

Umjetna inteligencija – Laboratorijska vježba 4

UNIZG FER, ak. god. 2011/12.

Zadano: 25.5.2012. Rok predaje: 3.6.2012. do 23.59 sati.

Tema ove laboratorijske vježbe jesu umjetne neuronske mreže i genetski algoritam. Zadatak vježbe jest napisati program za učenje (treniranje) umjetne neuronske mreže (tj. određivanje njezinih težina) pomoću genetskog algoritma. Neuronska mreža kakva se koristi u ovom zadatku tipično se uči (trenira) algoritmom propagacije pogreške unazad (engl. *backpropagation algorithm*). Međutim, učenje mreže pomoću genetskog algoritma šire je primjenjivo, a ima i neke druge prednosti. Osnovna ideja jest oblikovati populaciju neuronskih mreža (svaka jedinka odgovara jednoj inačici neuronske mreže i na sažet način kodira sve njezine težine), a zatim genetskim algoritmom optimirati iznose težina s obzirom na pogrešku mreže na skupu za učenje. U ovom zadatku arhitektura mreže je fiksna; u općenitom slučaju postupak bi se mogao ostvariti tako da se, pored težina, optimizira i arhitektura mreže. Nakon što je mreža naučena, njezinu je učinkovitost potrebno provjeriti na skupu za ispitivanje, koji je različit od skupa za učenje. Zadatak možete implementirati u proizvoljnom programskom jeziku.

Neuronska mreža

Koristite potpuno povezanu neuronsku mrežu arhitekture $1 \times 4 \times 1$ (jedan ulazni neuron, četiri neurona u skrivenom sloju i jedan neuron u izlaznom sloju). Neuroni u skrivenom sloju koriste sigmoidalnu prijenosnu funkciju. Neuron u izlaznom sloju kao prijenosnu funkciju koristi funkciju identiteta ($f(x) = x$). Tijekom postupka učenja program treba ispisivati broj generacije te iznos ukupne pogreške. Broj kromosoma te uvjete zaustavljanja postupka učenja (dopušten iznos ukupne pogreške i broj iteracija) neka zadaje korisnik kao parametre programa. Nakon što je postupak učenja završen, program treba omogućiti korisniku da unosi nove uzorke te za svaki uzorak treba ispisati izlaz prethodno naučene mreže. Također, program treba omogućiti korisniku da učitava datoteku sa ispitnim primjerima (skup za ispitivanje) te zatim ispitati ukupnu pogrešku neuronske mreže na tim primjerima (srednje kvadratno odstupanje).

Genetski algoritam

Učenje neuronske mreže ostvarite pomoću generacijskog genetskog algoritma s populacijom od 50 jedinki i ugrađenim elitizmom, koji prenosi najbolju jedinku u sljedeću generaciju. Kao prikaz rješenja koristite polje decimalnih brojeva – svaka jedinka treba biti predstavljena poljem koje sadrži onoliko elemenata koliko neuronska mreža ima težinskih faktora. Operator križanja izvedite kao izračun aritmetičke sredine između dviju jedinki. Operator mutacije izvedite dodavanjem broja nasumično odabranog iz normalne distribucije $N(0, K)$, gdje je K neka odabrana vrijednost. Pri tome mutirajte samo p posto težina odjednom (neka je $p = 4/N$, gdje je N broj težina). Isprobajte ponašanje algoritma za različite vrijednosti konstante K (npr. $K = 0.1$, $K = 1$ i $K = 10$). Omogućite da se

u programu p može jednostavno mijenjati. Kao mjeru pogreške svake neuronske mreže koristite srednje kvadratno odstupanje. Kao operator selekcije koristite proporcionalnu selekciju.

Skup za učenje

Mrežu treba naučiti preslikavanje $y = f(x)$, definirano sljedećom tablicom (skup je dostupan na ovoj [poveznici](#)):

x	$f(x)$
0.000	2.000
0.372	2.111
0.743	2.221
1.115	2.329
1.486	2.434
1.858	2.534
2.230	2.628
2.601	2.717
2.973	2.797
3.344	2.870
3.716	2.934
4.087	2.988
4.459	3.032
4.831	3.065
5.202	3.087
5.574	3.099
5.945	3.099
6.317	3.087
6.689	3.065
7.060	3.032
7.432	2.988
7.803	2.934
8.175	2.870
8.546	2.797
8.918	2.717
9.290	2.628
9.661	2.534
10.033	2.434
10.404	2.329
10.776	2.221
11.148	2.111
11.519	2.000

Skup za ispitivanje

Modeli strojnog učenja tipično se uče na jednom skupu (skupu za učenje), a ispituju na drugom skupu (skupu za ispitivanje). Na taj način može se poprilično pouzdano utvrditi koliko je dobro model sposoban generalizirati. Ispitajte rad neuronske mreže naučene

genetskim algoritmom na sljedećim primjerom (datoteka je dostupna na ovoj [poveznici](#))
te izračunajte pogrešku mreže (srednje kvadratno odstupanje):

x	$f(x)$
0.186	2.056
0.557	2.167
0.929	2.276
1.301	2.382
1.672	2.484
2.044	2.582
2.415	2.673
2.787	2.758
3.158	2.835
3.530	2.903
3.902	2.962
4.273	3.011
4.645	3.050
5.016	3.077
5.388	3.094
5.760	3.100
6.131	3.094
6.503	3.077
6.874	3.050
7.246	3.011
7.618	2.962
7.989	2.903
8.361	2.835
8.732	2.758
9.104	2.673
9.475	2.582
9.847	2.484
10.219	2.382
10.590	2.276
10.962	2.167
11.333	2.056
11.705	1.944