

# Klasifikacija slika pomoću neuronskih mreža

22. decembar 2015

U ovoj vježbi ćemo iskoristiti troslojnu neuronsku mrežu za prepoznavanje rukom pisanih cifara. Ulaz u neuronsku mrežu su slike cifara dimenzija  $20 \times 20$  piksela. Izlaz treba da budu oznake cifara koje se nalaze na slikama. Podaci za ovu vježbu se nalaze u fajlu `cifre.mat` čiji se sadržaj može učitati u radni prostor MATLAB-a pomoću naredbe `load`. Radi se o binarnom fajlu u kojem su sačuvane matrica  $X$  i vektor  $y$ . U redovima matrice  $X$  se nalazi 5000 primjera cifara koje su vektorizovane u 400-dimenzionalne vektore. Dakle, matrica  $X$  je dimenzija  $5000 \times 400$  i svaki njen red je jedan primjer slike rukom pisane cifre.

U fajlu `cifre.mat` se nalazi i 5000-dimenzionalni vektor  $y$  čiji su elementi oznake slika iz matrice  $X$ . Radi kompatibilnosti sa MATLAB-ovom konvencijom indeksiranja nizova, cifri "0" je dodijeljena oznaka "10", a cifre od "1" do "9" su označene oznakama od "1" do "9", respektivno.

Za klasifikaciju ćemo koristiti troslojnu neuronsku mrežu, prikazanu na Slici 1. Ulazni sloj mreže sadrži 400 neurona na čije ulaze se dovode vrijednosti piksela slike. Izlazni sloj sadrži 10 neurona koji odgovaraju klasama – ciframa. U fajlu `tezine.mat` nalaze se težine troslojne neuronske mreže sa 25 neurona u skrivenom sloju koja je već obučena za prepoznavanje rukom pisanih cifara. Težine veza između neurona ulaznog i skrivenog sloja se nalaze u matrici  $W1$  čije su dimenzije  $25 \times 401$ . Dodatna kolona odgovara težinama dodatnog neurona u ulaznom sloju čiji je izlaz fiksiran na vrijednost  $+1$ . Težine veza između neurona skrivenog i izlaznog sloja nalaze se u matrici  $W2$  čije su dimenzije  $10 \times 26$ . Dodatna kolona i u ovom slučaju potiče od dodatnog neurona u skrivenom sloju čiji je izlaz fiksiran na vrijednost  $+1$ .

U matricama  $W1$  i  $W2$  težine su organizovane tako da se u svakom redu nalaze težine veza do jednog neurona. Na primjer, u prvom redu matrice  $W1$  nalaze se težine veza od neurona ulaznog sloja do prvog neurona u skrivenom sloju.

U matrici  $X$  svaki ulazni primjer  $[x_1, x_2, \dots, x_n]^T$  se nalazi u jednom redu.



Konačno, za neurone izlaznog sloja imamo

$$\mathbf{z}^{(3)} = \mathbf{W}^{(2)}\mathbf{a}^{(2)} \quad (5)$$

i

$$\mathbf{a}^{(3)} = g(\mathbf{z}^{(3)}) . \quad (6)$$

U izlaznom sloju ne dodajemo neuron sa fiksnim izlazom zato što je to poslednji sloj neurona i nemamo više potrebu da modeliramo ofsete.

## 1 Uputstva za praktičan rad

Početni MATLAB kod i podaci su na adresi <http://dsp.etfbl.net/multimedia/media/nn.zip>. U datim m-fajlovima je implementirano učitavanje podataka, prikazivanje slika i obučavanje mreže korištenjem backpropagation algoritma.

Sve funkcije pozivaju se iz dva glavna programa:

- `nn_main.m`  
Glavni program u kojem se učitavaju podaci, prikazuje njihov podskup i poziva funkcija za klasifikaciju.
- `nn_train_main.m`  
Glavni program u kojem se obučava neuronska mreža za prepoznavanje rukom pisanih cifara.

## 2 Zadaci

1. Implementirati funkciju `predict` u fajlu `predict.m` kojom se realizuje klasifikacija primjera. Potrebno je izračunati odziv mreže, odnosno odzive neurona izlaznog sloja, za ulazni primjer i klasifikovati ulazni primjer u klasu određenu izlaznim neuronom sa najvećim odzivom. Izlazni vektor treba da sadrži oznake koje je mreža dodijelila svakom ulaznom uzorku.
2. Testirati funkciju na datim podacima. Za date težine mreže tačnost klasifikacije bi trebalo da bude oko 97.5%. Nakon toga će interaktivno biti prikazane slike cifara i oznake klasa u koje ih mreža klasifikuje.
3. U programu `nn_train_main.m` je kod kojim se inicijalizuju težine neuronske mreže i poziva funkcija za minimizaciju funkcije cilja `fmincg`

koja koristi gradijent funkcije cilja izračunat backpropagation algoritmom. Nakon učitavanja podataka iz fajla `cifre` formirajte trening skup kao matricu `Xtrain` koja sadrži 4000 slučajno izabranih uzoraka i vektor oznaka `ytrain` koji sadrži oznake za uzorke iz trening skupa. Takođe, formirajte i testni skup kao matricu `Xtest` koja sadrži preostale uzorke, te vektor oznaka testnih uzoraka `ytest`. U ovoj tački korisna je funkcija `randperm(n)` koja vraća slučajno permutovane cijele brojeve od 1 do `n`.

4. Obučite troslojnu neuronsku mrežu, sa 25 neurona u skrivenom sloju, korištenjem trening skupa iz prethodne tačke. Korištenjem obučene neuronske mreže klasifikovati primjere iz trening i test skupa i odrediti tačnosti klasifikacije u oba slučaja. Komentarisati rezultate sa stanovišta generalizacije.
5. Reprezentacioni kapacitet neuronske mreže je veliki i lako je moguće da neuronska mreža overfituje trening podatke i postigne tačnost od približno 100% na trening skupu. Međutim, u ovom slučaju generalizacija na novim podacima bi bila loša. Da biste ovo ispitali postavite regularizacioni parametar `lambda` na vrijednost 0, a broj iteracija `MaxIter` na 300. Obučite neuronsku mrežu, a zatim klasifikujte primjere iz trening i test skupa i odredite tačnost klasifikacije u oba slučaja.
6. Da biste ispitali uticaj broja neurona u skrivenom sloju na tačnost klasifikacije i sposobnost generalizacije obučite neuronske mreže sa 10 i 50 neurona u skrivenom sloju. Ne zaboravite da postavite `lambda = 1` i `MaxIter = 100`. Korištenjem obučene neuronske mreže klasifikovati primjere iz trening i test skupa i odrediti tačnosti klasifikacije u oba slučaja. Komentarisati rezultate sa stanovišta generalizacije.
7. Eksperimentišite sa brojem neurona u skrivenom sloju, kao i vrijednostima regularizacionog parametra i maksimalnog broja iteracija kako biste postigli najbolju tačnost klasifikacije primjera iz testnog skupa.