

# UBRZANJE UČENJA VEŠTAČKIH NEURONSKIH MREŽA STATISTIČKIM METODAMA

Zarković, K., Litovski, V. i Stojilković, S.  
Elektronski fakultet, ul. Beogradska 14, 18000 Niš

## I. UVOD

Neuronska mreža kojom je moguće aproksimirati bilo koju nelinearnu funkciju mora da sadrži barem jedan sakriveni sloj neurona koji je postavljen između ulaznog i izlaznog sloja.

Mreža uči da aproksimira funkciju na osnovu primera. Prezentiraju joj se tačne vrednosti ulaza i izlaza. Tom prilikom se izračunava srednje kvadratna greška koja predstavlja razliku između odziva mreže za date ulaze i željenih vrednosti izlaza. Na osnovu doprinosa grešci koriguju se vrednosti težina i pragova metodom gradijenta [1]. Opisani postupak učenja ne postane zadovoljavajuća. Proces učenja se zaustavlja i u slučaju da dalje menjanje vrednosti težina i pragova ne dovodi do smanjenja greške.

## 2. PROBLEM POČETNOG REŠENJA

Na kvalitet aproksimacije ostavarene mrežom utiče više faktora [2]. Broj neurona u sakrivenom sloju i broj podataka na kojima se obučava mreža su značajni sa stanovišta brzine učenja ali od njih zavisi i sposobnost mreže da aproksimira funkciju. Na brzinu učenja i krajnju vrednost greške takođe utiče i koeficijent  $\alpha$  [3] kojim se preslikava vrednost gradijenta funkcije greške na promenu vrednosti težina odnosno pragova.

Poseban problem predstavlja određivanje početnog rešenja za vrednosti težina i pragova. Učenje se može posmatrati kao pomeranje vektora greške po površini određenoj funkcijom greške u prostoru težina i pragova. Vektor greške je različit za svaku iteraciju (u upotrebi i ciklus, epoha) u procesu učenja i određen je aktuelnim vrednostima težina i pragova. Kod učenja ovaj vektor klizi ka dnu površi odnosno ka apsolutnom minimumu funkcije greške. U tom kontekstu se generisanje početnog rešenja može posmatrati kao postavljanje vektora greške na neku od tačaka na površi funkcije greške. Pretpostavka je da će vektor greške biti postavljen bliže dnu površi ukoliko je njegova vrednost manja i da će zato biti potreban manji broj ciklusa učenja kao i manja verovatnoća da se proces učenja zaustavi u nekom od lokalnih minimuma funkcije greške. Problem lokalnog minimuma nije jako izražen jer usled postojanja mnogo dimenzija prostora težina i pragova postoji mnogo stepena slobode.

Klasičan pristup procesu učenja podrazumeva generisanje početnih vrednosti za težine i pragove po slučajnoj raspodeli u intervalu malih brojeva (npr.  $\pm 0.5$ ). Ovde su prikazane unapredne varijante generisanja početnog rešenja: Monte Karlo (MK) metod i metod gravitacionih centara (center of gravity - COG).

## 3. MONTE KARLO METOD

Za razliku od uobičajenog generisanja početnog rešenja za vrednosti težina i pragova Monte Karlo metodom se vrši višestruko generisanje. Sva generisanja (u svakom od ciklusa i za svaki od parametara) imaju jedini tip i parametre raspodele. Za svako uzorkovanje izračunava se greška na izlazu iz mreže. Kao optimalan se bira onaj vektor čija je greška minimalna.

Monte Karlo pristup pogodan je zbog toga što povećanje broja dimenzija (u ovom slučaju broja težina i pragova) ne zahteva eksponencijalno povećanje broja uzorkovanja radi očuvanja dobitka na početnom rešenju. Drugim rečima, uvođenje nove dimenzije ne zahteva eksponencijalno veći broj uzorkovanja da bi se postigla istovetna pokrivenost površine funkcije greške. Kod primene na mnogo dimenzionalan prostor težina i pragova neuronske mreže ovakva osobina ima izuzetan značaj za stanovišta brzine.

## 4. METOD GRAVITACIONIH CENTARA

Metod gravitacionih centara [4] se takođe oslanja na generisanje slučajnih vrednosti. Međutim, ovaj algoritom se utiče na parametre raspodele.

Za svako uzorkovanje se izračunava greška na izlazu mreže. Ta greška se upoređuje sa unapred postavljenom graničnom vrednošću greške. Na osnovu toga da li je greška generisane mreže manja ili veća od granične greške uzorkovanje se proglašava "uspešnim" (pass) odnosno "neuspešnim" (fail). Na taj način se dobijaju oblasti na površini funkcije greške sa grupama tačaka na koje su upereni vektori "prihvaćenih" odnosno "neprihvaćenih" uzorkovanja. Zatim se izračuna "gravitacioni centar" (GC) za obe ove grupe. GC koji je dobijen prethodnom iteracijom se pomera paralelno potežu od GC neprihvaćenih (GCFail) ka GC prihvaćenih (GCPass) rešenja za veličinu rastojanja između CGP i CGF pomnoženog koeficijentom  $\lambda$ . Niz iteracija kojima se određuju gravitacioni centri naziva se nizom iteracija osvežavanja (fresh).

COG metod podrazumeva više ciklusa. U svakom od ciklusa izvršava se niz iteracija osvežavanja a zatim se na osnovu dobivenih vrednosti postavlja GC za naredni ciklus. Kod poslednjeg niza uzorkovanja ne vrši se izračunavanje gravitacionog centra već se upoređuju vrednosti greške na izlazu mreže kao MK metodom. U primeni na generisanje početnog rešenja za neuronsku mrežu COG metodom se utiče na pomeraj (offset) funkcije raspodele u odnosu na koordinatni početak. Ovaj parametar se menja za svaku težinu odnosno prag zasebno. Težište, odnosno GC, se izračunava kao aritmetička

sredina za svaku od koordinata posebno (tj. za svaku od težina i pragova). Početni pomeraj za svaku od raspodela težina i pragova je nula.

Pošto greška ima različite vrednosti za različite funkcije, graničnu vrednost greške je neophodno odrediti za svaku od aproksimiranih funkcija posebno. Za to je formiran niz uzorkovanja koji se izvršava pre no što se započne COG metod. Ovim uzorkovanjem se određuju minimalne i maksimalne vrednosti greške. Realno to nisu apsolutne granice greške ali sa dovoljnim brojem iteracija ovako dobivene vrednosti omogućuju postavljanje granične vrednosti greške.

## 5. REZULTATI PRIMENE

Poređeni su klasični, MK i COG pristup generisanju početnog rešenja. Neuronska mreža je imala 2 neurona na ulazu, 10 u sakrivenom sloju i 1 na izlazu. Aproksimirana je karakteristika MOS tranzistora (predstavljena uz pomoć 504 uzorka). U MK je korišćeno 100 ciklusa generisanja. Dakle, ovim metodom je stotinu puta izvršeno slučajno generisanje vrednosti težina i pragova. Od ovih stotinu generisanih početnih rešenja kao optimalno je birano ono sa najmanjom srednje kvadratnom greškom. COG metod je imao 4 ciklusa generisanja. Prva 3 ciklusa sadržala su po 20 iteracija osvežavanja. Njima se uticalo na postavljanje GC. Name, u svakom od ciklusa vrednosti težina i pragova su postavljane slučajno. Postupkom opisanim u odeljku 4 određivan je položaj GC. Na osnovu novoizračunatih GC u narednim ciklusima su određivani novi parametri raspodele kojom su generisane vrednosti težina i pragova. Nakon ova 3 ciklusa pristupalo se generisanju mreže sa minimalnom greškom koristeći predhodno dobivene vrednosti gravitacionih centara. Ovaj ciklus generisanja je izvršavan u 40 iteracija na isti način na koji je izvršavano stotinu iteracija MK metoda. U sva 4 ciklusa zajedno COG metod je imao 100 iteracija. Vremena potrebna da se izvrši jedna iteracija uzorkovanja za MK i COG metode su približno ista. Jedan ciklus učenja mreže metodom gradijenta traje približno 15 puta duže tako da je trajanje 100 iteracija uzorkovanja ekvivalentno trajanju 6 ciklusa učenja.

Nakon uzorkovanja, mreže dobivene na sva tri načina su učene koristeći metodu gradijenta u 10 ciklusa. Prikazane su vrednosti greške na početku učenja i nakon 10 ciklusa (tab. 1). Početna greška je greška karakteristična za mrežu čije su vrednosti težina i pragova dobivene nekim od metoda generisanja. Iz tabele 1 se može uočiti da je ova greška značajno manja kada su korišćeni MK ili COG metodi nego u slučaju kada je vršeno generisanje klasičnom metodom. Krajnja greška predstavlja vrednost greške nakon 10 ciklusa učenja metodom gradijenata. Zanim je ponovo pokretano učenje mreže generisanih klasičnim metodom. Beleženi su brojevi dodatnih ciklusa učenja nakon kojih je dosuzana greška približna grešci dostignutoj sa 10 ciklusa učenja mrežama generisanim MK i COG metodama (tab. 2). Na

taj način je predstavljena ušteda u broju ciklusa učenja metodom gradijenata koja je postignuta unaprednim metodama generisanja početnog rešenja. Treba imati u vidu da je trajanje jednog ciklusa učenja ekvivalentno trajanju 15 iteracija generisanja, bilo MK bilo COG metodom.

Tabela 1: Greška na početku učenja i nakon 10 ciklusa učenja

	Klasični metod		MK metod		COG metod	
	početna	krajnja	početna	krajnja	početna	krajnja
1.	650.459	2.90309	26.8574	2.17494	24.7183	1.59642
2.	50.0555	4.27794	20.9973	3.25125	30.1571	2.14551
3.	54.5071	4.08156	7.83483	3.63278	9.04679	1.26527
4.	14.3293	3.40330	8.84967	2.36373	19.1276	0.77575
5.	224.695	1.77580	10.7247	2.48246	18.9350	1.36230
6.	694.417	2.52369	21.0788	2.08929	23.6969	2.38817
7.	505.450	3.74228	11.6943	2.49842	25.2763	2.89461
8.	59.4308	1.68873	17.5145	3.16742	16.4511	1.21147
9.	188.556	2.17866	21.7445	2.70427	27.8268	1.42005
10.	270.596	3.18963	8.27871	1.95234	55.2761	1.75551
11.	242.425	3.56312	13.8702	1.49020	46.1953	1.98154

Tabela 2: Broj dodatnih ciklusa učenja za klasični metod

MK	5	4	5	4	-2	4	5	-4	-2	6	7
COG	6	7	15	15	4	2	5	-2	4	6	6

## 6. ZAKLJUČAK

Proces učenja neuronske mreže započinje se sa slučajno odabranim vrednostima težina i pragova. Ovako oformljena mreža aproksimira željenu funkciju sa određenom greškom. Što je ta greška manja, veća je verovatnoća da je početni položaj vektora greške postavljen bliže apsolutnom minimumu funkcije greške. Time se može smanjiti broj ciklusa učenja kojim se dostiže zahtevani kvalitet aproksimacije. MK i COG metodi omogućuju da se vrednost početnog vektora greške smanji i da se na taj način skratí put ovog vektora ka apsolutnom minimumu funkcije greške. Zbog kraćeg trajanja ciklusa (u odnosu na cikluse učenja), MK i COG metodi se mogu primeniti za postizanje smanjenja broja ciklusa učenja a samim tim i na smanjenje ukupnog vremena potrebnog za ostvarivanje aproksimacije.

## LITERATURA

- [1] Zografski Z., *A Novel Machine Learning Algorithm and its Use in Modelling and Simulation of Dynamical Systems*, 1991.
- [2] K. Knight, *Connectionist Ideas and Algorithms*, Communications of the ACM, November 1990.
- [3] J. Anderson and E. Rosenfeld, eds., *Neurocomputing: Foundations of Research*, Boston MIT Press, 1988.
- [4] R. Spence, R. S. Soin, *Tolerance of electric circuits*, Addison-Wesley Publishers, 1988.

Abstract: Good initial starting values for neural network's weights and thresholds decrease number of learning cycles which produce certain error. Monte Carlo and center of gravity methods are able to improve those starting values. IMPROVEMENTS OF THE ANN'S LEARNING BY STATISTICAL METHODS, Kristijan Zarković, Vančo Litovski, Stojan Stojilković