su u slučaju u kojem se koristi momentum prvog reda u stalnom padu. Što se tiče sinusne funkcije, primjenom momentuma drugog reda koraci se malo smanjuju, a kako se iznos momentuma povećava, tako koraci opet postaju sve veći. Gaussianova funkcija također upotrebom malog momentuma drugog reda zahtjeva veći broj koraka učenja, a upotrebom većeg momentuma drugog reda koraci učenja se smanjuju. Prikaz koraka učenja mreže prilikom korištenja momentuma drugog reda prikazan je u tablici 4.14.

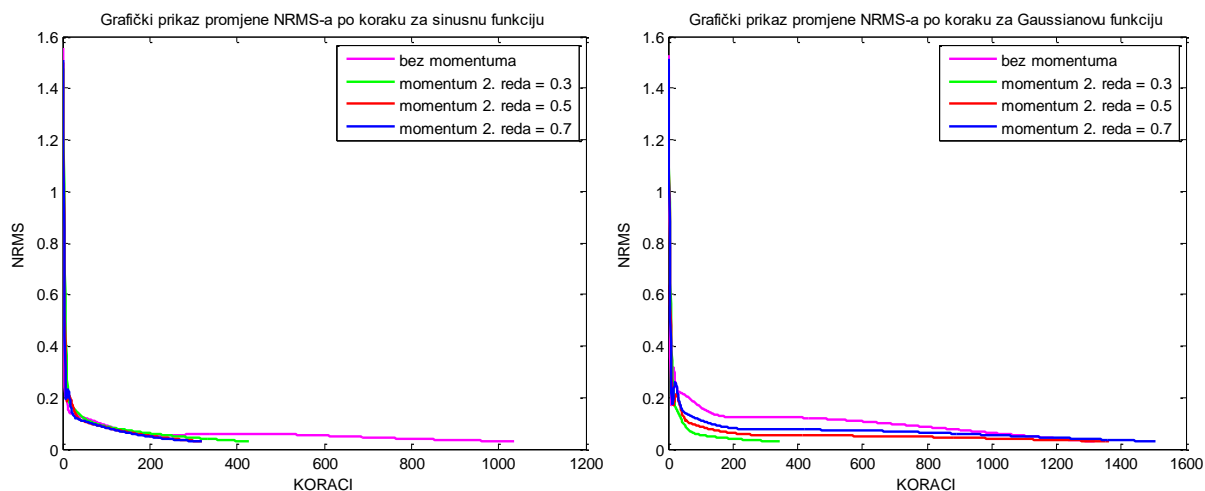
Tablica 4.14 Prikaz rezultata testiranja za vrijednosti momentuma drugog reda 0.3, 0.5 i 0.7

	Momentum 2. reda 0.3		Momentum 2. reda 0.5		Momentum 2. reda 0.7	
	Koraci:	Vrijeme (s):	Koraci:	Vrijeme (s):	Koraci:	Vrijeme (s):
Bipolarna sigmoidalna funkcija	1688	8,6258	1568	7,7953	1300	6,6477
Unipolarna sigmoidalna funkcija	1443	7,4765	1284	6,8595	1457	7,4893
Sinusna funkcija	320	2,6372	318	2,5048	1036	5,6849
Gaussianova funkcija	1364	9,9693	1507	9,4168	1319	10,0429



Slika 4.8 Grafički prikaz promjene NRMS-a po koraku za bipolarnu i unipolarnu sigmoidalnu funkciju primjenom momentuma drugog reda 0.3, 0.5 i 0.7





Slika 4.9 Grafički prikaz promjene NRMS-a po koraku za sinusnu i gaussianovu funkciju primjenom momentuma drugog reda 0.3, 0.5 i 0.7

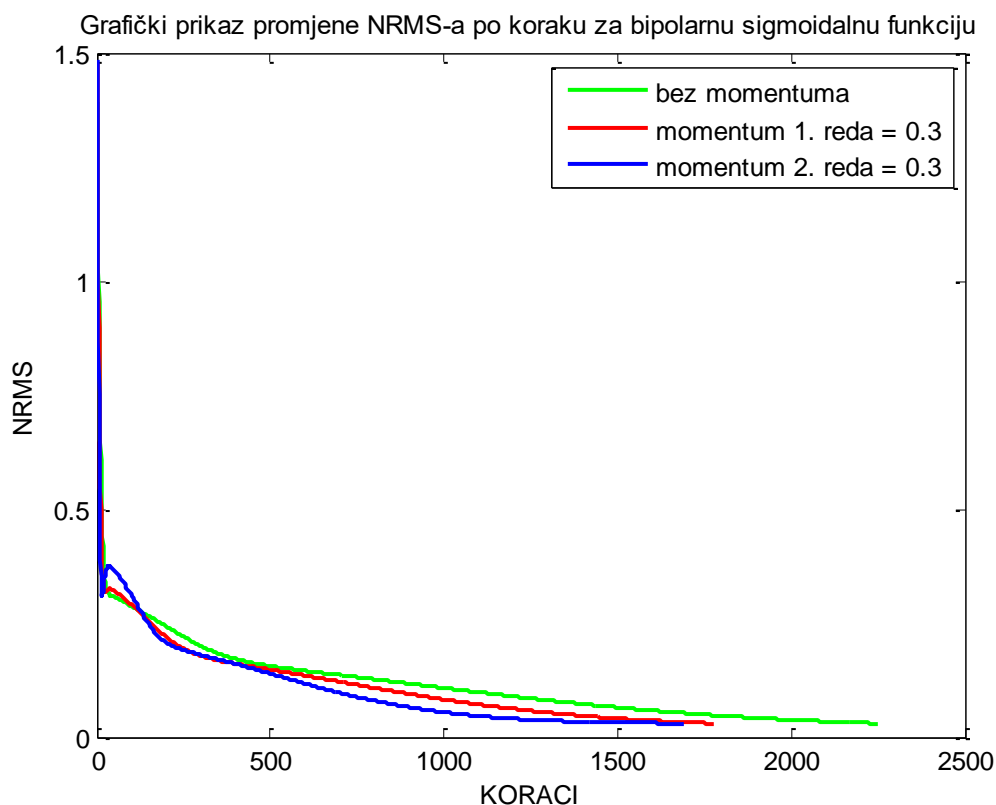
Radi boljeg pregleda, u nastavku slijedi prikaz koraka učenja mreže s dva ulaza i jednim izlazom u slučaju upotrebe momentuma prvog i drugog reda te bez momentuma.

Tablica 4.15 Prikaz rezultata testiranja različitih momentuma za bipolarnu sigmoidalnu funkciju

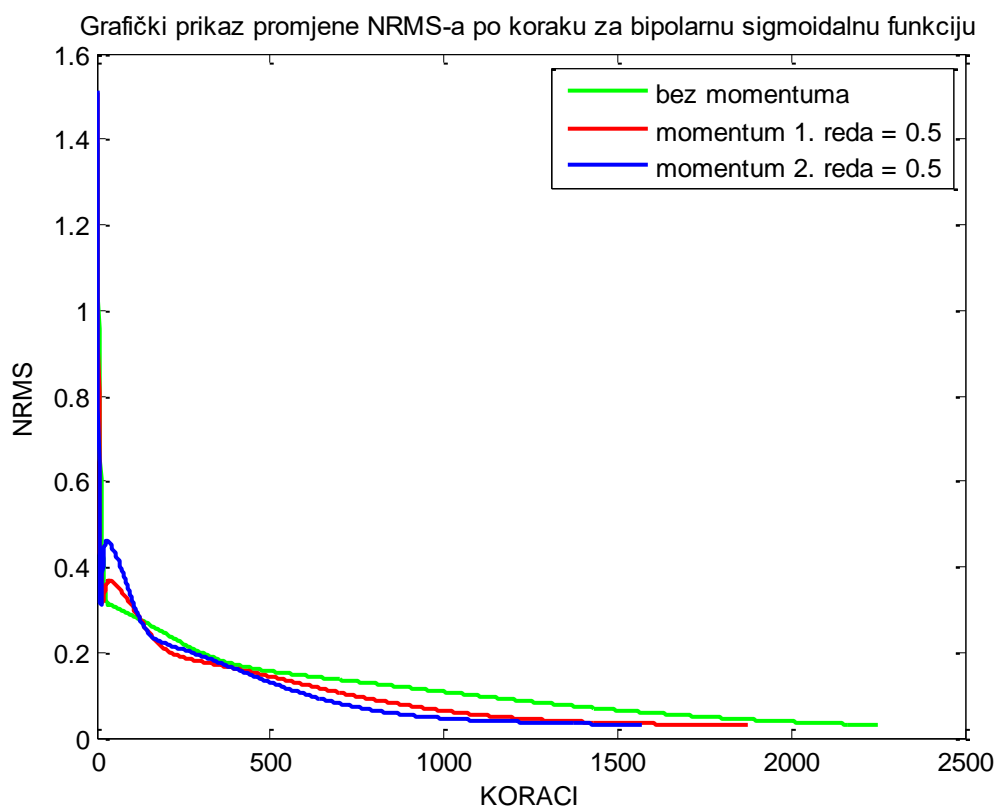
	Momentum 1. reda
	Momentum 2. reda
	Bez momentuma

Bipolarna sigmoidalna funkcija		
		Koraci:
Bez momentuma:	0	2251
Momentum prvog reda:	0,3	1778
Momentum drugog reda:	0,3	1688
Bez momentuma:	0	2251
Momentum prvog reda:	0,5	1877
Momentum drugog reda:	0,5	1568
Bez momentuma:	0	2251
Momentum prvog reda:	0,7	1479
Momentum drugog reda:	0,7	1300

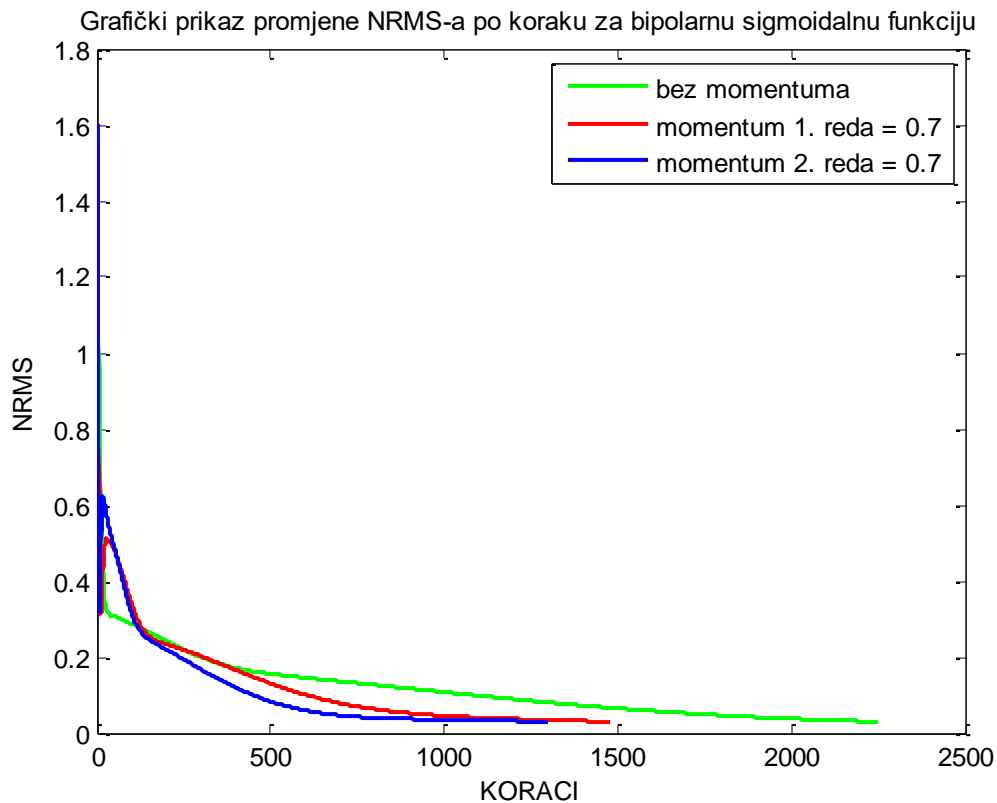
Na tablici 4.15 jasno je vidljiv pad koraka učenja bipolarne sigmoidalne funkcije uporabom momentuma prvog i drugog reda. Može se zaključiti da je upotreba momentuma drugog reda bolje rješenje za bipolarnu sigmoidalnu funkciju jer vidljivo ubrzava učenje mreže. U nastavku slijedi grafički prikaz dobivenih rezultata.



Slika 4.10 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3



Slika 4.11 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5



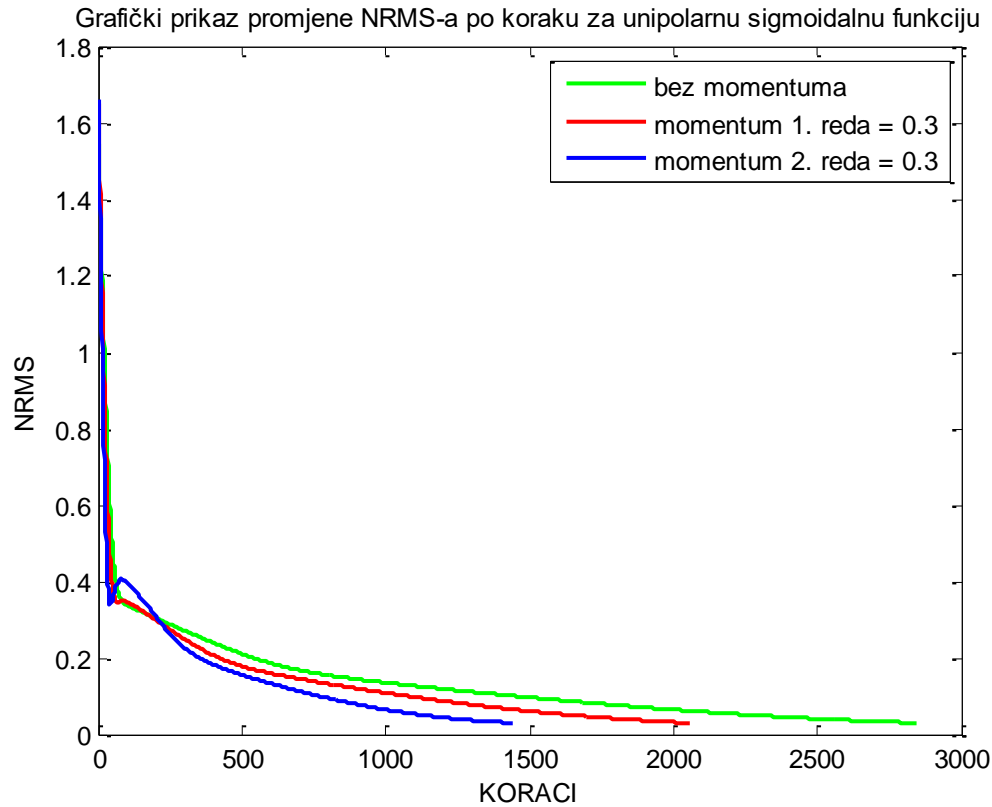
Slika

Slika 4.12 Grafički prikaz promjene NRMS-a po koraku za bipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7

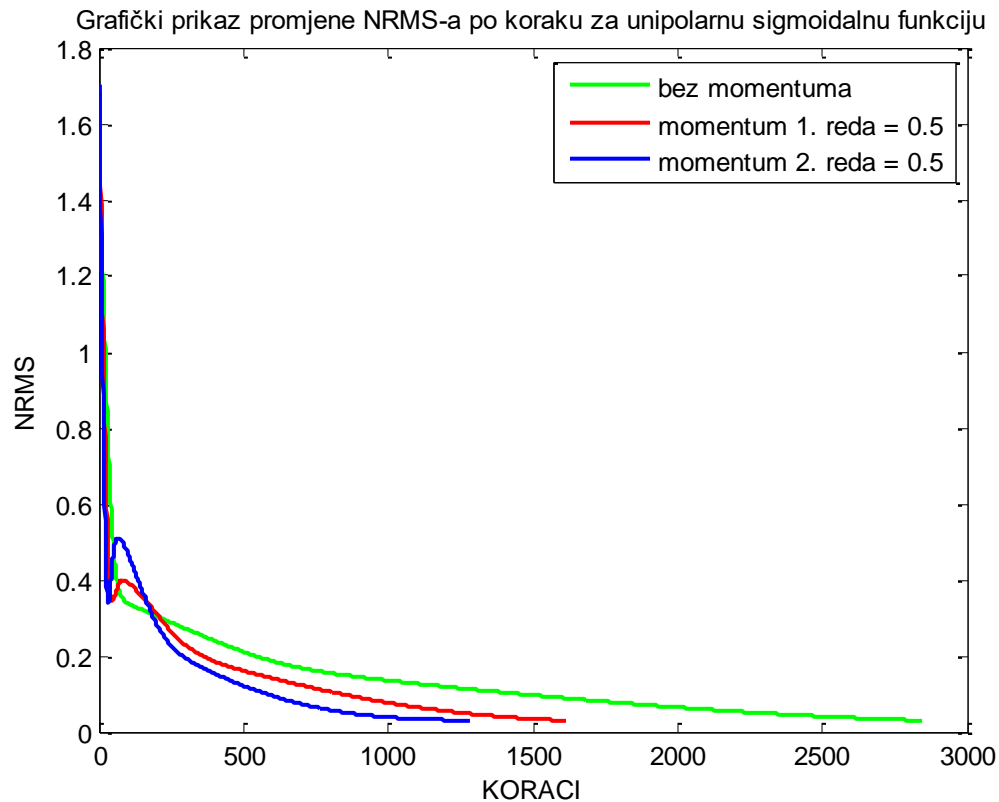
Tablica 4.16 Prikaz rezultata testiranja različitih momentuma za unipolarnu sigmoidalnu funkciju

Unipolarna sigmoidalna funkcija		
	Koraci:	
Bez momentuma:	0	2843
Momentum prvog reda:	0,3	2054
Momentum drugog reda:	0,3	1443
Bez momentuma:	0	2843
Momentum prvog reda:	0,5	1614
Momentum drugog reda:	0,5	1284
Bez momentuma:	0	2843
Momentum prvog reda:	0,7	1435
Momentum drugog reda:	0,7	1457

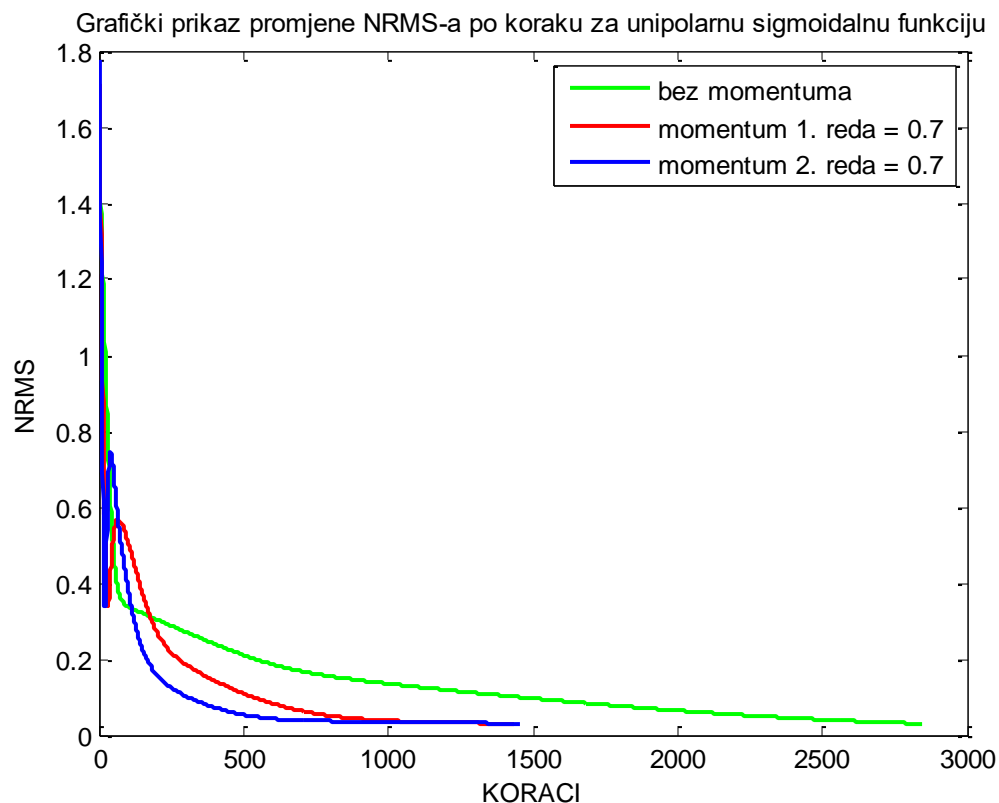
U tablici 4.16 vidljiv je pad koraka učenja primjenom momentuma prvog reda. Upotrebom momentuma drugog reda također postoji pad broja koraka učenja, međutim kod većih iznosa momentuma drugog reda broj koraka učenja raste. Prema tome bolje je prilikom korištenja unipolarne sigmoidalne funkcije koristiti momentum prvog reda jer je sigurno da će se broj koraka smanjivati.



Slika 4.13 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3



Slika 4.14 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5

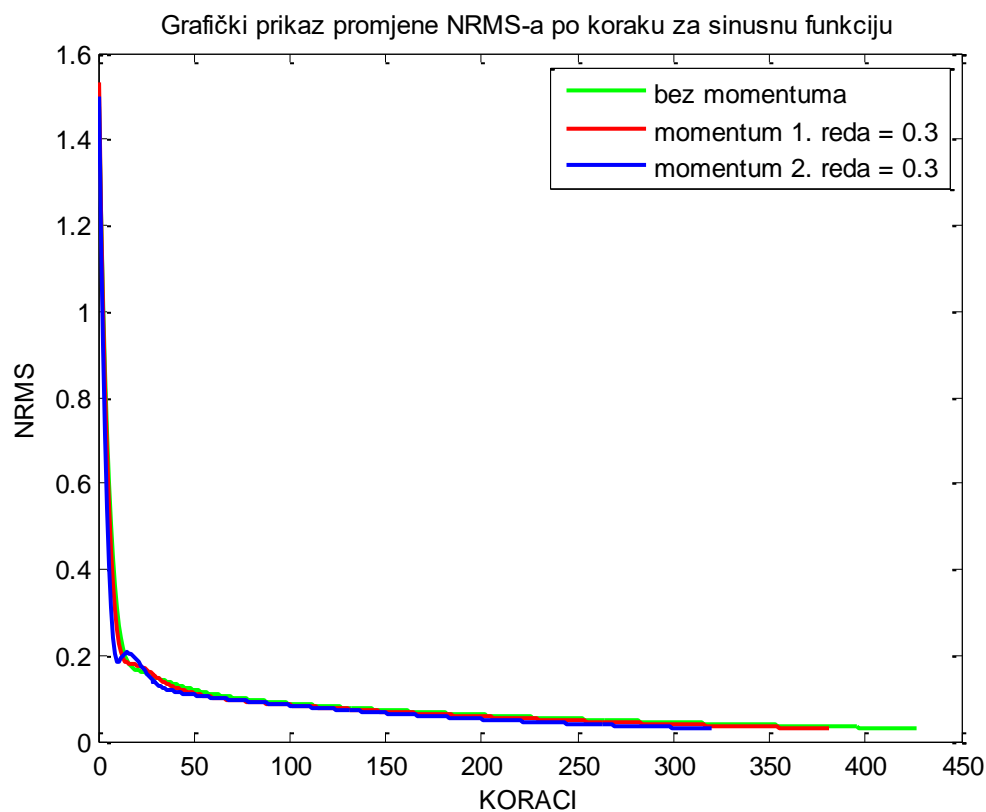


Slika 4.15 Grafički prikaz promjene NRMS-a po koraku za unipolarnu sigmoidalnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7

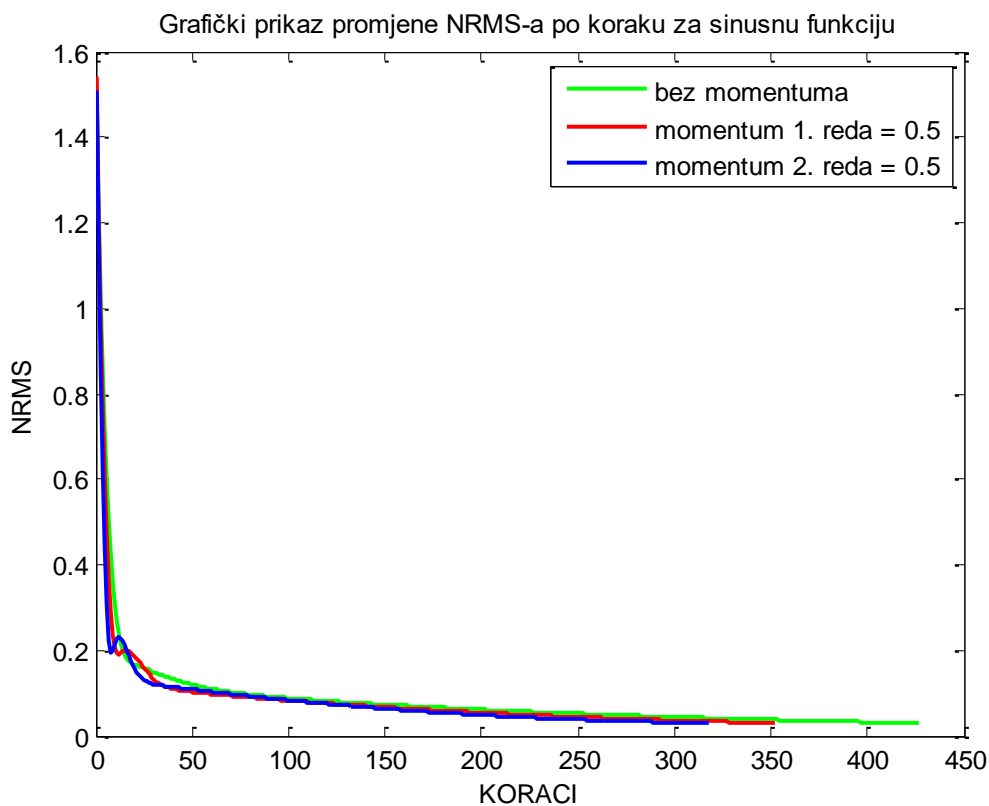
Tablica 4.17 Prikaz rezultata testiranja različitih momentuma za sinusnu funkciju

Sinusna funkcija		
		Koraci:
Bez momentuma:	0	427
Momentum prvog reda:	0,3	381
Momentum drugog reda:	0,3	320
Bez momentuma:	0	427
Momentum prvog reda:	0,5	352
Momentum drugog reda:	0,5	318
Bez momentuma:	0	427
Momentum prvog reda:	0,7	1050
Momentum drugog reda:	0,7	1036

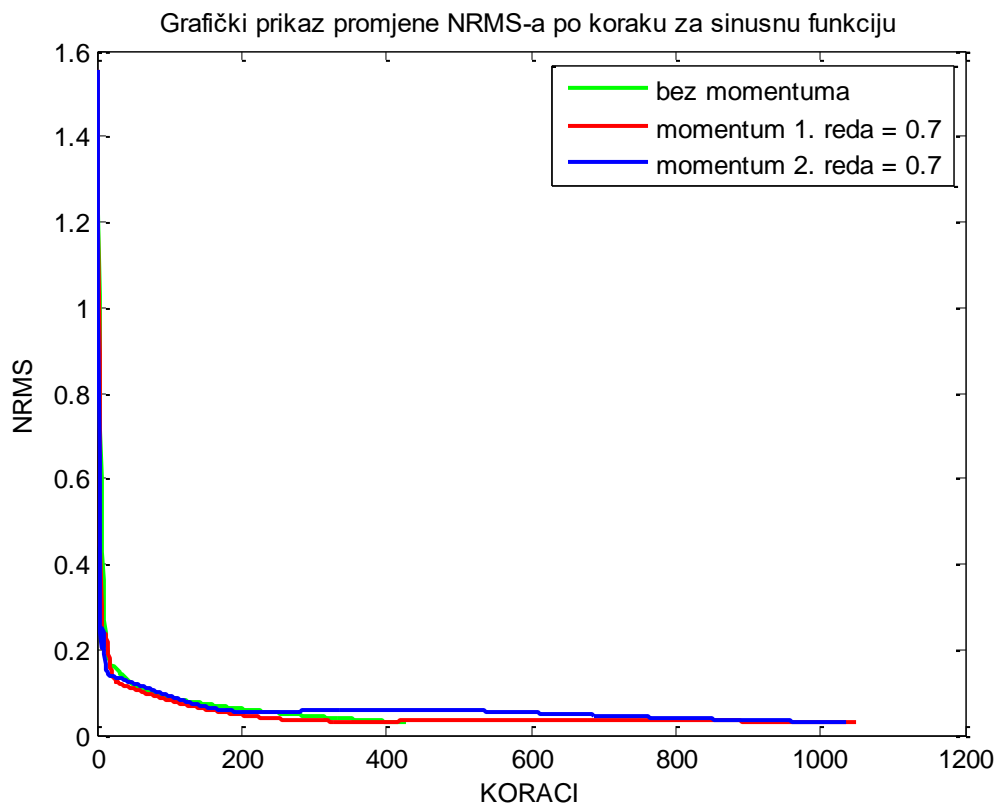
Vidljivo je na temelju tablice 4.17 da upotrebom momentuma prvog reda dolazi do pada broja koraka učenja ukoliko se radi o maloj vrijednosti momentuma. U slučaju većih vrijednosti dolazi do povećanja koraka učenja. Isti rezultat dobiva se primjenom momentuma drugog reda. Dakle, samo mali iznosi momentuma prvog i drugog reda smanjuju broj koraka učenja mreže. Može se još i primijetiti da je u slučaju sinusne funkcije momentum drugog reda brži, tj. za iste vrijednosti težina neurona skrivenog sloja i jednake parametre zahtjeva manji broj koraka učenja.



Slika 4.16 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3



Slika 4.17 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5

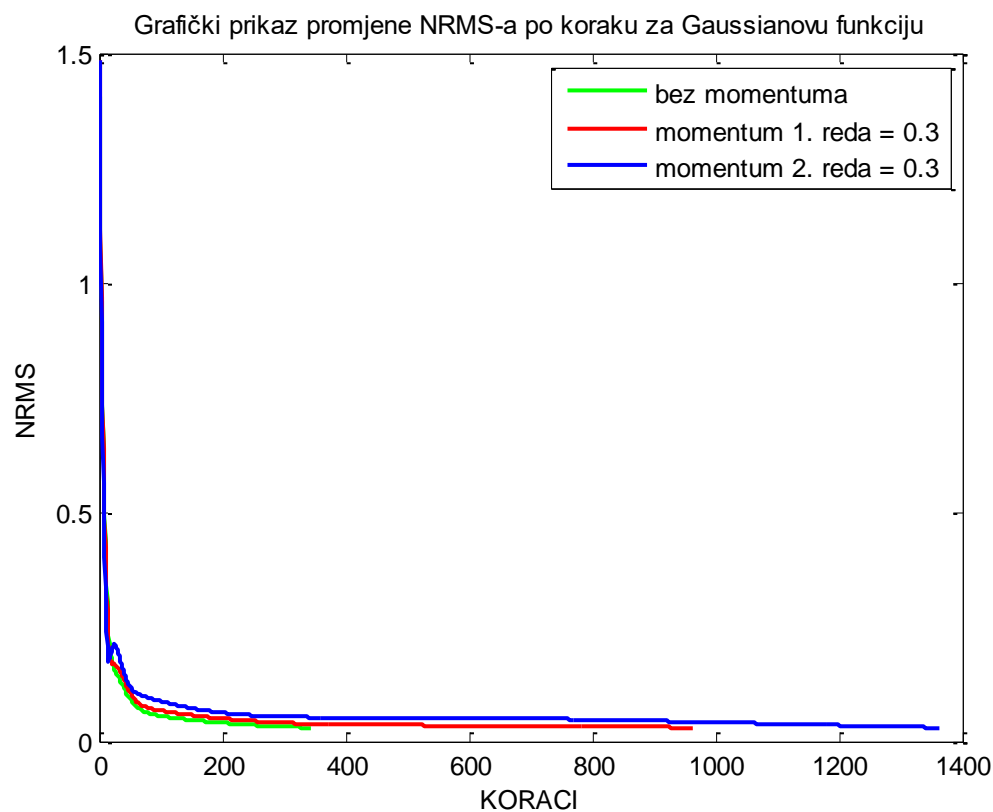


Slika 4.18 Grafički prikaz promjene NRMS-a po koraku za sinusnu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7

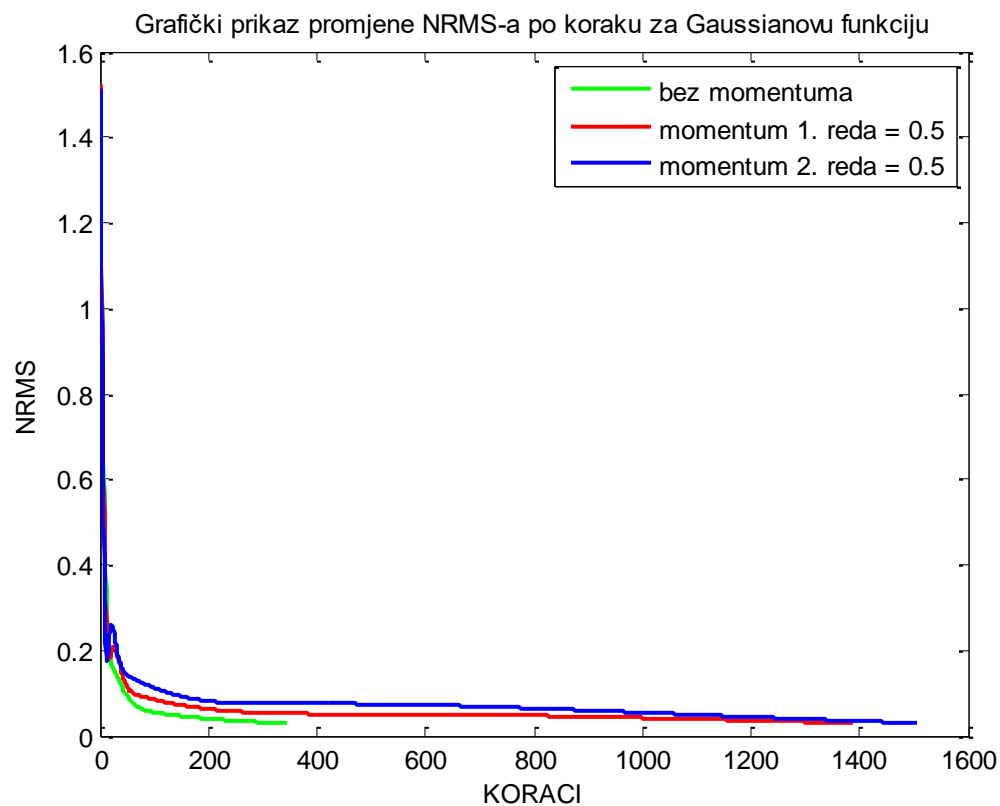
Tablica 4.18 Prikaz rezultata testiranja različitih momentuma za Gaussianovu funkciju

Gaussianova funkcija		
	Koraci:	
Bez momentuma:	0	344
Momentum prvog reda:	0,3	963
Momentum drugog reda:	0,3	1364
Bez momentuma:	0	344
Momentum prvog reda:	0,5	1387
Momentum drugog reda:	0,5	1507
Bez momentuma:	0	344
Momentum prvog reda:	0,7	1549
Momentum drugog reda:	0,7	1319

Iz tablice 4.18 vidljivo je da kod ove funkcije upotrebom momentuma i prvog i drugog reda, ukoliko unesemo malu vrijednost, dolazi da povećanja koraka učenja. Kod momentuma prvog reda koraci učenja rastu s povećanjem vrijednosti momentuma. Što se tiče momentuma drugog reda, koraci učenja rastu sve do viših vrijednosti momentuma, kada dolazi do blagog pada potrebnih koraka.

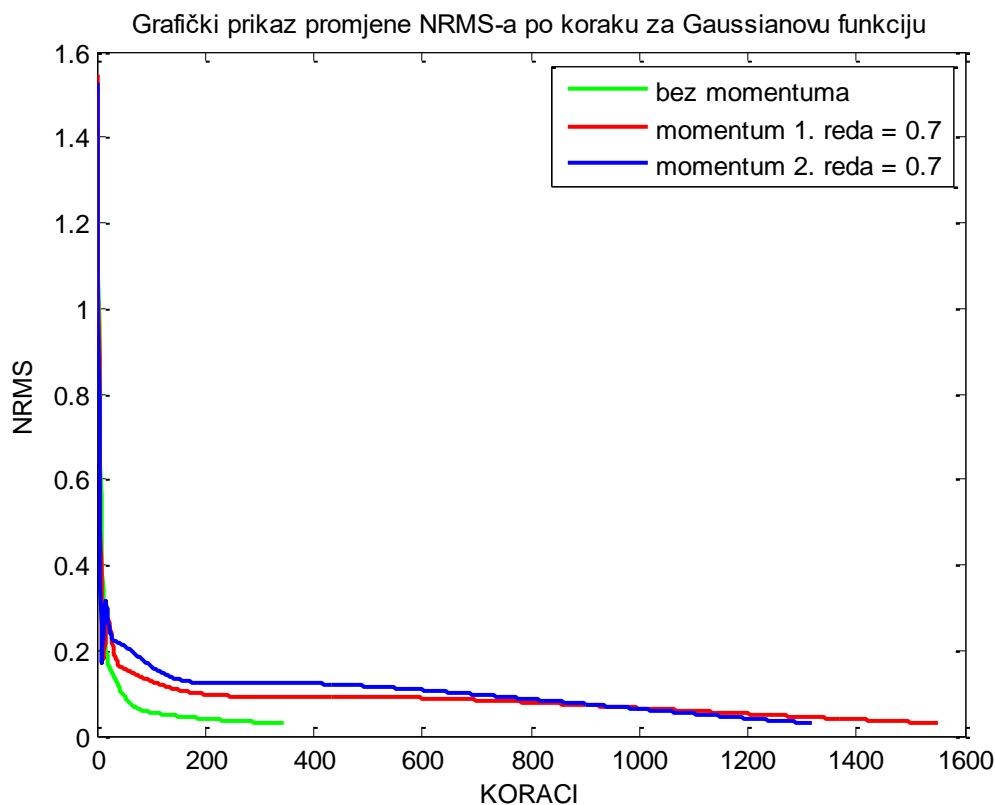


Slika 4.19 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.3



Slika 4.20 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.5





Slika 4.21 Grafički prikaz promjene NRMS-a po koraku za gaussianovu funkciju bez momentuma, momentumom prvog i drugog reda iznosa 0.7

#### 4.1.6 Testiranje mreže s dva ulaza i jednim izlazom

U nastavku će biti prikazani rezultati učenja mreže ako mreža ne nauči svih 25 kombinacija brojeva. Mogućih 25 kombinacija brojeva prikazani su u tablici 4.19.

Tablica 4.19 Prikaz 25 mogućih kombinacija potrebnih za učenje mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Prilikom učenja mreže koristit ću parametre učenja prikazane u tablici 4.20. Ispitivanje ću za početak vršiti pomoću sve četiri aktivacijske funkcije. Također, koristit ću iste težine skrivenog sloja kao i kod prethodnih testiranja, (4.1) i (4.2).

Tablica 4.20 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona skrivenog sloja:	5

Za početak mrežu ću prestati učiti množiti broj 3, tj. izbacit ću broj 3 iz svih kombinacija u kojima s pojavljuje. Označit ću te kombinacije zelenom bojom na tablici 4.21.

Tablica 4.21 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Dakle, mreža će u ovom primjeru naučiti množiti ukupno 16 kombinacija brojeva. Broj tri odabran je zato što se nalazi u sredini, tj. mreža je naučila množiti brojeve ispred i iza broja 3. Kako je mreža naučila ostalih 9 kombinacija brojeva bit će ispitano korištenjem sve četiri aktivacijske funkcije.

Tablica 4.22 Prikaz rezultata testiranja za 9 ne učenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		2983		3938		441		354	
Vrijeme (s):		8,6285		11,0624		2,1335		2,2015	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
1	3	2,7765	3	2,8085	3	2,5577	3	2,7788	3
2	3	6,101	6	6,2775	6	5,9631	6	5,8306	6
3	3	9,3187	9	9,4349	9	9,2495	9	8,9236	9
4	3	12,3848	12	12,4104	12	12,3335	12	12,0713	12
5	3	15,2684	15	15,1911	15	15,1396	15	15,2606	15
3	1	2,9094	3	2,9503	3	3,2528	3	2,7953	3
3	2	6,1066	6	6,2466	6	6,2455	6	5,708	6
3	4	12,4258	12	12,4285	12	12,2557	12	12,2008	12
3	5	15,3228	15	15,2018	15	15,2505	15	15,2766	15

Iz tablice 4.22 vidljivo je da je mreža uspjela dati točan rezultat za svih 9 kombinacija koje nije naučila. Razlog tom ishodu leži u činjenici da je mreža naučila množiti sve kombinacije brojeva

sa 2 i sa 4. U tablici 4.4, 4.7 i 4.9 mreža je testirana na umnožak broja 6 i nije davala dobar rezultat. Razlog je u tome što je mreža naučila množiti kombinacije brojeva do 5, a ni jedan broj nakon njega.

Slijedi testiranje mreže na način da joj se uz 3 maknu sve kombinacije koje u sebi sadrže 1. Na taj način mreži ostaje za naučiti 9 kombinacija brojeva.

Tablica 4.22 Prikaz rezultata testiranja za 9 ne učenih kombinacija

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Dakle, u ovom slučaju izbačen je broj 1 koji je na neki način sličan broju 6. Razlog je u činjenici da je mreža naučila množiti brojeve nakon 1, a ne zna ni jedan broj prije 1, kao što ne zna što je s brojevima iza 5. Prema tome pretpostavka je da mreža neće uspjeti naučiti množiti kombinacije brojeva koje u sebi sadrže 1.

Tablica 4.24 Prikaz rezultata testiranja za 16 ne naučenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		10430		11052		798		295	
Vrijeme (s):		19,3229		19,9507		2,5457		1,9362	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	3	5,9592	6	5,9579	6	5,7628	6	5,7687	6
1	1	0,09345	0	0,43479	0	0,43047	0	1,9967	2
1	2	1,3452	1	1,095	1	1,2353	1	2,1078	2
1	3	2,4436	2	2,3566	2	2,1448	2	2,567	3
1	4	3,4302	3	3,4994	3	3,2106	3	3,337	3
1	5	4,3753	4	4,7418	5	4,4751	4	4,3197	4
3	1	2,7387	3	2,6642	3	3,2799	3	3,404	3
3	2	6,023	6	6,0697	6	6,2103	6	5,9364	6
3	3	9,2557	9	9,3248	9	9,184	9	8,9372	9
3	4	12,3437	12	12,3524	12	12,1917	12	12,151	12
3	5	15,2143	15	15,1386	15	15,2182	15	15,2697	15
2	1	1,5036	2	1,2714	1	1,9837	2	2,6664	3
4	1	3,8913	4	3,4994	4	4,2728	4	4,3333	4
5	1	5,0638	5	5,0641	5	4,9312	5	5,5517	5
4	3	12,3192	12	12,3621	12	12,3139	12	12,0719	12
5	3	15,1604	15	15,1212	15	15,0715	15	15,1596	15

Na tablici 4.24 vidljivo je da je pretpostavka prije početka testiranja bila točna. Mreža ima problema s izračunom kombinacija brojeva koji u sebi sadrže jedan, zato što nije naučila ništa što se nalazi ispred jedinice. Ako promotrimo kombinacije s brojem tri, vidimo da je mreža i ovog puta uspješno naučila množiti te kombinacije brojeva, zato što je naučila množiti kombinacije s brojem dva i brojem četiri. Prema tome isti rezultat bi bio i kad bi u testu provjeravali množenje mreže brojem šest, tj. bila bi mala vjerojatnost dobivanja točnog rješenja kao i u ovom slučaju s jedinicom.

Budući da sve četiri mreže daju skoro jednak broj točnih kombinacija, u nastavku ću testirati samo po jednu aktivacijsku funkciju.

Pomoću Gaussianove funkcije još ću jednom prikazati razliku u točnosti rezultata učenja mreže ako se iz mreže izuzme broj 1 kao jedan slučaj i broj 2 kao drugi slučaj. Dakle, u prvom slučaju mreža će naučiti ukupno 16 kombinacija brojeva, ali očekuje se više od polovine netočnih rezultata učenja s obzirom na to da mreža ne zna množiti sa ni jednim brojem ispred 1. U drugom slučaju mreža će također naučiti 16 kombinacija, ali vrijedi pretpostavka da će mreža dati točan odgovor i za kombinacije brojeva koje u sebi sadrže 2, jednako kako je dala i u slučaju s brojem 3.

Tablica 4.25 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 1 i 2 radi testiranja Gaussianovom funkcijom

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.26 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1

Gaussianova funkcija			
Koraci:		287	
Vrijeme(s):		2,1149	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže
1	1	2,2431	2
1	2	2,2896	2
1	3	2,6608	3
1	4	3,3293	3
1	5	4,2113	4
2	1	2,8309	3
3	1	3,5017	4
4	1	4,388	4
5	1	5,5962	6

Tablica 4.27 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 2

Gaussianova funkcija			
Koraci:		331	
Vrijeme(s):		2,6037	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže
2	1	1,9914	2
2	2	3,7584	4
2	3	5,7988	6
2	4	7,9552	8
2	5	10,046	10
1	2	1,9625	2
3	2	5,6679	6
4	2	7,726	8
5	2	10,1239	10

Prema rezultatima testiranja potvrđene su pretpostavke te je mreža naučila jedva tri kombinacije koje u sebi sadrže jedan. Što se tiče broja dva, mreža je uspješno naučila sve kombinacije koje u sebi sadrže broj dva iako ga nije naučila, ali je naučila kombinacije brojeva koji u sebi sadrže jedan i tri.

U nastavku slijedi testiranje tako da se iz kombinacija izuzmu 3 i 4, na način da je 3 uvijek prvi broj, a 4 uvijek drugi broj.

Tablica 4.28 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže prvi broj 3 i drugi broj 4

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.29 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 3 i drugi broj 4

Unipolarna sigmoidalna funkcija			
Koraci:		3828	
Vrijeme(s):		10,6936	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže
3	1	2,9255	3
3	2	6,1723	6
3	3	9,3713	9
3	4	12,4276	12
3	5	15,2873	15
1	4	3,6747	4
2	4	8,1129	8

4	4	12,4276	12
5	4	20,152	20

Prema rezultatima vidljivo je da je mreža uspjela naučiti sve kombinacije brojeva koje joj prethodno nisu bile zadane za učenje. Razlog tome jest što je mreža naučila množiti brojeve ispred i nakon broja 3 u prvom stupcu, te prije i nakon broja 4 u drugom stupcu. Vrijedi provjeriti i učenje mreže ukoliko se iz učenja potpuno izuzme broj tri i one kombinacije u kojima je prvi broj dva.

Tablica 4.30 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 i prvi broj 2

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.31 Prikaz rezultata testiranja za 14 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 2

Unipolarna sigmoidalna funkcija			
Koraci:		5108	
Vrijeme(s):		13,4665	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže
3	1	2,9616	3
3	2	6,3183	6
3	3	9,8723	10
3	4	12,6103	13
3	5	15,3715	15
1	3	3,0633	3
2	3	6,4496	6
4	3	12,4465	12
5	3	15,1416	15
2	1	1,9377	2
2	2	4,3126	4
2	3	6,4496	6
2	4	8,3588	8
2	5	10,1376	10

Prema tablici 4.31 vidljivo je da je mreža uspjela točno odrediti čak 12 kombinacija ne naučenih brojeva. Jedine dvije pogreške javile su joj se u slučaju broja 3, zato što se 3 u prvom stupcu nalazio iza broja 2 koji također nije bio naučen, što je u konačnici rezultiralo krivim rezultatom.

Zadnje testiranje mreže s dva ulaza i jednim izlazom provest ću tako da ću potpuno izbaciti 2 i 4 kao moguće kombinacije brojeva. Također, testirat ću rezultat i s brojem 6.

Tablica 4.32 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže brojeve 2 i 4

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Znači mreža će naučiti ukupno 9 kombinacija, dok će 16 kombinacija biti nepoznato. Prema dosadašnjem testiranju, mreža bi za ovaj slučaj trebala biti potpuno točna iako se radi o velikom broju ne naučenih kombinacija brojeva.

Tablica 4.33 Prikaz rezultata testiranja za 16 ne naučenih kombinacija koje u sebi sadrže brojeve 2 i 4

Unipolarna sigmoidalna funkcija			
Koraci:		299	
Vrijeme(s):		2,3523	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže
2	1	1,9266	2
2	2	3,7656	4
2	3	5,8656	6
2	4	8,0458	8
2	5	10,1033	10
4	1	3,7963	4
4	2	7,778	8
4	3	12,0877	12
4	4	16,3922	16
4	5	20,3422	20
1	2	1,9008	2
1	4	3,7568	4
3	2	5,6991	6
3	4	12,2784	12
5	2	10,0518	10
5	6	20,3104	20
6	6	33,052	33
5	6	28,7357	29

Kao što je bilo predviđano, prema tablici 4.33 vidljivo je da je mreža uspješno naučila čak 16 kombinacija, iako ih je naučila samo 9. Također, kao što je već prije rečeno, mreža daje krivi rezultat ukoliko se testira broj 6. Da mreža može naučiti neki od brojeva koji dolaze nakon broja 6, mreža bi najvjerojatnije davala točniji, a možda i točan rezultat.

Važno je iz svih prikazanih primjera shvatiti kako neuronska mreža nije kalkulator. Jasno je vidljivo da rješenje neuronske mreže nikad nije cijeli broj, već decimalni. U programu mreža u konačnici izbacuje konačnu, zaokruženu vrijednost.

## 4.2 Ispitivanje mreže s dva ulaza i tri izlaza

Analognu ispitivanju rada neuronske mreže s dva ulaza i jednim izlazom slijedi ispitivanje mreže s dva ulaza i s tri izlaza.

### 4.2.1 Testiranje mreže s dva ulaza i tri izlaza

Za početak mrežu ću testirati tako da iz učenja uklonim sve kombinacije koje u sebi sadrže broj 3. Testiranja ću provesti pomoću sinusne i bipolarne sigmoidalne funkcije, s parametrima učenja prikazanim u tablici 4.20.

Tablica 4.34 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.35 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3

Bipolarna sigmoidalna funkcija							
Koraci:		3740					
Vrijeme(s):		12,1935					
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Umnožak	Odziv mreže
1	3	0,94196	1	9,8939	9	3,065	3
2	3	4,033	4	9,5142	9	5,9765	6
3	3	9,03	9	9,3532	9	8,8894	9
4	3	16,1195	16	9,414	9	11,8941	12
5	3	25,0508	25	9,6892	9	11,9739	15
3	1	8,9774	9	0,27774	1	2,8189	3
3	2	9,0121	9	4,0834	4	5,8604	6
3	4	9,0333	9	16,2178	16	11,9126	12
3	5	9,0405	9	24,5454	25	14,8967	15

Vidljivo je iz tablice 4.35 da mreža s dva ulaza i tri izlaza ima problema s izračunom devet ne naučenih kombinacija. Razlog pojave grešaka može biti taj što mreža računa s više izlaza. Također, iako matrica računa s 5 neurona skrivenog sloja, matrice težina  $\mathbf{V}$  i  $\mathbf{W}$  nisu jednake kao u slučaju mreže s jednim izlazom. Razlika je u matrici  $\mathbf{W}$  težina izlaznog sloja jer ona osim što ovisi o broju neurona skrivenog sloja, ovisi i o broju izlaznih neurona. Tako matrica  $\mathbf{W}$  u ovom slučaju poprima dimenziju  $[3 \times 6]$ , gdje se redovi odnose na broj izlaznih neurona, a stupci na broj skrivenih neurona uvećanih za bias. Matrice težina korištene u testiranju mreže s dva ulaza i tri izlaza prikazane su u (4.3) i (4.4).



$$\mathbf{V} = \begin{bmatrix} 0.3594053537 & 0.9194879170 & -0.4898097691 \\ 0.3101960079 & -0.3192285467 & 0.0119141033 \\ -0.6747765296 & 0.1705355020 & 0.3981534453 \\ -0.7620046369 & -0.5523761210 & 0.7818065051 \\ -0.0032718960 & 0.5025341186 & 0.9185828504 \end{bmatrix} \quad (4.3)$$

$$\mathbf{W} = \begin{bmatrix} -0.9076572187 & -0.8057364375 & 0.6469156567 \\ 0.3896572460 & -0.3658010399 & 0.9004440977 \\ -0.9311078390 & -0.1225112807 & -0.2368830858 \\ 0.5310335763 & 0.5903998023 & -0.6262547909 \\ -0.0204712084 & -0.1088275986 & 0.2926260202 \\ 0.4187296617 & 0.5093733640 & -0.4479498460 \end{bmatrix}^T \quad (4.4)$$

Slijedi testiranje mreže na način da ću iz mreže uz broj 3 izbaciti 4 kao prvi ulaz.

Tablica 4.36 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 i prvi broj 4

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.37 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 4

Bipolarna sigmoidalna funkcija							
Koraci:		4617					
Vrijeme(s):		11,882					
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Umnožak	Odziv mreže
1	3	0,83821	1	9,9068	10	2,9094	3
2	3	4,1243	4	9,4627	9	6,0057	6
3	3	9,2748	9	9,2759	9	9,0125	9
4	3	16,3754	16	9,3554	9	11,9954	12
5	3	25,0722	25	9,6815	10	14,9112	15
3	1	9,1472	9	0,15809	0	2,8754	3
3	2	9,2256	9	3,9935	4	5,936	6
3	4	9,3052	9	16,135	16	12,0877	12
3	5	9,3525	9	24,42	24	15,0783	15
4	1	16,2033	16	0,22705	0	3,8923	4

4	2	16,3235	16	4,0696	4	7,8717	8
4	3	16,3754	16	9,3554	9	11,9954	12
4	4	16,3939	16	16,202	16	16,1608	16
4	5	16,1568	16	24,4506	24	20,1437	20

Mreža dakle i u slučaju kad joj se oduzme prvi broj 4 griješi u izračunu kvadrata drugog broja. Zanimljivo je da mreža uspijeva u svim primjerima točno kvadrirati broj 4 koji nije naučila kao prvi broj, iako nije naučila ni broj 3, ali negativno kvadrira brojeve koji su izabrani kao drugi, a njih je naučila. U nastavku pokušat ću popraviti rezultate učenja mreže tako da ću nrms smanjiti sa 0.03 na 0.01, tj tražit ću veću točnost učenja.

Tablica 4.38 Zadani parametri prilikom novog ispitivanja mreže

Vrijednost momentuma :	0
Brzina učenja:	0,1
NRMS	0,01
Broj neurona skrivenog sloja:	5

Tablica 4.39 Prikaz rezultata testiranja za 14 ne naučenih kombinacija koje u sebi sadrže broj 3 i prvi broj 4 nakon promjene nrms-a sa 0.03 na 0.01

Bipolarna sigmoidalna funkcija							
Koraci:		6513					
Vrijeme(s):		150525					
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Umnožak	Odziv mreže
1	3	0,9678	1	9,2098	9	3,0294	3
2	3	4,0334	4	8,9727	9	5,9899	6
3	3	9,0602	9	8,8739	9	8,9156	9
4	3	16,1841	16	8,9087	9	11,892	12
5	3	25,0536	25	9,0741	9	14,8728	15
3	1	8,9832	9	0,7707	1	2,7627	3
3	2	9,0269	9	3,923	4	5,8458	6
3	4	9,0927	9	15,8878	16	11,9785	12
3	5	9,1631	9	24,8296	25	14,956	15
4	1	16,0368	16	0,79709	1	3,8041	4
4	2	16,1377	16	3,9493	4	7,7816	8
4	3	16,1841	16	8,9087	9	11,892	12
4	4	16,211	16	16,0566	16	16,0566	16
4	5	16,3057	16	24,8309	25	20,1418	20

Iz tablice 4.40 vidljivo je da je mreža uspjela naučiti svih 14 kombinacija sad kad joj je NRMS pomaknut s 0.03 na 0.01. Dakle, veća točnost mreže s dva ulaza i tri izlaza postiže se smanjenjem vrijednosti NRMS-a. Što je NRMS manji, točnost je veća.

Ovu tvrdnju pokušat ću dokazati pomoću sinusne funkcije. Za početak ponovno ću vratiti vrijednost NRMS-a na 0.03. Mrežu ću ispitati tako da maknem 2 kao prvi broj, tj. prvu ulaz i 4 kao drugo broj tj. drugi ulaz.

Tablica 4.40. Zadani parametri prilikom ispitivanja sinusne funkcije

Vrijednost momentuma :	0
Brzina učenja:	0.1
NRMS	0.03
Broj neurona skrivenog sloja:	5

Tablica 4.41 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže prvi broj 2 i drugi broj 4

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.42 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 2 i drugi broj 4

Sinusna funkcija							
Koraci:		1661					
Vrijeme(s):		5,9695					
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Umnožak	Odziv mreže
2	1	3,7761	4	6,85729	1	1,2017	1
2	2	3,5151	4	3,7851	4	3,5721	4
2	3	3,4003	4	9,0627	9	5,8189	6
2	4	3,4483	3	16,29	16	7,9131	8
2	5	3,6784	4	24,9078	25	9,825	10
4	1	16,1145	16	1,0452	1	3,7636	4
4	3	16,0057	16	9,0217	9	16,285	12
4	4	16,082	16	16,2883	16	16,3552	16
4	5	16,2771	16	25,0892	25	20,231	20

Dakle, mreža je krivo prikazala samo dvije vrijednosti ne učenih kombinacija brojeva. Smanjenjem nrms-a pokušat ću postići da mreža odgovori točno na svih 9 ne naučenih kombinacija.

Tablica 4.43 Zadani parametri prilikom ispitivanja sinusne funkcije s manjim NRMS-om

Vrijednost momentuma :	0
Brzina učenja:	0,1
NRMS	0,01

Broj neurona skrivenog sloja:	5
-------------------------------	---

Tablica 4.44 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže prvi broj 2 i drugi broj 4 nakon smanjenja NRMS -a s 0.03 na 0.01

Sinusna funkcija							
Koraci:		24416					
Vrijeme(s):		69,7844					
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Umnožak	Odziv mreže
2	1	3,72	4	1,0526	1	1,8479	2
2	2	3,695	4	3,8719	4	3,8162	4
2	3	3,6736	4	9,0636	9	5,8757	6
2	4	3,6736	4	16,268	16	7,9254	8
2	5	3,7189	4	24,9853	25	9,8414	10
4	1	16,1068	16	1,0649	1	3,9962	4
4	3	16,1024	16	9,0645	9	12,0879	12
4	4	16,0992	16	16,2736	16	16,1776	16
4	5	16,1272	16	25,0032	25	20,0807	20

Kao što je i pretpostavljeno, mreža je uspješno naučila svih 9 kombinacija tako da joj se mogućnost pogreške smanjila s 0.03 na 0.01. Također, bitno je zamijetiti da i kod mreže s tri izlaza broj koraka učenja raste sa smanjenjem vrijednosti NRMS-a, a s time raste i vrijeme izračuna.

### 4.3 Ispitivanje mreže s dva ulaza i četiri izlaza

Mrežu s četiri izlaza testirat ću pomoću Gaussianove funkcije budući da se ona pokazalo kao najbrža i sa najtočnijim rezultatima. Za početak mrežu ću testirati tako da iz nje izbacim sve kombinacije koje u sebi sadrže broj 3, sa vrijednošću NRMS-a 0.03. Budući da očekujem pogrešku, mrežu ću potom testirati i sa vrijednošću NRMS-a 0.01. Zatim ću pokazati utjecaj širine i centra Gaussianove funkcije na samo učenje mreže.

I kod ovog testiranja vrijednosti težina razlikuju se u odnosu na prethodne primjere, te sad matrica izlaza  $\mathbf{W}$  ima oblik  $[4 \times 6]$  gdje je 4 broj izlaza iz mreže, a 6 broj skrivenih neurona zajedno s biasom.

$$\mathbf{V} = \begin{bmatrix} -0.8482914209 & -0.7401875831 & -0.6756353836 \\ -0.8920997627 & 0.1376473217 & 0.5885690814 \\ 0.0615951060 & -0.0612187179 & -0.3775699159 \\ 0.5583344602 & -0.9761958610 & 0.0570662710 \\ 0.8680213685 & -0.3257547112 & -0.6687025410 \end{bmatrix} \quad (4.5)$$

$$W = \begin{bmatrix} 0.0944310599 & -0.7227511143 & -0.7014119889 & -0.4849834918 \\ 0.6814345120 & -0.4914356421 & 0.6285696521 & -0.5129500626 \\ 0.8585272464 & -0.3000324680 & -0.6068094991 & -0.4978322840 \\ 0.2320893523 & -0.0534223022 & -0.2966809859 & 0.6616572558 \\ 0.1705281823 & 0.0994472166 & 0.8343873277 & -0.4283219624 \\ 0.5144004582 & 0.5074581886 & -0.2391083060 & 0.1356432815 \end{bmatrix}^T \quad (4.6)$$

Parametri učenja isti su kao što su prikazani u tablici 4.40, samo što sad ispitivanje nije pomoću sinusne već pomoću Gaussianove funkcije. Nakon toga mrežu ću učiti prema parametrima iz tablice 4.43.

Tablica 4.45 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 4.46 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 za NRMS 0.03

Gaussianova funkcija									
Koraci:		3136							
Vrijeme(s):		13,2896							
Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Suma	Odziv mreže	Umnožak	Odziv mreže
1	3	0,70084	1	8,943	9	3,8406	4	2,8089	3
2	3	3,7585	4	9,0548	9	4,8375	5	5,6962	6
3	3	8,709	9	9,265	9	5,8842	6	8,925	10
4	3	15,8944	16	9,3317	9	6,9218	7	12,217	12
5	3	25,0552	25	9,2277	9	7,8906	8	15,3638	15
3	1	8,9037	9	1,0457	1	4,1545	4	2,9716	3
3	2	8,7161	9	43,903	4	4,9299	5	5,8514	6
3	4	8,8942	9	16,5739	17	6,964	7	12,021	12
3	5	9,2912	9	25,1295	25	8,1011	8	15,009	15

Tablica 4.46 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 3 za NRMS 0.01

Gaussianova funkcija	
Koraci:	46854
Vrijeme(s):	111,847

Prvi broj:	Drugi broj:	Kvadrat 1. broja	Odziv mreže	Kvadrat 2. broja	Odziv mreže	Suma	Odziv mreže	Umnožak	Odziv mreže
1	3	1,0292	1	9,066	9	4,0206	4	2,9711	3
2	3	3,952	4	8,9116	9	4,9806	5	5,9448	6
3	3	8,7518	9	8,9256	9	5,9876	6	9,9885	10
4	3	16,0272	16	8,9101	9	7,0002	7	12,0502	12
5	3	25,0267	25	9,05	9	8,0265	8	15,0746	15
3	1	8,7153	9	1,0986	1	3,9708	4	2,9122	3
3	2	8,7595	9	3,8162	4	4,9637	5	5,906	6
3	4	8,7534	9	16,151	16	7,0115	7	12,035	12
3	5	8,8178	9	25,0774	25	8,009	8	14,9692	15

Dakle, mreža s četiri izlaza također je uspjela naučiti svih 9 kombinacija brojeva tako da joj se iznos NRMS -a smanjio s 0.03 na 0.01. Vidljivo je da je prilikom promjene NRMS -a došlo do drastičnog porasta koraka učenja, koji su se popeli s 3136 na 46854 , iako je mreža trebala naučiti samo još jednu kombinaciju. Također je vidljivo kako su vrijednosti izlaza korištenjem NRMS -a 0.01 puno bliže točnijim vrijednostima u odnosu na rezultate dobivene korištenjem NRMS -a 0.03. Dakle, veća preciznost zahtjeva veći broj koraka.

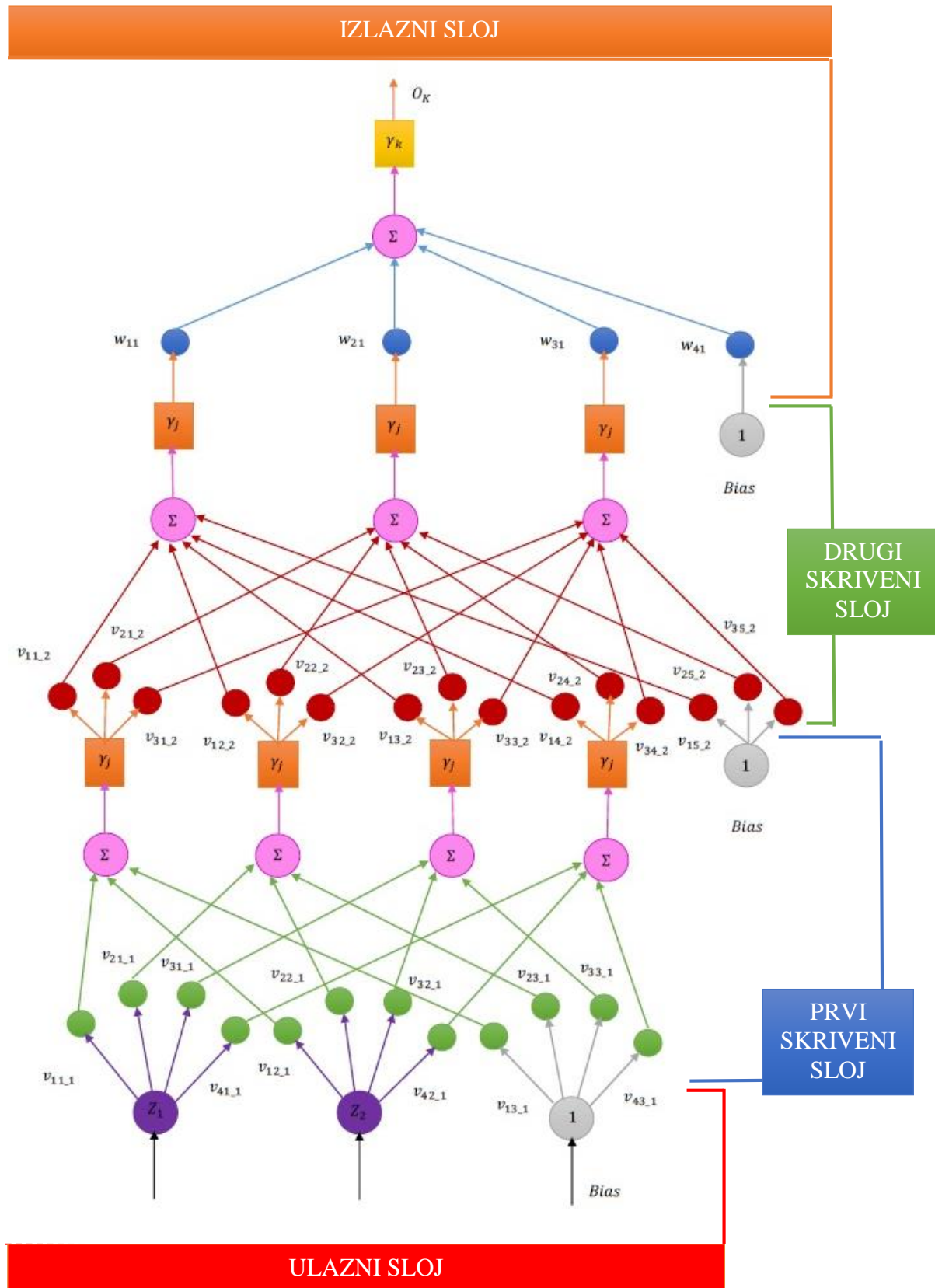
#### 4.3.1 Promjena širine i centra Gaussove funkcije

Tablica 4.47 Tablica promjena centra i širine Gaussianove funkcije

Centar funkcije	Širina funkcije	Koraci
0,5	1	1587
1	1	2407
0	1	2298
2	1	1030
1	0,7	951
1	2	3116
1	2,5	5434

Ispitivanje centra i širine Gaussianove funkcije provedeno je prema parametrima prikazanim u tablici 4.40. Vidljivo je da je najbolje kretati se sa vrijednošću centra i širine funkcije oko 1. Ako centar Gaussianove funkcije smanjujemo, dolazi do pada koraka, međutim naglo dolazi i do njihovog porasta, ukoliko uzmemo mali ili centar veći od 2. Širina Gaussianove funkcije ne može poprimiti vrijednost 0, a iz tablice je vidljivo kako njenim povećanjem dolazi i do povećanja broja koraka učenja mreže.

## 5. UNAPRIJEDNA STATIČKA DVOSLOJNA NEURONSKA MREŽA



Slika 5.1 Prikaz strukture dvoslojne neuronske mreže

## 5.1 Unaprijedna faza učenja dvoslojne neuronske mreže

Račun unaprijedne faze za višeslojnu mrežu počinje jednako kao i u slučaju jednoslojne neuronske mreže detaljno prikazane u prvom dijelu rada, u poglavlju 2.4. Dakle, iz treninga zapisnika ili seta učenja uzimaju se vrijednosti svih ulaza mreže  $\mathbf{Z}$  kako bi dobili željeni izlaz mreže  $\mathbf{O}$ . Budući da se radi o dvoslojnoj neuronskoj mreži, potrebno je odrediti vrijednost težinskog koeficijenta  $\mathbf{W}$  te vrijednosti težinskog koeficijenta  $\mathbf{V}_1$  prvog sloja i težinskog koeficijenta  $\mathbf{V}_2$  drugog sloja. Dimenzije matrica težinskih faktora i u ovom slučaju mogu se odabrati generatorom slučajnih brojeva u rasponu od 0.1 do -0.1.

### 5.1.1 Prvi sakriveni sloj

Funkcija sume  $net_{Hj1}$  neurona prvog skrivenog sloja  $H$  za dvoslojnu neuronsku mrežu dobiva se iz izraza:

$$net_{Hj1} = \sum_{i=1}^I v_{ji1} Z_i \quad j = 1, 2, \dots, J_1 - 1, \quad i = 1, 2, \dots, I \quad (5.1)$$

gdje je:

- $net_{Hj1}$  - funkcija sume neurona prvog skrivenog sloja  $H_1$
- $H_1$  - prvi sakriveni sloj
- $v_1$  - težine prvog skrivenog sloja
- $Z$  - ulazni sloj
- $J_1$  - broj neurona u prvom skrivenom sloju sa biasom
- $I$  - ulazni neuroni +1

Aktivacijske funkcije korištene u prvom skrivenom sloju analogne su aktivacijskim funkcijama korištenima kod mreže s jednim skrivenim slojem. Razlika je jedino što će se  $net_{Hj}$  razlikovati u svakom sloju.

- Unipolarna sigmoidalna funkcija: 
$$y_{j1} = \frac{1}{1 + e^{-net_{Hj1}}} \quad (5.2)$$

- Bipolarna sigmoidalna funkcija: 
$$y_{j1} = \frac{2}{1 + e^{-net_{Hj1}}} - 1 \quad (5.3)$$

- Sinusna funkcija: 
$$y_{j1} = \sin(net_{Hj1}) \quad (5.4)$$

- Gaussianova funkcija: 
$$y_{j1} = e^{-\frac{1}{2} \left( \frac{net_{Hj1} - c_{j1}}{\sigma_{j1}} \right)^2} \quad (5.5)$$



Za funkcije (5.2), (5.3), (5.4) i (5.5) vrijedi  $j = 1, 2, \dots, J_1 - 1$ , gdje je  $J_1$  broj neurona prvog sakrivenog sloja uvećan za bias. Što se tiče biasa, u sva četiri slučaja vrijedi:

$$y_{j1} = 1 \quad (5.6)$$

### 5.1.2 Drugi sakriveni sloj

Funkcija sume  $net_{Hj2}$  neurona drugog skrivenog sloja  $H$  za dvoslojnu neuronsku mrežu dobiva se iz izraza:

$$net_{Hj2} = \sum_{i=1}^I v_{ji2} y_{i1} \quad j = 1, 2, \dots, J_2 - 1, \quad i = 1, 2, \dots, J_1 \quad (5.7)$$

Gdje je:

- $net_{Hj2}$  - funkcija sume neurona drugog skrivenog sloja  $H_2$
- $H_2$  - drugi sakriveni sloj
- $v_2$  - težine drugog sakrivenog sloja
- $y_1$  - aktivacijska funkcija prvog skrivenog sloja
- $J_1$  - broj neurona u prvom sakrivenom sloju sa biasom
- $J_2$  - broj neurona u drugom sakrivenom sloju sa biasom

Aktivacijske funkcije korištene u drugom sakrivenom sloju analogne su aktivacijskim funkcijama korištenima u prvom sakrivenom sloju:

- Unipolarna sigmoidalna funkcija: 
$$y_{j2} = \frac{1}{1 + e^{-net_{Hj2}}} \quad (5.8)$$

- Bipolarna sigmoidalna funkcija: 
$$y_{j2} = \frac{2}{1 + e^{-net_{Hj2}}} - 1 \quad (5.9)$$

- Sinusna funkcija: 
$$y_{j2} = \sin(net_{Hj2}) \quad (5.10)$$

- Gaussianova funkcija: 
$$y_{j2} = e^{-\frac{1}{2} \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}} \right)^2} \quad (5.11)$$

Za funkcije (5.8), (5.9), (5.10) i (5.11) vrijedi  $j = 1, 2, \dots, J_2 - 1$ , gdje je  $J_2$  broj neurona prvog sakrivenog sloja uvećan za bias. Što se tiče biasa, u sva četiri slučaja vrijedi:

$$y_{j2} = 1 \quad (5.12)$$

### 5.1.3 Izlazni sloj

Funkcija sume *net* neurona izlaznog sloja *O* dobiva se iz izraza (5.13) analogno kao i u slučaju jednoslojne neuronske mreže, iako ovog puta u izračun ulaze neuroni i aktivacijska funkcija drugog skrivenog sloja:

$$net_{Ok} = \sum_{j=1}^J w_{kj} y_{2j} \quad j = 1, 2, \dots, J_2 - 1, \quad k = 1, 2, \dots, K \quad (5.13)$$

gdje je:

<i>net</i>	-	funkcija sume neurona izlaznog sloja <i>O</i>
<i>O</i>	-	izlazni sloj
<i>w</i>	-	težine izlaznog sloja
<i>J<sub>2</sub></i>	-	broj neurona u drugom skrivenom sloju sa biasom
<i>K</i>	-	broj izlaznih neurona

Što se tiče aktivacijske funkcije izlaznog sloja, vrijedi izraz (2.18) korišten kod jednoslojne mreže.

## 5.2 Povratna faza učenja dvoslojne neuronske mreže

Izračun povratne faze učenja kod dvoslojne neuronske mreže jednak je prikazanom postupku za jednoslojnu neuronske mreže, te vrijede izrazi (2.19), (2.20), (2.21), (2.22), (2.23), (2.24), (2.25), (2.26), (2.27) i (2.28).

### 5.2.1 Promjena težina izlaznog sloja

Promjena težina izlaznog sloja odvija se analogno kao i u slučaju jednoslojne neuronske mreže, samo što se sad uzimaju u obzir neuroni drugog skrivenog sloja te aktivacijska funkcija također drugog skrivenog sloja. Ponovno vrijede izrazi:

$$w_{kj}(n+1) = w_{kj}(n) + \alpha \Delta w_{kj}(n-1) - \beta \Delta w_{kj}(n-2) - \eta \nabla E(n) \quad (5.14)$$

$$\Delta E(n) = \frac{\partial E(n)}{\partial w_{kj}} \quad (5.15)$$

Gradijent pogreške izračunava se primjenom uzastopnih parcijalnih derivacija prema izrazu:

$$\frac{\partial E(n)}{\partial w_{kj}} = \frac{\partial E(n)}{\partial O_k} \cdot \frac{\partial O_k}{\partial net_{Ok}} \cdot \frac{\partial net_{Ok}}{\partial w_{kj}} \quad (5.16)$$

Vrijednosti pojedinih članova izraza (5.16) dobivaju se nakon parcijalnog deriviranja te glase:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k) \quad (5.17)$$

$$\frac{\partial O_k}{\partial net_{Ok}} = \gamma'_k = 1 \quad (5.18)$$

$$\frac{\partial net_{Ok}}{\partial w_{kj}} = y_{j2} \quad (5.19)$$

Uvrštavanjem izraza (5.17), (5.18), (5.19) u izraz (5.16) dobivamo:

$$\Delta E(n) = \frac{\partial E(n)}{\partial w_{kj}} = -(d_k - O_k)y_{j2} \quad (5.20)$$

Konačno, uvrštavanjem vrijednosti (5.20) u (5.14) dobiva se algoritam promjene težina izlaznog sloja:

$$w_{kj}(n+1) = w_{kj}(n) + \alpha \Delta w_{kj}(n-1) - \beta \Delta w_{kj}(n-2) + \eta (d_k - O_k)y_{j2} \quad (5.21)$$

Vrijednost  $k$  u svim navedenim izrazima jednaka je  $k = 1, 2, \dots, K$ , gdje je  $K$  broj izlaza mreže. Što se tiče vrijednosti  $j$  ona iznosi  $j = 1, 2, \dots, J_2 - 1$ , gdje je  $J_2$  broj neurona drugog skrivenog sloja s biasom.

### 5.2.2 Promjena težina drugog izlaznog sloja

Težine skrivenog sloja mijenjaju se nakon što se promijene sve težine izlaznog sloja. Prvo se mijenjaju težine drugog skrivenog sloja, a potom prvog skrivenog sloja.

$$v_{ji2}(n+1) = v_{ji2}(n) + \alpha \Delta v_{ji2}(n-1) - \beta \Delta v_{ji2}(n-2) - \eta \nabla E(n) \quad (5.22)$$

Gradijent pogreške računa se kao i kod jednoslojne neuronske mreže, prema izrazu::

$$\frac{\partial E(n)}{\partial v_{ji2}} = \frac{\partial E(n)}{\partial y_{j2}} \cdot \frac{\partial y_{j2}}{\partial net_{Hj2}} \cdot \frac{\partial net_{Hj2}}{\partial v_{ji2}} \quad (5.23)$$

Vrijednosti pojedinih članova izraza (5.23) dobivaju se parcijalnim deriviranjem, te nakon sređivanja dobivamo:

$$\begin{aligned}
\frac{\partial E(n)}{\partial y_{j2}} &= \frac{\partial E(n)}{\partial O_1} \cdot \frac{\partial O_1}{\partial net_{01}} \cdot \frac{\partial net_{01}}{\partial y_{j2}} + \\
&+ \frac{\partial E(n)}{\partial O_2} \cdot \frac{\partial O_2}{\partial net_{02}} \cdot \frac{\partial net_{02}}{\partial y_{j2}} + \\
&+ \frac{\partial E(n)}{\partial O_3} \cdot \frac{\partial O_3}{\partial net_{03}} \cdot \frac{\partial net_{03}}{\partial y_{j2}} + \\
&\quad \vdots \\
&\quad \vdots \\
&+ \frac{\partial E(n)}{\partial O_K} \cdot \frac{\partial O_K}{\partial net_{0K}} \cdot \frac{\partial net_{0K}}{\partial y_{j2}}
\end{aligned} \tag{5.24}$$

gdje je:

$$\frac{\partial E(n)}{\partial O_K} = -(d_k - O_k) \quad k = 1, 2, \dots, K \tag{5.25}$$

$$\frac{\partial O_K}{\partial net_{0K}} = 1 \quad k = 1, 2, \dots, K \tag{5.26}$$

$$\frac{\partial net_{0K}}{\partial y_{j2}} = w_{kj} \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, J_2 - 1 \tag{5.27}$$

Parametar  $J_2$  u ovom slučaju odnosi se na neurone drugog skrivenog sloja zajedno s vrijednošću biasa. Vrijednost  $K$  odnosi se na broj izlaza mreže bez biasa.

Uvrštavanjem izraza (5.25)-(5.26) u (5.27) dobiva se:

$$\frac{\partial E(n)}{\partial y_{j2}} = - \sum_{k=1}^K (d_k - O_k) w_{kj} \tag{5.28}$$

Uvrstimo li izraz (2.37) u (5.28) dobiva se konačan oblik rješenja prvoga člana izraza (5.22):

$$\frac{\partial E(n)}{\partial y_{j2}} = - \sum_{k=1}^K \delta_{ok} w_{kj} \tag{5.29}$$

Treći član izraza (5.23) nakon derivacije glasi:

$$\frac{\partial net_{Hj2}}{\partial v_{ji2}} = y_{j1} \quad (5.30)$$

Za srednji član izraza (5.23) vrijedi:

$$\frac{\partial y_{j2}}{\partial net_{Hj2}} = \gamma'_{j2} \quad (5.31)$$

Gdje je  $\gamma'_{j2}$ :

- Unipolarna sigmoidalna funkcija: 
$$\gamma'_{j2} = \frac{e^{net_{Hj2}}}{(1 + e^{net_{Hj2}})^2} \quad (5.32)$$

- Bipolarna sigmoidalna funkcija: 
$$\gamma'_{j2} = \frac{1}{2}(1 - y_{j2}^2) \quad (5.33)$$

- Sinusna funkcija: 
$$\gamma'_{j2} = \cos(net_{Hj2}) \quad (5.34)$$

- Gaussianova funkcija: 
$$\gamma'_{j2} = -y_{j2} \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}^2} \right) \quad (5.35)$$

Uvrstimo li izraze (5.30), (5.29) i (5.31) u izraz (5.23) dobivamo algoritam promjene težinskih koeficijenata drugog skrivenog sloja.

$$v_{ji2}(n+1) = v_{ji2}(n) + \alpha \Delta v_{ji2}(n-1) - \beta \Delta v_{ji2}(n-2) + \eta \gamma'_{j2} y_{j1} \sum_{k=1}^K \delta_{Ok} w_{kj} \quad (5.36)$$

Kod korištenja Gaussianove funkcije u procesu učenja ni u slučaju dvoslojne neuronske mreže nije dovoljno poznavati samo algoritam promjene težinskih faktora skrivenog sloja, već i algoritam promjene centra Gaussianove funkcije te širine Gaussove funkcije u prvom i drugom sloju neuronske mreže.

Za početak potrebno je na identičan način pomoću parcijalnih derivacija izračunati gradijente pogreške centra i širine funkcije.

Gradijent pogreške centra Gaussianove funkcije drugog skrivenog sloja dobiva se iz izraza:

$$\frac{\partial E(n)}{\partial c_{j2}} = \frac{\partial E(n)}{\partial y_{j2}} \cdot \frac{\partial y_{j2}}{\partial c_{j2}} \quad (5.37)$$

Prva vrijednost člana analogna je izrazu (5.29), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $c_{j2}$ , koja glasi:

$$\frac{\partial y_{j2}}{\partial c_{j2}} = y_{j2} \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}^2} \right) \quad (5.38)$$

Uvrste li se izrazi (5.29) i (5.38) u (5.37) dobiva se gradijent pogreške centra Gaussianove funkcije u drugom sloju neuronske mreže:

$$\frac{\partial E(n)}{\partial c_{j2}} = - \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_{j2} \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}^2} \right) \quad (5.39)$$

Analogno izrazu (5.36) dobiva se algoritam promjene težinskih koeficijenata drugog sakrivenog sloja centra Gaussianve funkcije:

$$c_{j2}(n+1) = c_{j2}(n) + \alpha \Delta c_{j2}(n-1) - \beta \Delta c_{j2}(n-2) + \eta \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_{j2} \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}^2} \right) \quad (5.40)$$

Identičan postupak primjenjuje se i u slučaju promijene širine Gaussianove funkcije, pa se gradijent promjene širine Gaussianove funkcije drugog sakrivenog sloja dobiva iz izraza:

$$\frac{\partial E(n)}{\partial \sigma_{j2}} = \frac{\partial E(n)}{\partial y_{j2}} \cdot \frac{\partial y_{j2}}{\partial \sigma_{j2}} \quad (5.41)$$

Prva vrijednost člana opet je identična izrazu (5.29), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $\sigma_{j2}$ , te glasi:

$$\frac{\partial y_{j2}}{\partial \sigma_{j2}} = y_{j2} \frac{(net_{Hj2} - c_{j2})^2}{\sigma_{j2}^3} \quad (5.42)$$

Uvrštavanjem izraza (5.29) i (5.42) u (5.41) dobiva se gradijent pogreške širine Gaussianove funkcije drugog skrivenog sloja koji glasi:

$$\frac{\partial E(n)}{\partial \sigma_{j2}} = - \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_{j2} \frac{(net_{Hj2} - c_{j2})^2}{\sigma_{j2}^3} \quad (5.43)$$

Konačno se uvrštavanjem (5.43) dobiva algoritam promjene težinskih koeficijenata drugog sakrivenog sloja širine Gaussianove funkcije:

$$\sigma_{j_2}(n+1) = \sigma_{j_2}(n) + \alpha \Delta \sigma_{j_2}(n-1) - \beta \Delta \sigma_{j_2}(n-2) + \eta \sum_{k=1}^K \delta_{Ok} w_{kj} \cdot y_{j_2} \left( \frac{\text{net}_{Hj_2} - c_{j_2}}{\sigma_{j_2}^2} \right) \quad (5.44)$$

### 5.2.3 Promjena težina prvog izlaznog sloja

Težine prvog skrivenog sloja mijenjaju se nakon težina drugog skrivenog sloja, budući da se povratna faza učenja odvija od izlaznog prema ulaznom sloju.

$$v_{ji1}(n+1) = v_{ji1}(n) + \alpha \Delta v_{ji1}(n-1) - \beta \Delta v_{ji1}(n-2) - \eta \nabla E(n) \quad (5.45)$$

Gradijent pogreške računa se prema izrazu jednakom kao i kod drugog sloja, koji glasi:

$$\frac{\partial E(n)}{\partial v_{ji1}} = \frac{\partial E(n)}{\partial y_{j_1}} \cdot \frac{\partial y_{j_1}}{\partial \text{net}_{Hj_1}} \cdot \frac{\partial \text{net}_{Hj_1}}{\partial v_{ji1}} \quad (5.46)$$

Vrijednosti pojedinih članova izraza (5.46) dobivaju se parcijalnim deriviranjem, te nakon sređivanja dobivamo:

$$\begin{aligned} \frac{\partial E(n)}{\partial y_{j_1}} &= \frac{\partial E(n)}{\partial O_1} \cdot \frac{\partial O_1}{\partial \text{net}_{O1}} \cdot \frac{\partial \text{net}_{O1}}{\partial y_{j_2}} \cdot \frac{\partial y_{j_2}}{\partial \text{net}_{Hj_2}} \cdot \frac{\partial \text{net}_{Hj_2}}{\partial y_{j_1}} + \\ &+ \frac{\partial E(n)}{\partial O_2} \cdot \frac{\partial O_2}{\partial \text{net}_{O2}} \cdot \frac{\partial \text{net}_{O2}}{\partial y_{j_2}} \cdot \frac{\partial y_{j_2}}{\partial \text{net}_{Hj_2}} \cdot \frac{\partial \text{net}_{Hj_2}}{\partial y_{j_1}} + \\ &+ \frac{\partial E(n)}{\partial O_3} \cdot \frac{\partial O_3}{\partial \text{net}_{O3}} \cdot \frac{\partial \text{net}_{O3}}{\partial y_{j_2}} \cdot \frac{\partial y_{j_2}}{\partial \text{net}_{Hj_2}} \cdot \frac{\partial \text{net}_{Hj_2}}{\partial y_{j_1}} + \\ &\quad \vdots \\ &+ \frac{\partial E(n)}{\partial O_K} \cdot \frac{\partial O_K}{\partial \text{net}_{Ok}} \cdot \frac{\partial \text{net}_{Ok}}{\partial y_{j_2}} \cdot \frac{\partial y_{j_2}}{\partial \text{net}_{Hj_2}} \cdot \frac{\partial \text{net}_{Hj_2}}{\partial y_{j_1}} \end{aligned} \quad (5.47)$$

gdje je:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k) \quad k = 1, 2, \dots, K \quad (5.48)$$

$$\frac{\partial O_K}{\partial net_{OK}} = 1 \quad k = 1, 2, \dots, K \quad (5.49)$$

$$\frac{\partial net_{OK}}{\partial y_{j2}} = w_{kj} \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, J_2 - 1 \quad (5.50)$$

$$\frac{\partial y_{j2}}{\partial net_{Hj2}} = \gamma'_{j2} \quad j = 1, 2, \dots, J_2 - 1 \quad (5.51)$$

$$\frac{\partial net_{Hj2}}{\partial y_{j1}} = v_{ji2} \quad j = 1, 2, \dots, J_2 - 1, \quad i = 1, 2, \dots, J_1 \quad (5.52)$$

Parametar  $J_2$  u ovom slučaju odnosi se na neurone drugog skrivenog sloja zajedno s vrijednošću biasa, a paramterar  $J_1$  na neurone prvog skrivenog sloja zajedno sa biasom, dok se vrijednost  $K$  odnosi se na broj izlaza mreže bez biasa.

Uvrštavanjem izraza (5.48)-(5.52) u (5.47) dobiva se:

$$\frac{\partial E(n)}{\partial y_{j2}} = - \sum_{k=1}^K (d_k - O_k) w_{kj} \gamma'_{j2} v_{ji2} \quad (5.53)$$

Treći član izraza (5.46) nakon derivacije glasi:

$$\frac{\partial net_{Hj1}}{\partial v_{ji1}} = Z_i \quad (5.54)$$

Za srednji član izraza (5.46) vrijedi:

$$\frac{\partial y_{j1}}{\partial net_{Hj1}} = \gamma'_{j1} \quad (5.55)$$

gdje je  $\gamma'_{j1}$ :

- Unipolarna sigmoidalna funkcija: 
$$\gamma'_{j1} = \frac{e^{net_{Hj2}}}{(1 + e^{net_{Hj2}})^2} \quad (5.56)$$

- Bipolarna sigmoidalna funkcija: 
$$\gamma'_{j1} = \frac{1}{2}(1 - y_{j2}^2) \quad (5.57)$$

- Sinusna funkcija: 
$$\gamma'_{j1} = \cos(net_{Hj2}) \quad (5.58)$$

- Gaussianova funkcija: 
$$\gamma'_{j1} = -y_j \left( \frac{net_{Hj2} - c_{j2}}{\sigma_{j2}^2} \right) \quad (5.59)$$



Uvrstimo li izraze (5.54), (5.53) i (5.55) u izraz (5.46) te potom u (5.45) dobivamo algoritam promjene težinskih koeficijenata prvog sakrivenog sloja.

$$v_{ji1}(n+1) = v_{ji1}(n) + \alpha \Delta v_{ji1}(n-1) - \beta \Delta v_{ji1}(n-2) + \eta \gamma'_{j1} Z_i \sum_{k=1}^K \delta_{Ok} w_{kj} \gamma'_{j2} v_{ji2} \quad (5.60)$$

Potrebno je još izračunati i algoritam promjene centra Gaussianove funkcije te širine Gaussove funkcije u prvom sloju neuronske mreže.

Gradijent pogreške centra Gaussianove funkcije prvog sakrivenog sloja dobiva se iz izraza:

$$\frac{\partial E(n)}{\partial c_{j1}} = \frac{\partial E(n)}{\partial y_{j1}} \cdot \frac{\partial y_{j1}}{\partial c_{j1}} \quad (5.61)$$

Prva vrijednost člana analogna je izrazu (5.53), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $c_{j1}$ , koja glasi:

$$\frac{\partial y_{j1}}{\partial c_{j1}} = y_{j1} \left( \frac{net_{Hj1} - c_{j1}}{\sigma_{j1}^2} \right) \quad (5.62)$$

Uvrste li se izrazi (5.61) i (5.53) u (5.61) dobiva se gradijent pogreške centra Gaussianove funkcije u prvom sloju neuronske mreže:

$$\frac{\partial E(n)}{\partial c_{j1}} = - \sum_{k=1}^K \delta_{Ok} w_{kj} \gamma'_{j2} v_{ji2} y_{j1} \left( \frac{net_{Hj1} - c_{j1}}{\sigma_{j1}^2} \right) \quad (5.63)$$

Analogno izrazu (5.60) dobiva se algoritam promjene težinskih koeficijenata prvog sakrivenog sloja centra Gaussianve funkcije:

$$c_{j1}(n+1) = c_{j1}(n) + \alpha \Delta c_{j1}(n-1) - \beta \Delta c_{j1}(n-2) + \eta \sum_{k=1}^K \delta_{Ok} w_{kj} \gamma'_{j2} v_{ji2} y_{j1} \left( \frac{net_{Hj1} - c_{j1}}{\sigma_{j1}^2} \right) \quad (5.64)$$

Gradijent promjene širine Gaussianove funkcije prvog sakrivenog sloja dobiva iz izraza:

$$\frac{\partial E(n)}{\partial \sigma_{j1}} = \frac{\partial E(n)}{\partial y_{j1}} \cdot \frac{\partial y_{j1}}{\partial \sigma_{j1}} \quad (5.65)$$

Prva vrijednost člana opet je identična izrazu (5.53), dok je drugi član parcijalna derivacija Gaussianove funkcije po  $\sigma_{j1}$ , te glasi:

$$\frac{\partial y_{j1}}{\partial \sigma_{j1}} = y_{j1} \frac{(net_{Hj1} - c_{j1})^2}{\sigma_{j1}^3} \quad (5.66)$$

Uvrštavanjem izraza (5.53) i (5.66) u (5.65) dobiva se gradijent pogreške širine Gaussianove funkcije drugog skrivenog sloja koji glasi:

$$\frac{\partial E(n)}{\partial \sigma_{j1}} = - \sum_{k=1}^K \delta_{Ok} w_{kj} \gamma'_{j2} v_{ji2} y_{j1} \frac{(net_{Hj1} - c_{j1})^2}{\sigma_{j1}^3} \quad (5.67)$$

Algoritam promjene težinskih koeficijenata prvog sakrivenog sloja širine Gaussianove funkcije glasi:

$$\sigma_{j1}(n+1) = \sigma_{j1}(n) + \alpha \Delta \sigma_{j1}(n-1) - \beta \Delta \sigma_{j1}(n-2) + \eta \sum_{k=1}^K \delta_{Ok} w_{kj} \gamma'_{j2} v_{ji2} y_{j1} \frac{(net_{Hj1} - c_{j1})^2}{\sigma_{j1}^3} \quad (5.68)$$

## 6. ISPITIVANJE RADA DVOSLOJNE NEURONSKE MREŽE

Grafičko sučelje dvoslojne neuronske mreže u odnosu na jednoslojnu mrežu razlikuje se jedino po prozoru za učenje neuronske mreže. Kod dvoslojne neuronske mreže potrebno je upisati broj neurona prvog sakrivenog sloja te broj neurona drugog sakrivenog sloja. Aktivacijska funkcija koju odaberemo u prozoru za učenje jednaka je u oba sloja mreže. U slučaju odabira Gaussianove funkcije potrebno je unijeti centar i širinu funkcije u prvom i u drugom sloju mreže.

The screenshot shows a web-based interface for configuring a neural network. At the top, a red bar contains the title 'Definiranje parametara učenja'. Below this, there are several sections:

- Neuron Count:** 'Broj ulaznih neurona: 2' and 'Broj izlaznih neurona: 1'.
- Function Selection:** 'Odabir funkcije' is set to 'Gaussianova'. 'Odabir težine' is set to 'Proizvoljne'. 'Odabir momentuma' is set to 'Bez momentuma'.
- Layer Parameters:**
  - Prvi sloj:** 'Centar funkcije: 1', 'Širina funkcije: 1', 'Broj neurona prvog skrivenog sloja: 2'.
  - Drugi sloj:** 'Centar funkcije: 1', 'Širina funkcije: 1', 'Broj neurona drugog skrivenog sloja: 3'.
  - Iznos momentuma:** 0.
- Learning Parameters:** 'Upišite brzinu učenja: 0.1', 'Upišite željeni NRMS: 0.03', 'Upišite broj koraka učenja: 5000'.
- Učenje mreže:** 'Bez grafičkog prikaza'.
- Buttons:** 'UČENJE' (orange) and 'ISPIS' (yellow).

Slika 6.1 Grafičko sučelje za definiranje parametara učenja dvoslojne neuronske mreže sa dva ulazna i jednim izlaznim neuronom

Budući da mreža radi sa dva neurona sakrivenog sloja, u procesu učenja koristit će dvije matrice težinskih koeficijenta  $V_1$  i  $V_2$ , te matrica  $W$  vezana za neurone izlaznog sloja. Dimenzija matrice  $V_1$  ovisi o broju ulaznih neurona te broju neurona sakrivenog sloja. Ukoliko je broj neurona prvog sakrivenog sloja 4, matrica  $V_1$  biti će dimenzija  $[4 \times 3]$ , gdje se broj redaka odnosi na broj neurona sakrivenog sloja, a broj stupaca na broj ulaznih neurona uvećanih za bias. Što se tiče matrice  $V_2$ , ukoliko uzmemo 6 sakrivenih neurona, dimenzija matrice biti će  $[6 \times 5]$ , gdje se broj redaka matrice odnosi na broj neurona drugog sakrivenog sloja bez biasa, a broj stupaca na broj neurona prvog sakrivenog sloja uvećanog za bias. Matrica  $W$  u ovom slučaju imat će dimenzije  $[1 \times 7]$ , gdje se jedan odnosi na broj neurona izlaznog sloja, dok se broj stupaca odnosi na broj neurona drugog sakrivenog sloja uvećanog za bias.

## 6.1 Testiranje dvoslojne mreže sa dva ulaza i jednim izlazom

Pretpostavka prije testiranja jest da će mreža sa dva sakrivena sloja davati približno podjednake rezultate kao i mreža sa jednim sakrivenim slojem, ali da će zbog složenijeg algoritma učenja sam proces biti znatno sporiji te da će se odvijati u više koraka.

Kao i u slučaju jednoslojne neuronske mreže, na korake i vrijeme učenja imat će utjecaj i raspodjela parametara unutar matrica  $V_1$ ,  $V_2$  i  $W$ , koji će se i u ovom slučaju kretati u rasponu od 0.1 do -0.1. Bitno je napomenuti da matrice  $V_1$  i  $V_2$  nikad ne mogu biti jednake, ni po parametrima ni po dimenzijama ni ako odaberemo jednak broj neurona u prvom i u drugom sakrivenom sloju mreže.

Za početak mrežu ću testirati tako da iz nje izbacim sve kombinacije koje u sebi sadrže broj 3.

Tablica 6.1 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže broj 3 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 6.2 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona prvog sakrivenog sloja:	5
Broj neurona drugog sakrivenog sloja:	5

Tablica 6.3 Prikaz rezultata testiranja za 9 ne učenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		8222		10794		2265		4797	
Vrijeme (s):		35,0545		46,6307		10,2038		31,1701	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
1	3	3,1885	3	3,1151	3	2,8103	3	3,172	3
2	3	6,1996	6	6,073	6	5,813	6	5,8866	6
3	1	2,9184	3	3,3538	3	2,6222	3	3,0595	3

3	2	6,2769	6	6,2885	6	5,7329	6	6,0368	6
3	3	9,3667	9	9,1991	9	8,9069	9	9,1657	9
3	4	12,1887	12	12,0217	12	12,0528	12	12,191	12
3	5	14,7705	15	14,7644	15	15,0911	15	15,0631	15
4	3	12,3483	12	12,181	12	12,1225	12	12,3181	12
5	3	15,0667	15	14,8004	15	15,3944	15	14,903	15

Iz tablice 6.3 vidljivo je da je dvoslojna mreža podjednako dobro izračunala vrijednosti 9 kombinacija brojeva koje nije učila, što je bilo i u slučaju jednoslojne neuronske mreže. Kao što je pretpostavljeno, mreži je sad trebalo daleko više vremena kako bi naučila 16 kombinacija brojeva, te više koraka što je, kako je već spomenuto, ovisno i o raspodijeli i vrijednosti parametara unutar matrica težinskih koeficijenata. Iz ovih rezultata može se zaključiti da su i dvoslojna i jednoslojna mreža podjednako dobre u izvršavanju ovog tipa zadatka. Također, vidljivo je da je sinusna funkcija i u slučaju dvoslojne neuronske mreže znatno brža u odnosu na druge aktivacijske funkcije.

Tablica 6.4 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona prvog sakrivenog sloja:	3
Broj neurona drugog sakrivenog sloja:	6

Tablica 6.5 Prikaz rezultata testiranja za 9 ne učenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		2484		62659		985		7763	
Vrijeme (s):		12,9841		246,1306		5,1822		46,4135	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
1	3	2,9946	3	3,1892	3	2,6149	3	3,1535	3
2	3	5,9092	6	5,8235	6	5,8661	6	6,0318	6
3	1	3,0768	3	3,1123	3	3,0395	3	3,0725	3
3	2	5,6951	6	5,7301	6	5,9168	6	5,6987	6
3	3	8,8361	9	8,6848	9	9,094	9	8,7906	9
3	4	12,1435	12	11,9077	12	12,3004	12	12,0174	12
3	5	15,0665	15	15,077	15	15,2523	15	14,9461	15
4	3	11,782	12	11,7535	12	12,24	12	11,7327	12
5	3	14,7562	15	14,9611	15	15,2763	15	14,8899	15

Tablica 6.6 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona prvog sakrivenog sloja:	6
Broj neurona drugog sakrivenog sloja:	3

Tablica 6.7 Prikaz rezultata testiranja za 9 ne učenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		8984		10178		882		1339	
Vrijeme (s):		41,4287		41,4084		4,3446		12,5322	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
1	3	2,7538	3	2,8901	3	3,1739	3	2,4332	2
2	3	5,6026	6	5,6055	6	5,7987	6	5,75	6
3	1	3,2835	3	3,0914	3	2,5922	3	3,3959	3
3	2	5,7241	6	5,6438	6	5,7194	6	6,1233	6
3	3	8,5806	9	8,4684	9	8,7769	9	9,0873	9
3	4	11,7487	12	11,6169	12	11,842	12	12,0397	12
3	5	15,0173	15	15,0943	15	14,9664	15	14,8254	15
4	3	11,7411	12	11,5608	12	11,9787	12	12,176	12
5	3	15,0335	15	14,909	15	15,2407	15	14,0561	15

U tablici 6.5 i 6.7 prikazani su rezultati učenja mreže također za 9 ne učenih kombinacija, sa razlikom u broju neurona prvog i drugog sakrivenog sloja. U slučaju prikazanom u tablici 6.5 zadano je 3 neurona u prvom i 6 neurona u drugom sakrivenom sloju. Vidljivo je da je mreža uspješno izračunala svih 9 kombinacija koje nije učila. Sinusna funkcija opet se pokazala najbržom, dok je kombinacija matrica težinskih faktora bila loš odabir za unipolarnu sigmoidalnu funkciju. Bitno je spomenuti da su kod svih aktivacijskih funkcija korištene matrice težinskih faktora jednakih parametara, kao i u slučaju testiranja jednoslojne neuronske mreže. Dakle, jasno se vidi kako jednaka raspodjela parametara može različito utjecati na učenje primjenom različitih aktivacijskih funkcija.

U tablici 6.7 vidljivi su rezultati ne učenih kombinacija brojeva nakon što su zamijenjene matrice  $V_1$  i  $V_2$ , tj. matrica  $V_1$  iz prethodnog primjera postaje  $V_2$  i obratno. Iz rezultata je vidljiva prva pogreška u izračunu, za slučaj Gaussianove funkcije, pa se može zaključiti da je ovakva raspodjela parametara težinskih faktora bila loša za navedenu funkciju. S druge strane, sinusnoj funkciji sad je trebalo daleko manje koraka i vremena kako bi naučila svih 16 kombinacija brojeva i dala 9 točnih vrijednosti za ne učene kombinacije.

Pretpostavka je da će i dvoslojna mreža imati poteškoća sa učenjem kombinacija brojeva koji u sebi sadrže 1 i 5, zato što nije naučila ni jednu kombinaciju prije 1 ni nakon 5, pa je potrebno provesti testiranje.

Tablica 6.8 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 1 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 6.9 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1

Bipolarna sigmoidalna funkcija					
Vrijednost momentuma:			0		
Brzina učenja:			0,1		
NRMS			0,03		
Koraci:			3073	8363	
Vrijeme(s):			13,9866	35,0304	
Broj neurona prvog sakrivenog sloja:			5	3	
Broj neurona drugog sakrivenog sloja:			5	6	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
1	1	0,31895	0	0,97309	1
1	2	1,2365	1	2,3321	2
1	3	2,2036	2	3,7253	4
1	4	3,2792	3	5,2087	5
1	5	4,5467	5	6,8268	7
2	1	1,919	2	2,1444	2
3	1	3,3156	3	3,2638	3
4	1	4,6041	5	4,2975	4
5	1	5,8772	6	5,3332	5

Kao što je pretpostavljeno, dvoslojna mreža ima problema sa izračunom kombinacija koje u sebi sadrže broj 1 zbog toga što nije naučila ništa što se nalazi ispred 1. U ovom slučaju mreža neće uspjeti naučiti sve kombinacije ni ako bi se vrijednost NRMS-a smanjila na 0.01 ili manje, već bi i dalje pokazivala određenu količinu pogrešaka.

Tablica 6.10 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 5 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 6.11 Prikaz rezultata testiranja za 9 ne naučenih kombinacija koje u sebi sadrže broj 1

Gaussianova funkcija					
Vrijednost momentuma:			0		
Brzina učenja:			0,1		
NRMS			0,03		
Koraci:			5738	5117	
Vrijeme(s):			35,8004	29,5052	
Broj neurona prvog sakrivenog sloja:			5	3	
Broj neurona drugog sakrivenog sloja:			5	6	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
5	1	5,5736	6	4,4756	4
5	2	9,9925	10	8,6659	9
5	3	15,0505	15	13,8067	14
5	4	19,6023	20	18,6705	19
5	5	22,9861	23	22,3956	22
1	5	5,6592	6	4,772	5
2	5	9,7316	10	9,2614	9
3	5	14,4026	14	14,361	14
4	5	18,9856	19	19,0906	19

Kao što je bilo i u slučaju jednoslojne neuronske mreže, mreža sa dva sakrivena sloja daje loše rezultate za kombinacije brojeva koji u sebi sadrže broj 5. Razlog je isti kao i u slučaju izostavljanja broja 1 iz mogućih kombinacija, a to je taj što mreža nije naučila ni jedan broj nakon broja 4. Vidljivo je i više netočnih kombinacija u slučaju odabira mreže sa 3 i 6 neurona u sakrivenim slojevima. Razlog ponovo može biti lošija slučajna raspodjela parametara unutar matrica težinskih faktora, zbog čega je mreža sa po 5 neurona sakrivenog sloja bila točnija.



Tablica 6.12 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 2 i 4 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 6.13 Parametri učenja neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona prvog sakrivenog sloja:	4
Broj neurona drugog sakrivenog sloja:	2

Tablica 6.14 Prikaz rezultata testiranja za 16 ne učenih kombinacija

		Bipolarna sigmoidalna funkcija		Unipolarna sigmoidalna funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		13115		10812		396		881	
Vrijeme (s):		34,1324		27,0603		1,9818		4,8916	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	1	2,1608	2	2,0046	2	1,7771	2	1,8675	2
2	2	3,7437	4	4,1098	4	3,4907	3	3,7158	4
2	3	5,5787	6	6,0212	6	5,6159	6	5,7618	6
2	5	9,9994	10	9,7025	10	9,7106	10	10,0172	10
4	1	3,8842	4	4,1147	4	3,755	4	3,7775	4
4	2	7,545	8	7,9545	8	7,772	8	7,8861	8
4	3	11,7531	12	11,9231	12	12,1689	12	12,2615	12
4	4	16,1452	16	15,9044	16	16,4664	16	16,5619	17
4	5	20,3271	20	19,8693	20	20,2667	20	20,4595	20
1	2	2,0416	2	1,9966	2	1,9739	2	1,8675	2
3	2	5,544	6	6,0963	6	5,5048	6	5,7618	6
5	2	9,8583	10	9,7748	10	9,9582	10	10,0172	10
1	4	3,7712	4	4,1928	4	4,1524	4	3,7385	4
3	4	11,88	12	11,7843	12	12,0557	12	12,0608	12
5	4	20,272	20	20,0651	20	20,5052	21	20,5569	21

Iz tablice 6.14 vidljivo je da mreža ne uspijeva točno izračunati sve kombinacije koje nisu učene u slučaju sinusne i Gaussianove funkcije. Zbog dobro poklopljenih vrijednosti težina sakrivenih slojeva mreža uspijeva dobiti sve dobre rezultate za slučaj bipolarne i unipolarne sigmoidalne funkcije. Budući da se ovog puta radi o čak 16 kombinacija koje mreža nije učila, poželjno je

manjiti vrijednost NRMS-a kako bi rezultati bili točniji i mogućnost pogreške se smanjila. U ovom primjeru, za razliku od prethodnih u kojima mreža nije učila brojeve 1 i 5, te brojeve koji se nalaze ispred i iza njih, mreža bi trebala računati preciznije smanjenjem NRMS-a.

Tablica 6.15 Parametri učenja dvoslojne neuronske mreže s dva ulaza i jednim izlazom

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,01
Broj neurona prvog sakrivenog sloja:	4
Broj neurona drugog sakrivenog sloja:	2

Tablica 6.16 Prikaz rezultata testiranja za 16 ne učenih kombinacija sa različitim vrijednostima NRMS-a

		Sinusna funkcija		Gaussianova funkcija		Sinusna funkcija		Gaussianova funkcija	
Koraci:		396		881		1056		3118	
Vrijeme (s):		1,9818		4,8916		3,5238		13,7465	
NRMS:		0.03		0.03		0.01		0.01	
Prvi broj:	Drugi broj:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	1	1,7771	2	1,8675	2	1,8929	2	2,0254	2
2	2	3,4907	3	3,7158	4	3,634	4	3,8681	4
2	3	5,6159	6	5,7618	6	5,7406	6	5,777	6
2	5	9,7106	10	10,0172	10	9,7406	10	9,6032	10
4	1	3,755	4	3,7775	4	3,9834	4	3,8731	4
4	2	7,772	8	7,8861	8	7,8825	8	7,8983	8
4	3	12,1689	12	12,2615	12	12,2182	12	12,1714	12
4	4	16,4664	16	16,5619	17	16,4966	16	16,4359	16
4	5	20,2667	20	20,4595	20	20,302	20	20,4013	20
1	2	1,9739	2	1,8675	2	1,8493	2	1,9649	2
3	2	5,5048	6	5,7618	6	5,724	6	5,8865	6
5	2	9,9582	10	10,0172	10	9,8182	10	9,8659	10
1	4	4,1524	4	3,7385	4	4,0704	4	3,8986	4
3	4	12,0557	12	12,0608	12	12,1899	12	12,0912	12
5	4	20,5052	21	20,5569	21	20,3483	20	20,269	20

Kao što je pretpostavljeno, u tablici 6.16 vidljivo je da je mreža uspješno izračunala vrijednosti svih 16 kombinacija brojeva koje prethodno nije učila. Dovoljno je bilo smanjiti NRMS na vrijednost od 0.01 kako bi se rezultati poboljšali. Kao što je već spomenuto, razlog je u tome što je mreža naučila kombinacije brojeva 1, 3 i 5. Uz to, presudan je i utjecaj težina sakrivenog sloja.

Prema tablici 6.16 vidljivo je kao i u slučaju jednoslojne neuronske mreže da se povećanjem NRMS-a broj koraka za iste vrijednosti težina sakrivenog sloja povećao. Povećanjem koraka učenja došlo je i do povećanja vremena učenja.

Bitno je nakon ovih testiranja primijetiti da, iako mreža ima jedan sloj više, vrijeme učenja i koraci ne moraju biti nužno duži i veći u odnosu na jednoslojnu neuronsku mrežu. Konačan rezultat više će ovisiti o slučajnoj raspodijeli vrijednosti unutar matrica težinskih faktora. Povećanje i smanjenje koraka učenja možemo pratiti u slučajevima kad koristimo postojeće vrijednosti matrica težinskih faktora.

Tablica 6.17 Prikaz utjecaja promjene neurona u sakrivenim slojevima na korake i vrijeme učenja

Sinusna funkcija			
Broj neurona u prvom sakrivenom sloju	Broj neurona u drugom sakrivenom sloju	KORACI	VRIJEME
5	5	2255	20,5494
4	6	2134	17,615
6	4	2844	23,2798
7	7	986	9,79
7	7	1475	14,6205
7	7	1663	16,1526
7	7	698	9,9843

Tablica 6.17 prikazuje kako slučajno generiranje matrica težina sakrivenih slojeva na temelju broja neurona u oba sloja utječe na vrijeme i korake učenja. Prema rezultatima ukoliko jednom promijenimo broj neurona nekog sakrivenog sloja nikad više nećemo dobiti istu vrijednost matrica težina i jednake korake učenja. Što se tiče trajanja procesa učenja, jasno je da na vrijeme utječu koraci učenja, pa tako povećanjem koraka učenja rast će i vrijeme učenja mreže.

## 6.2 Testiranje dvoslojne mreže sa dva ulaza i četiri izlaza

Kao što je vidljivo u prikazanim primjerima jednoslojne neuronske mreže, povećanjem izlaza mreže povećava se vjerojatnost pojave krivih rezultata testiranja. Tako je kod mreže sa 4 izlaza greška u izračunu nastupila već prilikom izbacivanja 9 kombinacija brojeva iz procesa učenja.

U nastavku mreža će biti ispitana tako da će iz nje biti izbačene sve kombinacije koje sadrže brojeve 2 i 4.

Tablica 6.18 Prikaz izbačenih kombinacija brojeva koje u sebi sadrže 2 i 4 iz procesa učenja mreže

Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj	Prvi broj	Drugi broj
1	1	2	1	3	1	4	1	5	1
1	2	2	2	3	2	4	2	5	2
1	3	2	3	3	3	4	3	5	3
1	4	2	4	3	4	4	4	5	4
1	5	2	5	3	5	4	5	5	5

Tablica 6.19 Parametri učenja neuronske mreže s dva ulaza i četiri izlaza

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,03
Broj neurona prvog sakrivenog sloja:	5
Broj neurona drugog sakrivenog sloja:	5

Tablica 6.20 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Gaussianove funkcije

Gaussianova funkcija									
Koraci:		2032							
Vrijeme(s):		13,3566							
Prvi broj:	Drugi broj:	Kvadrat 1. broja		Kvadrat 2. broja		Suma		Umnožak	
		Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	1	4,2749	4	0,91004	1	2,9813	3	2,1163	2
2	2	3,984	4	3,8506	4	3,8018	4	3,8943	4
2	3	3,8226	4	8,9374	9	4,804	5	5,8228	6
2	5	4,3767	4	25,1106	25	6,9908	7	9,4655	9
4	1	15,8609	16	0,66237	1	4,905	5	4,0541	4
4	2	16,1555	16	3,7835	4	5,8589	6	7,7918	8
4	3	16,2744	16	8,9127	9	6,9051	7	12,2365	12
4	4	16,3076	16	16,2468	16	8,0731	8	16,7313	17
4	5	16,3575	16	25,0245	25	9,0733	9	20,5583	21
1	2	0,86849	1	4,1197	4	3,0141	3	1,9194	2
3	2	8,8723	9	3,7075	4	4,7661	5	5,9189	6
5	2	25,3909	25	4,203	4	6,9701	7	9,5132	10
1	4	0,56281	1	16,4028	16	4,9905	5	3,9817	4
3	4	8,9924	9	16,2855	16	6,9652	7	12,1523	12
5	4	25,4607	25	16,381	16	9,0954	9	20,4169	20

Tablica 6.21 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Bipolarne sigmoidalne funkcije

Bipolarna sigmoidalna funkcija									
Koraci:		1827							
Vrijeme(s):		7,9896							
Prvi broj:	Drugi broj:	Kvadrat 1. broja		Kvadrat 2. broja		Suma		Umnožak	
		Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	1	4,2454	4	0,88817	1	3,0321	3	2,0312	2
2	2	4,0539	4	4,0909	4	3,8622	4	4,1703	4
2	3	3,9784	4	9,0458	9	4,8037	5	5,9974	6

2	5	4,1719	4	24,9625	25	6,9499	7	9,3757	9
4	1	16,435	16	1,1283	1	4,9622	5	3,9143	4
4	2	16,392	16	4,1283	4	5,8594	6	8,3128	8
4	3	16,389	16	8,953	9	6,8572	7	12,5876	13
4	4	16,4497	16	15,9894	16	7,9362	8	16,5913	17
4	5	16,5968	16	25,0892	25	9,065	9	20,2361	20
1	2	0,6203	1	4,1468	4	3,1172	3	2,0445	2
3	2	9,2155	9	4,1219	4	4,7956	5	6,3415	6
5	2	24,9416	25	4,0815	4	6,9457	7	9,8151	10
1	4	0,54761	1	16,1344	16	5,0358	5	4,053	4
3	4	9,2442	9	16,0524	16	6,8376	7	12,036	12
5	4	24,9021	25	15,8844	16	8,9956	9	20,2208	20

Iz tablica 6.20 i 6.21 vidljivo je da u slučaju mreže s četiri izlaza, iako mreža nauči brojeve ispred i iza 2 i 4, mreža ipak daje krive rezultate u četvrtom izlazu i u slučaju bipolarne sigmoidalne i u slučaju Gaussianove funkcije. Pretpostavka je da i unipolarna sigmoidalna funkcija daje loše rezultate, a da sinusna funkcija zbog malog koraka učenja daje najviše krivih rješenja za kombinacije koje mreža nije učila. Moguće rješenje ovog problema je smanjenje NRMS-a sa 0.03 na 0.01.

Tablica 6.22 Parametri učenja neuronske mreže s dva ulaza i četiri izlaza

Vrijednost momentuma:	0
Brzina učenja:	0,1
NRMS	0,01
Broj neurona prvog sakrivenog sloja:	5
Broj neurona drugog sakrivenog sloja:	5

Tablica 6.23 Prikaz rezultata testiranja za 16 ne učenih kombinacija za slučaj Bipolarne sigmoidalne funkcije

Unipolarna sigmoidalna funkcija									
Koraci:		7239							
Vrijeme(s):		25,1596							
Prvi broj:	Drugi broj:	Kvadrat 1. broja		Kvadrat 2. broja		Suma		Umnožak	
		Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:	Stvarna vrijednost odziva:	Odziv mreže:
2	1	4,0261	4	0,8798	1	2,8646	3	2,0631	2
2	2	3,9186	4	4,125	4	3,7894	4	3,8845	4
2	3	3,935	4	9,0177	9	4,8479	5	5,6865	6
2	5	4,0188	4	25,0573	25	6,9182	7	9,4688	9
4	1	16,5132	17	1,0434	1	5,077	5	3,7967	4
4	2	16,3382	16	4,1187	4	6,0332	6	7,9473	8
4	3	16,2731	16	8,8962	9	7,1078	7	12,3284	12
4	4	16,2536	16	15,8876	16	8,1853	8	16,6003	17

4	5	16,1931	16	24,9854	25	9,1509	9	20,4869	20
1	2	0,89764	1	4,2668	4	2,962	3	1,9753	2
3	2	8,9892	9	4,0946	4	4,8872	5	5,9706	6
5	2	24,985	25	4,1375	4	6,9677	7	9,771	10
1	4	1,0445	1	16,0605	16	5,0287	5	3,8483	4
3	4	8,9485	9	15,9372	16	7,0613	7	12,0814	12
5	4	25,1041	25	15,9328	16	9,0534	9	20,244	20

Iz tablice 6.23 vidljivo je da smanjenje NRMS-a sa 0.03 na 0.01 ne znači i eliminacija pogreške, tj. ne garantira da će mreža izbacivati točne vrijednosti kombinacija koje nije učila, kao što je bio slučaj kod mreže s jednim izlaznim slojem. Uz to, greška u izračunu pojavila se i u četvrtom i u prvom izlazu iz mreže.

Prema rezultatima dobivenima testiranjem mreža sa četiri izlaza daje znatno lošije rezultate u odnosu na mrežu sa jednim izlazom jer može točno izračunati više kombinacija brojeva koje prethodno nije naučila. Razlog je u strukturi mreže koja je kod četiri izlazna sloja bitno složenija za izračun u odnosu na mrežu da jednim izlaznim slojem. To se odnosi na koeficijente težinskih faktora, posebice za matricu težina izlaznog sloja  $W$ , koja kod mreže sa 4 izlaza sadrži 4 retka čime je složeniji postupak njena izračuna.

## 7. ZAKLJUČAK

Na temelju rezultata testiranja svih triju mreža vidljivo je da na potreban broj koraka učenja može utjecati više faktora. Jedan je izbor aktivacijske funkcije. Funkcija kojoj je bilo potrebno najviše koraka učenja za sve primjere je unipolarna sigmoidalna funkcija. Kao najbržom funkcijom u radu moglo bi se reći da je sinusna funkcija. Također, pokazano je da na korake učenja, a i na konačan rezultat veliki utjecaj ima željena vrijednost očekivanog NRMS parametra točnosti učenja. Što je očekivani ili zadani nrms manji, to je veći broj koraka učenja za sve četiri aktivacijske funkcije. Međutim, većim brojem koraka učenja raste točnost odziva mreže, tj. mreža je korištenjem NRMS-a vrijednosti 0.01 puno bliže točnom rezultatu nego kad koristi NRMS 0.03, iako u konačnici ponekad i izbaci dobar rezultat s obzirom da je u njoj definirano da zaokruži konačni rezultat.

U radu je prikazan i utjecaj brzine učenja  $\eta$  na proces učenja. Brzina učenja također utječe na korake učenja mreže. Ukoliko uzmemo malu vrijednost brzine učenja, broj koraka učenja će se povećati. U slučaju odabira veće vrijednosti brzine učenja broj koraka će se smanjiti, a samim time odstupanje od konačnog rezultata će se povećati.

Također, prikazan je i utjecaj momentuma prvog i drugog reda na proces učenja mreže. Vidljivo je da momentum ne mora nužno ubrzati učenje i smanjiti korake učenja, već da za slučaj sinusne i Gaussianove funkcije može dodatno usporiti proces učenja.

Prema rezultatima testiranja vidljivo je da mreža s dva ulaza i jednim izlazom može davati točne rezultate ukoliko iz nje izbacimo čak 9 kombinacija brojeva. To se odnosi na brojeve ispred kojih i iza kojih se nalazi broj koji je mreža naučila.

Mreža s dva ulaza i tri izlaza pokazala je da za veći iznos NRMS-a teže dolazi do točnog rezultata. Isto vrijedi i za mrežu s dva ulaza i četiri izlaza. Kod ovih dviju mreža potrebno je koristiti što manji iznos NRMS-a kako bi dobili točni rezultat. Pravilnim odabirom NRMS-a mreža je uspjela izračunati čak 14 nepoznatih kombinacija.

Na korake učenja utječe i postavljanje težina skrivenog sloja  $\mathbf{W}$  i  $\mathbf{V}$ . Povećanjem izlaznih slojeva mijenjaju se vrijednosti i dimenzije matrice  $\mathbf{W}$ . Vidljivo je iz rezultata da povećanjem matrice  $\mathbf{W}$  i izlaza iz mreže raste i broj koraka učenja. Dakle, ukoliko uzmemo jednake parametre, dobit ćemo za svaku mrežu drugačiji broj koraka učenja. Povećanjem izlaza mreže, tj. povećanjem matrice  $\mathbf{W}$  rast će i koraci učenja, a time i vrijeme učenja mreže.

Ukoliko se koristi mreža sa dva sakrivena sloja, rezultati učenja biti će poprilično slični rezultatima koje daje jednoslojna neuronska mreža za sve četiri aktivacijske funkcije. Razlika će biti u broju koraka učenja i vremenu izračuna koji često puta znaju biti znatno veći nego kod jednoslojne mreže.

Na kraju mogu zaključiti da je za prikazani problem najefikasnija jednoslojna mreža s jednim izlazom. Jednoslojna mreža uspijeva izračunati najveći broj nepoznatih umnožaka brojeva bez

smanjivanja NRMS-a. Također, do konačnih rezultata dolazi uz najmanji broj koraka i najmanjeg vremena izračuna, te je sam proračun procesa učenja znatno jednostavniji.



## 8. LITERATURA

- [1] Branko Novaković, Dubravko Majetić, Mladen Široki, *Umjetne neuronske mreže*, FSB, Zagreb, 2011
- [2] W. McCulloch, and W. Pitts, *A logic calculus of the ideas immanent in nervous activity*, Bulletin of Mathematical Biophysics 5, pp. 115-113, 1943.
- [3] B. Novaković, *Stanje i trend umjetnih neuronskih mreža i robotike*, Vjesnik HAZU, vol. 1-3, str. 125-147, Zagreb, 1993.
- [4] Dubravko Majetić, *Identifikacija nelinearnih sustava dinamičkom neuronskom mrežom*, Doktorska disertacija, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, Zagreb, 1996.
- [5] M. Široki, *Neiterativni postupak učenja parametara neuronskih mreža*, Doktorska disertacija, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, Zagreb, 1996.
- [6] T. Poggio and F. Girosi, *Extensions of a theory of networks for approximation and learning: dimensionality reduction and clustering*, A. I. Memo 1167, C.B.I.P. Paper no. 44, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1990.
- [7] B. Novaković, *Discrete time neural network synthesis using input and output activation functions*, IEEE Trans. on Systems, Man, and Cybernetics -Part B: Cybernetics, vol 26, no. 4, pp. 533-541, August 1996.
- [8] P.D. Wasserman, *Neural computing: Theory and practice*, Van Nostrand Reinhold, New York, 1989.
- [9] M. Minsky, and S. Papert, *Perceptrons; An introduction to computational geometry*, The MIT Press, Cambridge, 1969.
- [10] P.D. Wasserman, *Neural computing: Theory and practice*, Van Nostrand Reinhold, New York, 1989.
- [11] J.M. Zurada, *Artificial Neural Systems*, W.P Company, USA, 1992.
- [12] M. Smith, *Neural Networks for Statistical Modeling*, Van Nostrand Reinhold, New York, 1993.
- [13] D.E. Rumelhart, G.E. Hinton, R. J. Williams, *Learning Internal Representations by Error Propagation*, poglavlje 8. iz knjige Parallel and Distributed Processing, MIT, Press, Vol. 1, 1986.
- [14] B. Widrow, M.A. Lehr, 30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation, Proceedings of the IEEE, Vol. 78 pp, 1415-1442, 1990.
- [15] V. Kecman, B. M. Pfeifer, *Exploiting The Structural Equivalence of Learning Fuzzy Systems and Radial Basis Function Neural Networks*, Proceeding of Sec. Eur. Congress on Intell. Techniq, and Soft Comp., EUFIT 94, Aachen, Vol. 1., pp. 58-66, 1994.
- [16] M. Essert, T. Ćilić, Matlab- Matrični laboratorij, Zagreb, 2004.

## 9. PRILOG

Tablica 9.1 Pregled grafičkog sučelja

### Jednoslojna neuronska mreža s dva ulaza i jednim izlazom

**Početak rada neuronske mreže**

broj ulaznih neurona - 2  
broj izlaznih neurona - 1

U nastavku koristi:

Opis programa:

**Odaberite brojeve koje želite množiti**

Odabir prvog broja:   
 Odabir drugog broja:

	množenik	mnoštak
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3

**Definiranje parametara učenja**

Broj ulaznih neurona:   
 Broj izlaznih neurona:

Odabir funkcije:   
 Odabir težine:   
 Odabir momentuma:

Upišite brzinu učenja:   
 Upišite željeni NRMS:   
 Upišite broj koraka učenja:

Centar funkcije:  Širina funkcije:   
 Broj neurona skrivenog sloja:   
 Iznos momentuma:

Učenje mreže:

**TEST MREŽE**

Odabir kombinacija brojeva za testiranje:

Ručni unos brojeva za testiranje:  
 Prvi broj:   
 Drugi broj:

Odziv mreže:   
 Stvarna vrijednost:   
 Točna vrijednost:

Podsjetak na odabrane kombinacije:  

	množenik	mnoštak
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3

**Početak rada neuronske mreže**

broj ulaznih neurona - 2  
broj izlaznih neurona - 3

U nastavku koristi:

Opis programa:

**Odaberite željenu kombinaciju brojeva**

Prvi broj:   
 Drugi broj:

	prvi broj	drugi broj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3

Definiranje parametara učenja

Odabir funkcije:

Odabir težine:

Odabir momentuma:

Upišite brzinu učenja:

Upišite željeni NRMS:

Upišite broj koraka učenja:

Broj ulaznih neurona:

Broj izlaznih neurona:

Centar funkcije:  Širina funkcije:

Broj skrivenih neurona:

Iznos momentuma:

Učenje mreže:

**UČENJE**

**ISPIS**

TEST MREŽE

Podsjetnik na odabrane kombinacije

	prvi broj	drugi broj	
1	1	1	1^
2	1	2	
3	1	3	
4	1	4	
5	1	5	
6	2	1	
7	2	2	2^

Odaberite kombinaciju koju želite testirati

Prvi broj:  Drugi broj:

**TEST**

Ručni unos brojeva za testiranje

Prvi broj:

Drugi broj:

**TEST**

	Kvadrat prvog broja	Kvadrat drugog broja	Umnožak
Odziv mreže:	<input type="text"/>	<input type="text"/>	<input type="text"/>
Stvarna vrijednost:	<input type="text"/>	<input type="text"/>	<input type="text"/>
Točna vrijednost:	<input type="text"/>	<input type="text"/>	<input type="text"/>

U nastavku možete:  
 - ponoviti učenje za istu kombinaciju brojeva  
 - odabrati novu kombinaciju brojeva i učiti  
 - završiti sa radom u programu

**POVRATAK NA UČENJE**

**POVRATAK NA POČETAK**

**KRAJ**

### Jednoslojna neuronska mreža s dva ulaza i četiri izlaza

Početak rada neuronske mreže

broj ulaznih neurona - 2

broj izlaznih neurona - 4

U nastavku koristi:

Nove kombinacije brojeva

Postojeće kombinacije brojeva

Opis programa:

Odaberite kombinacije brojeva koje želite koristiti kod učenja mreže:

Odabir prvog broja:

Odabir drugog broja:

Odabrane kombinacije:

	mnogobroj	mnogobroj	
1	1	1	1^
2	1	2	
3	1	3	
4	1	4	
5	1	5	
6	2	1	
7	2	2	
8	2	3	2^

**NASTAVI**

Definiranje parametara učenja

Odabir funkcije:

Odabir težine:

Odabir momentuma:

Upišite brzinu učenja:

Upišite željeni NRMS:

Upišite broj koraka učenja:

Broj ulaznih neurona:

Broj izlaznih neurona:

Centar funkcije:  Širina funkcije:

Broj skrivenih neurona:

Iznos momentuma:

Učenje mreže:

**UČENJE**

**ISPIS**

### TEST MREŽE

**Podsjetnik na odabrane kombinacije**

	prvi broj	drugi broj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2

Odaberite kombinaciju koju želite testirati

Prvi broj:

1

Drugi broj:

1

**TEST**

Ručni unos brojeva za testiranje

Prvi broj:

Drugi broj:

**TEST**

Odziv mreže:

Stvarna vrijednost:

Točna vrijednost:

Kvadrat prvog broja:

Kvadrat drugog broja:

Suma:

Umnožak:

U nastavku možete:

- ponoviti učenje za istu kombinaciju brojeva
- odabrati novu kombinaciju brojeva i učiti
- završiti sa radom u programu

**POVRATAK NA UČENJE**

**POVRATAK NA POČETAK**

**KRAJ**

## Dvoslojna neuronska mreža s dva ulaza i jednim izlazom

### Početak rada neuronske mreže

broj ulaznih neurona - 2

broj izlaznih neurona - 1

**U nastavku koristi:**

Nove kombinacije brojeva

Postojeće kombinacije brojeva

Opis programa:

Odaberite brojeve koje želite množiti

Odabir prvog broja:

1

Odabir drugog broja:

1

Odabrane kombinacije:

	množenik	množitelj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3

**NASTAVI**

### Definiranje parametara učenja

Odabir funkcije:

Odabir težine:

Odabir momentuma:

Upišite brzinu učenja:

Upišite željeni NRVS:

Upišite broj koraka učenja:

Broj ulaznih neurona:

Broj izlaznih neurona:

Prvi sloj

Centar funkcije:

Širina funkcije:

Broj neurona prvog skrivenog sloja:

Iznos momentuma:

Drugi sloj

Centar funkcije:

Širina funkcije:

Broj neurona drugog skrivenog sloja:

**UČENJE**

**ISPIS**

### TEST MREŽE

Odabir kombinacija brojeva za testiranje

1

1

**TEST**

Podsjetnik na odabrane kombinacije:

	množenik	množitelj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3

Ručni unos brojeva za testiranje

Prvi broj:

Drugi broj:

**TEST**

Učenje mreže

Odziv mreže:

Stvarna vrijednost:

Točna vrijednost:

**NOVI TEST**

**VRATI NA UČENJE**

**VRATI NA POČETAK**

**KRAJ**

## Dvoslojna neuronska mreža s dva ulaza i četiri izlaza

### Početak rada neuronske mreže

broj ulaznih neurona - 2  
broj izlaznih neurona - 4

**U nastavku koristi:**

Opis programa:

Odaberite kombinacije brojeva koje želite koristiti kod učenja mreže:

Odabir prvog broja:

1
2
3
4
5

Odabir drugog broja:

1
2
3
4
5

Odabrane kombinacije:

	množenik	mnoštvo	
1	1	1	^
2	1	2	
3	1	3	
4	1	4	
5	1	5	
6	2	1	
7	2	2	^
8	2	3	
>	>	>	>

### Definiranje parametara učenja

Odabir funkcije:

Odabir težine:

Odabir momentuma:

Upišite brzinu učenja:

Upišite željeni NRMS:

Upišite broj koraka učenja:

Broj ulaznih neurona:

Broj izlaznih neurona:

Prvi sloj

Centar funkcije:

Širina funkcije:

Broj neurona prvog skrivenog sloja:

Iznos momentuma:

Drugi sloj

Centar funkcije:

Širina funkcije:

Broj neurona drugog skrivenog sloja:

Učenje mreže:

### TEST MREŽE

Podsjetnik na odabrane kombinacije

	prvi broj	drugi broj
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2

Odaberite kombinaciju koju želite testirati

Prvi broj:

1
2
3
4
5

Drugi broj:

1
2
3
4
5

Ručni unos brojeva za testiranje

Prvi broj:

Drugi broj:

Odziv mreže:

Stvarna vrijednost:

Točna vrijednost:

Kvadrat prvog broja:

Kvadrat drugog broja:

Suma:

Umnožak:

U nastavku možete:

- ponoviti učenje za istu kombinaciju brojeva
- odabrati nova kombinacija brojeva i učiti
- završiti sa radom u programu