

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR

**Prepoznavanje znamenki korištenjem neuronskih
mreža**

Student: Marin Ovčariček

Voditelj: Marko Đurasević

Zagreb, svibanj, 2019.

Sadržaj

1. Općenito o neuronskim mrežama	Error! Bookmark not defined.
2. Karakteristike neuronske mreže	Error! Bookmark not defined.
3. Prepoznavanje znamenki korištenjem neuronskih mreža	11
4. Rezultati i Zaključak	12
5. Sažetak	12

1. Općenito o neuronskim mrežama

Neuronska mreža je računalni model temeljen na strukturi i funkciji bioloških neuronskih mreža. Sastoji se od skupa povezanih umjetnih neurona. Kao i u mozgu neuroni su povezani *sinapsama*, primaju signale, obrađuju ih te proslijeđuju krajnji signal s ostalim povezanim neuronima.

Sama neuronska mreža nije algoritam nego temeljna struktura za mnogo različitih algoritama strojnog učenja koji zajedno služe za procesuiranje kompleksnih ulaznih podataka. Ovakvi sustavi uče kroz velik broj ulaznih primjera i obično nisu programirani za specifične zadatke. Umjesto toga neuronske mreže za svaki primjer daju svoj rezultat, ako on nije zadovoljavajuć, same veze između neurona se podešavaju kako bi se kroz velik broj primjera smanjila greška rezultata neuronske mreže.

U drugom poglavlju su opisane karakteristike neuronske mreže, arhitektura, aktivacijska funkcija, backpropagation, i gradijentni silazak, te njihova primjena i vrste neuronskih mreža. U trećem poglavlju je opisan problem koji se želi riješiti te način rješavanja uz pomoć neuronske mreže. Četvrto poglavlje prezentira i interpretira rezultate dobivene iz napravljene neuronske mreže. Peto poglavlje je sažetak sadržaja seminara, ukratko govori o tome što su neuronske mreže, koja je njihova uloga, te što se postiglo napravljenom neuronskom mrežom.



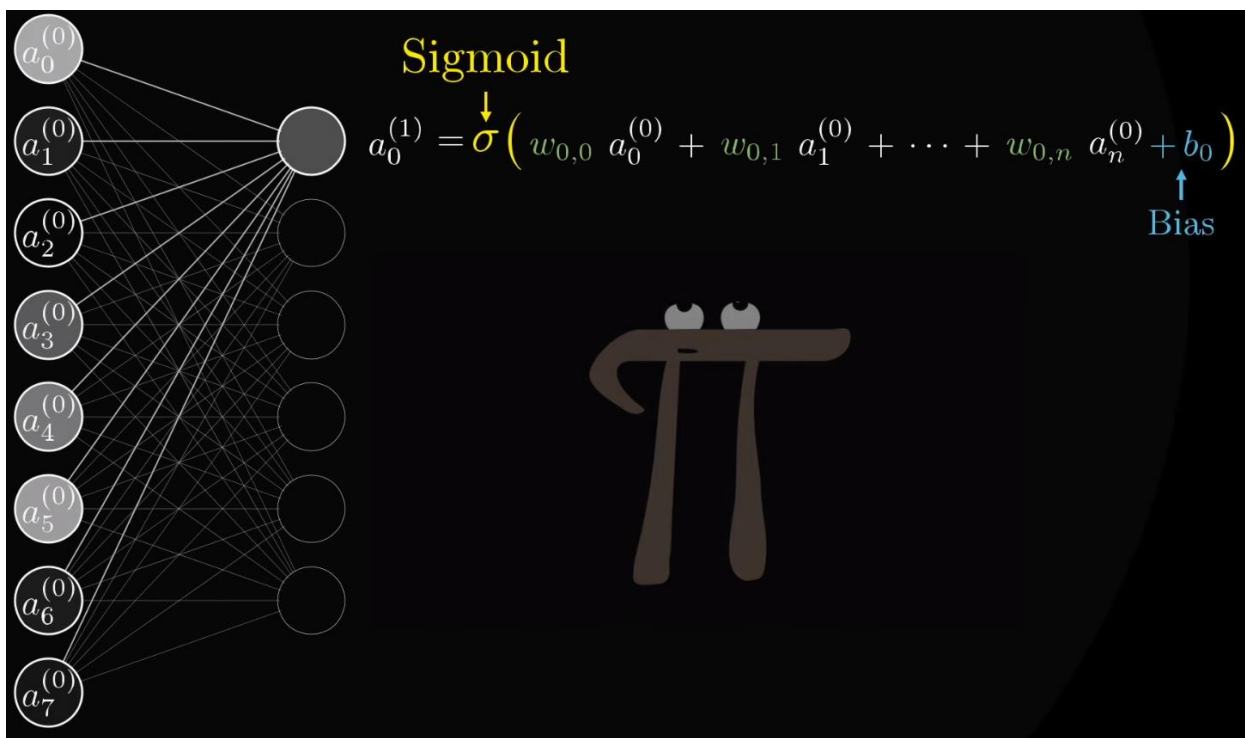
Slika 1: Umjetnički prikaz apstraktne neuronske mreže

2. Karakteristike neuronske mreže

U ovome poglavlju će biti opisani dijelovi i karakteristike općenite neuronske mreže, njihovi nazivi, te njihova uloga u funkcionalnosti mreže.

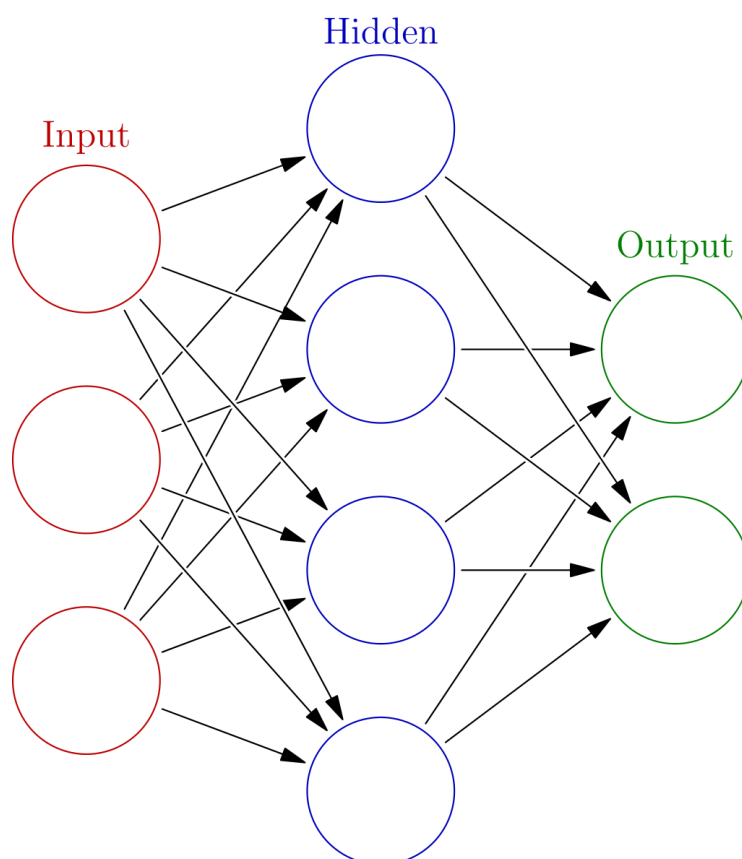
2.1 Arhitektura neuronske mreže

Osnova arhitekture neuronske mreže je neuron. Neuron je apstraktna struktura koja može primiti, obraditi i proslijediti signal drugim neuronima. Neuroni su međusobno povezani vezama (engl. *connections*) koje mogu biti različitih jačina. U većini implementacija neuronskih mreža, signal na vezi dva neurona je realan broj, a izlazni signal neurona se računa nekom ne-linearnom funkcijom zbroja ulaznih signala. Jačine veza između neurona ovise o težinama (engl. *weights*) koje su dodijeljene svakoj vezi i biasu. Težina veze je realan broj kojim se množi signal izlaznog neurona i on određuje koliko je veza između ta dva neurona utjecajna u samoj mreži. Bias (pristranost) je obično realna konstanta koja se zbraja umnošku težina i ulaznog signala, te je jedan od alata za bolje učenje neuronskih mreža.



Slika 2: Primjer neuronske mreže

Neuroni se obično grupiraju u 3 sloja: ulazni (engl. *input*), skriveni (eng. *hidden*), izlazni (engl. *output*) sloj. U neuronskoj mreži su jedan ulazni i izlazni sloj, te proizvoljan broj skrivenih slojeva. Broj skrivenih slojeva te broj neurona u svakom sloju ovise o željenom načinu rješavanja nekog problema. Broj neurona u ulaznom sloju ovisi o broju ulaznih vrijednosti, broj izlaznih o broju traženih izlaza, dok broj skrivenih slojeva ovisi o konkretnom problemu i povećanjem broja skrivenih slojeva i broja neurona u njima se povećava složenost ali i ekspresivnost neuronske mreže. Više neurona u skrivenom sloju i više skrivenih slojeva ne povećavaju nužno točnost neuronske mreže, ali usporavaju njen rad. Njihov broj koji bi davao zadovoljavajući rezultat ovisi o problemu kojeg želimo riješiti.



Slika 3 :Općenita arhitektura neuronske mreže

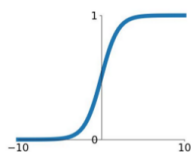
2.1 Aktivacijska funkcija

Aktivacijske funkcije su važne za neuronske mreže jer pomažu mreži da nauči komplicirane i ne-linearne korelacije između ulaznih podataka i izlazne vrijednosti. Kako bi to postigli, biraju se ne-linearne aktivacijske funkcije (prikazane na slici 4).

Activation Functions

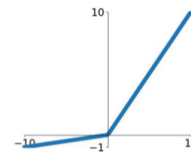
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



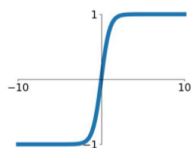
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

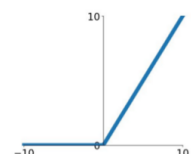


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

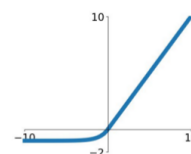
ReLU

$$\max(0, x)$$



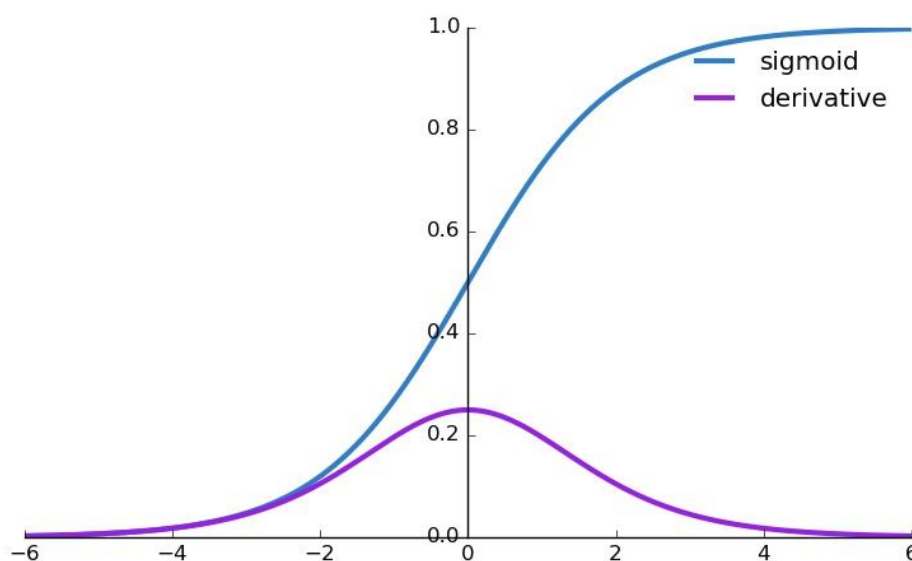
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Slika 4: Primjeri aktivacijskih funkcija

Na slici 5 su prikazane izabrana aktivacijska funkcija i njena derivacija.



Slika 5: Sigmoid funkcija i njena derivacija

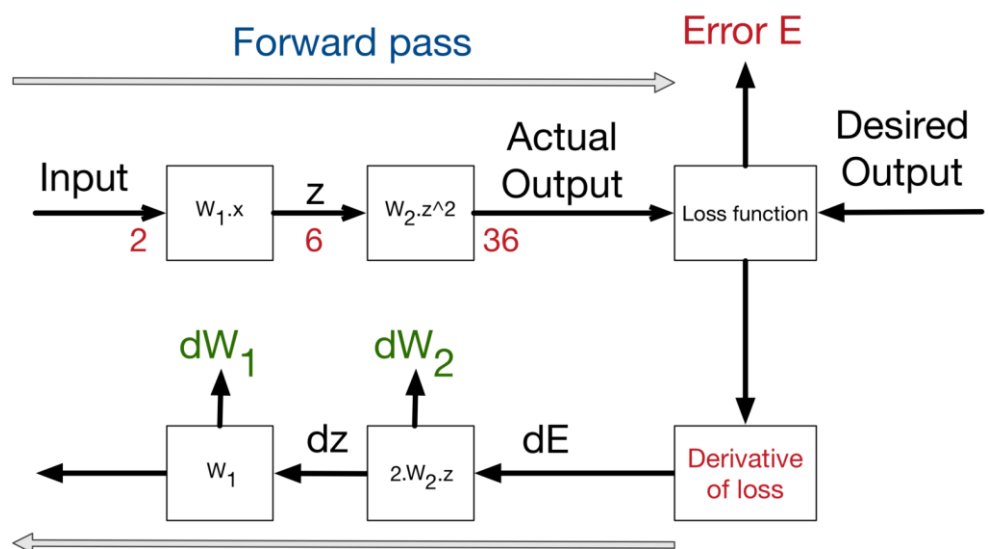
2.2 Cost function, backpropagation i gradijentni silazak

Cilj učenja neuronske mreže je da sa što manjom greškom rješava neki problem. Mjera koliko dobro neuronska mreža radi se zove funkcija greške (engl. *cost function*). Funkcija greške za svaki izlazni neuron se jednostavno računa prema formuli srednje kvadratne greške kako je prikazano na slici.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$

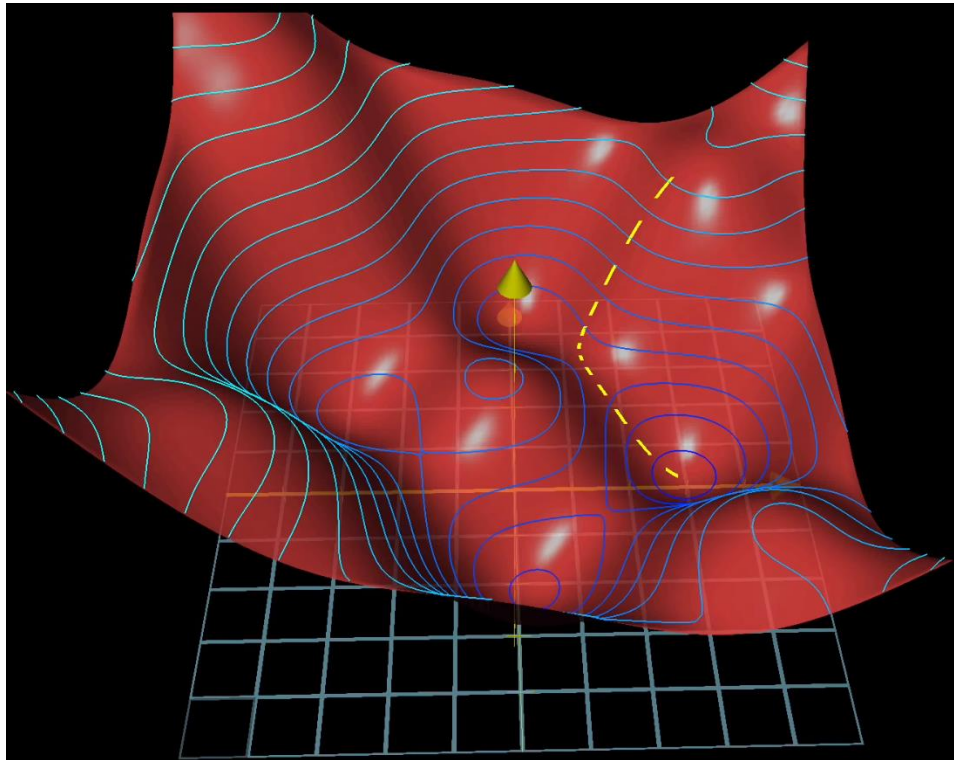
Slika 6: Formula srednje kvadratne greške

Kvadratna greška je jednaka sumi kvadrata razlike svakog mjerenja (Y_i) i srednje vrijednosti (\bar{Y}) podijeljenom s brojem mjerenja (n). Kako bi neuronska mreža bila što točnija cilj je smanjiti grešku, odnosno funkciju greške. U neuronskoj mreži jedino na što možemo imati utjecaj, a što utječe na krajnji rezultat su jačine povezanosti između neurona. Oni se mijenjaju postupkom backpropagationa. Backpropagation je metoda izračunavanja gradijenta funkcije greške kako bi ju se minimiziralo. U backpropagationu bitan je negativan gradijent funkcije jer on pokazuje u kojem „smjeru“ funkcija najbrže opada, odnosno kako treba mijenjati jačinu povezanosti neurona od izlaznih, pa sve do ulaznih neurona (tim redoslijedom).



Slika 8: Backpropagation

Učenje se odvija u serijama (eng. batch). Kako bi se ubrzao postupak učenja uz minimalan gubitak točnosti podaci se dijele u manje serije (npr. po 100 komada). Ako se mreža više puta uči na istom skupu podataka običaj je između učenja pomiješati podatke kako bi se svaki put učilo s drugačijim nizom podataka.



Slika 9: Primjer funkcije više varijabli te njenih minimuma i maksimuma

Kao što je prikazano na slici, neka funkcija ne mora imati jedan minimum. Backpropagation gotovo sigurno neće pronaći optimalno rješenje, ali mogu naći neki lokalni minimum koji je zadovoljavajuće rješenje našeg problema. U koji lokalni minimum će backpropagation smjestiti funkciju greške ovisi o arhitekturi mreže, ulaznim podacima (redoslijedu i vrijednosti), početnim vrijednostima jačine povezanosti neurona te ostalim parametrima (*learning rate, momentum* itd.)

2.2 Primjene neuronskih mreža

Neuronske mreže se primjenjuju kod problema *predikcija* i *klasifikacija*. Iako su prvobitno nastale kao način za rješavanje problema na način kako bi ih rješavao ljudski mozak, kroz vrijeme se prešlo na rješavanje specifičnih zadataka kao što su računalni vid, prepoznavanje glasa, strojno prevođenje, *social network filtering*, medicinske dijagnoze te čak i igranje video igara i *board games*.

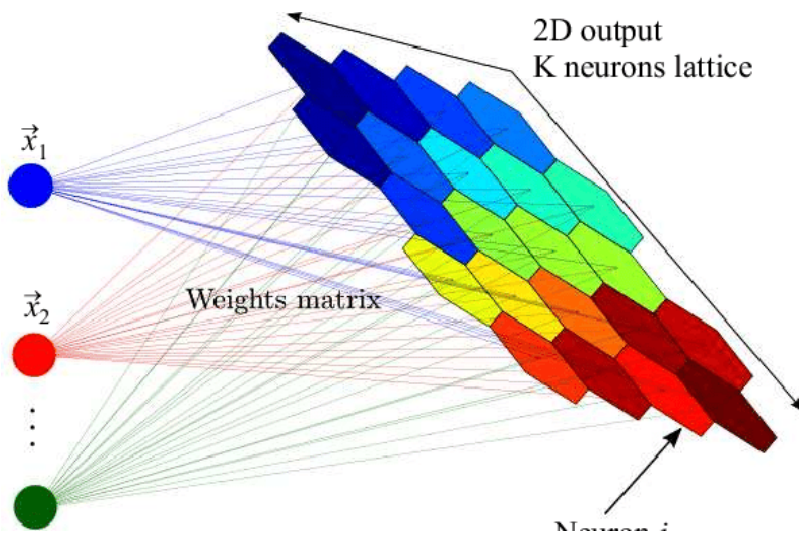
2.3 Vrste neuronskih mreža

Najčešće korištene neuronske mreže su:

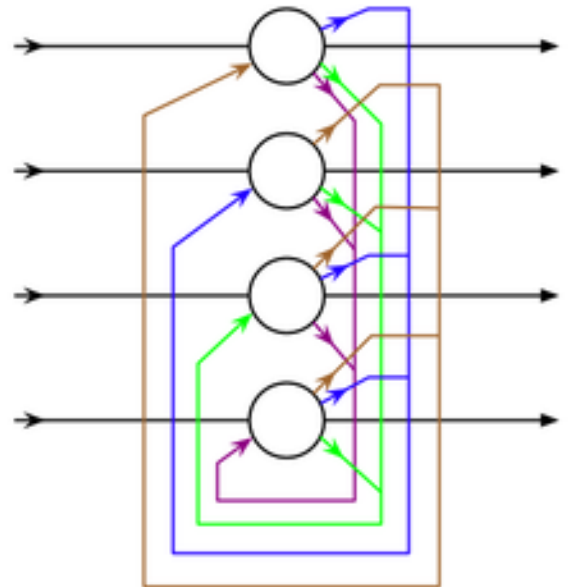
- višeslojne perceptronske mreže (engl. *Multi layer perceptron*)
- samoorganizirajuće karte (engl. *Self organising maps*)
- Hopfieldova neuronska mreža.(engl. *Hopfield network*)

Višeslojne perceptronske mreže su korisne za rješavanje kompleksnih problema:prepoznavanja glasa, slika i strojnog prevođenja. Samoorganizirajuće karte se koriste u meteorologiji, oceanografiji, seizmičkom mapiranju kod traženja nafte i plina. Hopfieldove neuronske mreže se koriste kod prepoznavanja slika, izoštravanja rendgen slika itd.

Ovaj seminar će se baviti isključivo višeslojnim perceptronskim mrežama.



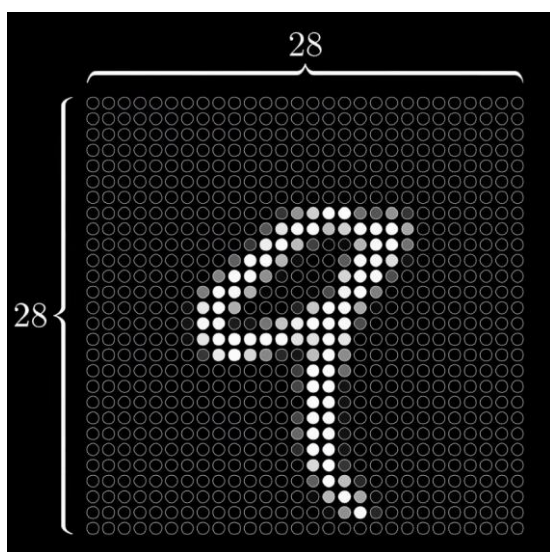
Slika 3: Primjer samoorganizirajuće mape



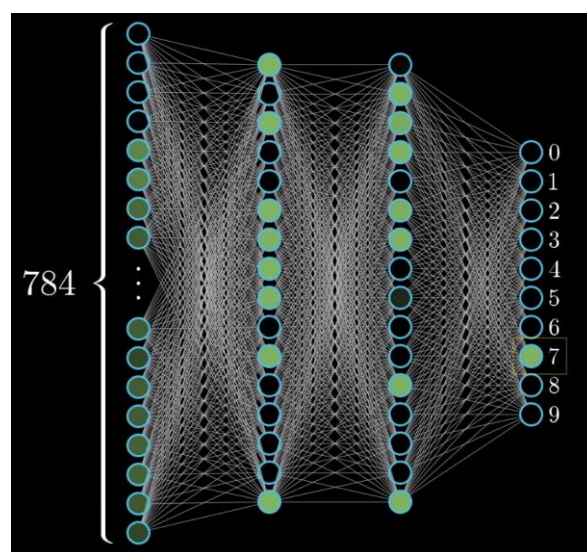
Slika 4: Primjer Hopfieldove mreže

3. Prepoznavanje znamenki korištenjem neuronskih mreža

Za rješavanje ovog problema odabrana je višeslojna perceptronska mreža. Znamenke koje se prepoznaju su slike dimenzija 28x28 pikela u *grayscale* formatu. Svaki pixel ima pripadajući neuron u ulaznom sloju (sveukupno 784), dok izlazni sloj ima 10 neurona (po jedan za svaku znamenku). Skriveni sloj se sastoji od dva sloja po 16 neurona, te su svi neuroni susjednih slojeva međusobno povezani.



Slika 10: Prikaz znamenke u grayscaleu



Slika 11: Arhitektura korištene mreže

Za učenje se koristi dataset „The MNIST database of handwritten digits“, koji se sastoji od 60 000 primjera za učenje i 10 000 testnih primjera. Kao ne-linearna aktivacijska funkcija odabrana je sigmoida što uzrokuje da je vrijednost svakog neurona realni broj između 0 i 1. Rezultat mreže se računa kao redni broj neurona s najvećom vrijednošću u izlaznom sloju. Algoritam za backpropagation je gradijentni spust (eng. *gradient descent*). Ovim algoritmom se nakon određenog broja primjera mreži promijene atributi (nauči ju se da na drugačiji način prepoznaje slike). Želimo da neuron s odgovarajućim indeksom na izlazu poprimi vrijednost koja teži u 1, dok vrijednosti ostalih neurona teže u 0. Za ovu mrežu je nakon testiranja izabrana veličina serija od 50 slika koja uz learning rate 0.1 i 5 uzastopnih učenja (bez miješanja podataka) daje najbolji rezultat. Cilj ovog algoritma je nakon učenja na velikom skupu primjera svesti grešku na minimum, odnosno maksimizirati točnost prepoznavanja znamenki.

4. Rezultati i zaključak

Testiranje točnosti stvorene neuronske mreže napravljeno je u dva dijela modificiranjem varijabli batch size i learning rate. U prvom dijelu podaci se nisu miješali između učenja dok u drugom jesu. Tablica lijevo označava prosječnu točnost, a desna standardnu devijaciju (obje u postocima, zaokruženo na dvije decimale).

Tablica 1 Rezultati bez miješanja

Learn- ing rate \ Batch size	Batch size		Learn- ing rate \ Batch size	Batch size	
	50	100		50	100
1	75,04	75,32	1	2,54	3,57
0.1	78,26	72,34	0.1	1,64	2,71

Tablica 2 Rezultati s miješanjem

Learn- ing rate \ Batch size	Batch size		Learn- ing rate \ Batch size	Batch size	
	50	100		50	100
1	75,1	75,07	1	2,24	4,63
0.1	76,76	72,92	0.1	3,22	2,48

Iz dobivenih podataka se može zaključiti da napravljena neuronska mreža za ovaj konkretan problem ima točnost između 70 i 80 posto (prosječno 75 posto), te jako varira ovisno o početnom stanju mreže i rasporedu ulaznih podataka. Veća točnost se mogla postići raznim tehnikama postavljanja početnih vrijednosti težina, promjenjivim *learning rate*om, tehnikom momentuma, velikim brojem pokušaja uz mijenjanje svih parametara mreže (eng. *trial and error*) itd.

Za točniju statistiku bi trebalo provesti veći broj testova za svako stanje mreže.

5. Sažetak

Neuronske mreže su univerzalni način za aproksimiranje funkcija. One imaju sposobnost učiti i modelirati ne-linearne i kompleksne relacije, što je jako korisno jer su većina relacija u stvarnom svijetu baš takve. Neuronske mreže također mogu i generalizirati, odnosno nakon učenja na nekim podacima i njihovim povezanostima mreža može modelirati i predviđati nad podacima koje nikad nije „vidjela“.

Neke od uporaba umjetnih neuronskih mreža su

- **Prepoznavanje**-neuronske mreže se mogu naučiti da prepoznaju slike, oblike, znakove i zvuk. Koriste se i u medicini za detekciju raka te u kombinaciji sa satelitskim snimkama za agrikulturu i sigurnost.
- **Predviđanje**-Jedno od natežih i nakompleksnijih predviđanja su financije i burze dionica zbog utjecaja velikog broja faktora (od kojih su neki „skriveni“). Tu u pomoć dolaze neuronske mreže koje, ako su napravljene na pravi način mogu modelirati kompleksne i ne-linearne povezanosti između podataka .

U sklopu ovog seminara je napravljena umjetna neuronska mreža koja ima mogućnost prepoznavanja ručno pisanih znamenki na slikama određenog formata. Najveća postignuta točnost je ~81% i to stanje mreže će se koristiti kod prepoznavanja znamenki sa skinutih slika/slikanih mobitelom. Zbog kompleksnosti i velike količine potrebnog predznanja nisu implementirane napredne tehnike kojima bi se postigla još veća točnost.

6. Literatura

You tube kanal 3blue1brown(ideja i slike):

https://www.youtube.com/playlist?list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi

Dataset:

<http://yann.lecun.com/exdb/mnist/>

Teorija i slike:

https://hr.wikipedia.org/wiki/Umjetna_neuronska_mre%C5%BEa

https://en.wikipedia.org/wiki/Artificial_neural_network#Theoretical_issues

<https://towardsdatascience.com/weight-initialization-techniques-in-neural-networks-26c649eb3b78>

<https://www.cse.unsw.edu.au/~cs9417ml/MLP2/BackPropagation.html>

https://stats.stackexchange.com/questions/47590/what-are-good-initial-weights-in-a-neural-network?fbclid=IwAR3gf7w-TABtaMPU_wHsZMtY3l452mWhcSf18Z80P2vd--hxmr4Vda_mm4Q

<https://medium.com/usf-msds/deep-learning-best-practices-1-weight-initialization-14e5c0295b94>

<https://www.jeremyjordan.me/nn-learning-rate/>

<https://ml-cheatsheet.readthedocs.io/en/latest/backpropagation.html>

<https://towardsdatascience.com/improving-vanilla-gradient-descent-f9d91031ab1d>

<https://towardsdatascience.com/introduction-to-neural-networks-advantages-and-applications-96851bd1a207>

|