# PARAMETRI I HIPERPARAMETRI U UMJETNIM NEURONSKIM MREŽAMA

Već se dosta toga spomenulo i opisalo o parametrima koje umjetna neuronska mreža može naučiti, kao i o hiperparametrima koji su nešto što se za svaku mrežu određuje prije procesa učenja. Međutim, do sada se nije dao koncizni popis što sve neuronska mreža može naučiti.

U ovom poglavlju će se definirati što je to parametar koji neuronska mreža može naučiti. Nakon toga će se opisati primjer kako se broj parametara računa kod tradicionalnih neuronskih mreža i kod konvolucijskih neuronskih mreža te će oba opisa biti potkrijepljena primjerima.

Nakon toga će se opisati što su to hiperparametri i koji je njihov utjecaj na sam proces učenja neuronske mreže kao i na same performanse neuronske mreže

## PARAMETRI

Parametar je svaki pojam čiju optimalnu vrijednost neuronska mreža pokušava naučiti tijekom procesa učenja, to su težine veza i pragovi neurona. Broj parametara se računa za svaki sloj neuronske mreže pojedinačno. Ukupan broj parametara čini zbroj parametara svakog pojedinog sloja neuronske mreže.[1]

Kako bi se izračunao ukupan broj parametara u jednom sloju tradicionalne neuronske mreže, prvo je potrebno izračunati nekoliko stvari:

1. broj ulaza u određeni sloj
2. broj izlaza iz određenog sloja
3. ako sloj sadrži pragove, koliko ih sadrži.[1]

Ovdje je potrebno napomenuti da se misli na potpuno povezanu mrežu (eng. *fully.connected network*) koja je sačinjena od gustih slojeva(eng. *dense layers*).

Ukupan broj parametara za dani sloj se računa sljedećom formulom:

Na primjer, neka je dana potpuno povezana neuronska mreža koja ima 3 sloja sa sljedećim brojem čvorova:

1. ulazni sloj s 3 čvora
2. skriveni sloj s 4 čvora
3. izlazni sloj s 3 čvora

NAPRAVIT SLIKU

Također, neka mreža sadržava pragove. To znači da svaki sloj u skrivenom i izlaznom sloju sadrži prag.

Broj parametara ove neuronske mreže se računa na sljedeći način:

1. Ulazni sloj ne sadrži parametre koji se mogu naučiti jer on sadrži samo ulazne podatke.
2. Broj ulaza u skriveni sloj jednak je broju čvorova prethodnog, odnosno ulaznog, sloja. To je broj 3. Broj izlaza iz skrivenog sloja jednak je broju čvorova u skrivenom sloju, 4. Ta dva broja se pomnože i dobivena vrijednost jednaka je broju težina, 12. Kako bi se dobio ukupan broj parametara za skriveni sloj, umnožak se zbraja s brojem pristranosti koji je opet jedna broju čvorova skrivenog sloja, 4. Ukupan broj parametara u skrivenom sloju je 16.
3. Na isti način se računa broj parametara za izlazni sloj i dobije se broj od parametara koji se mogu naučiti u izlaznom sloju.
4. Zbroje se rezultati skrivenog i izlaznog sloja te se dobije ukupno 31 parametar koji ova neuronska mreža može naučiti tijekom procesa učenja.

Kao i kod tradicionalne neuronske mreže, broj parametara u konvolucijskoj mreži se računa za svaki sloj posebno. Ukupan broj parametara se dobije zbrajanjem parametara iz svakog pojedinog sloja.

Međutim, konvolucijski slojevi nisu gusti slojevi te sadrže filtere. O tome je potrebno voditi računa jer se formula za izračun broja parametara modificira s obzirom na to:

1. je li prethodni sloj konvolucijskog sloja bio gusti sloj. Ako je, ulaz u konvolucijski sloj je broj čvorova koji se nalaze u prethodnom sloju.
2. Ako je prethodni sloj određenog konvolucijskog sloja također konvolucijski sloj, broj ulaza u taj određeni sloj je jednak broju filtera koje posjeduje taj prethodni konvolucijski sloj.[1]

Također, broj izlaza iz konvolucijskog sloja jednak je umnošku broja filtera s njihovim dimenzijama. Konačno, broj pragova jednak je broju filtera u danom konvolucijskom sloju.

Na primjer, neka konvolucijska neuronska mreža sadrži ulazni sloj, dva skrivena konvolucijska sloja i jedan gusti sloj kao izlazni sloj:

1. ulazni sloj koji sadrži podatke dimenzija 10x10x3
2. prvi skriveni konvolucijski sloj koji sadrži 3 filtera dimenzija 3x3
3. drugi skriveni konvolucijski sloj koji sadrži 2 filtera dimenzija 3x3
4. gusti izlazni sloj koji sadrži 2 izlazna neurona

Broj parametara se računa na sljedeći način:

1. Kao i kod tradicionalnih neuronskih mreža, ulazni sloj nema parametre koji se mogu naučiti jer on samo sadrži ulazne podatke.
2. Broj ulaza u prvi konvolucijski sloj je 3. Broj izlaza iz tog sloja je umnožak broja filtera s njihovim dimenzijama, odnosno . Broj ulaza u sloj se pomnoži s brojem izlaza iz sloja i dobije se težina u prvom konvolucijskom sloju koja se može naučiti. Ukupan broj parametara jednak je .
3. Broj ulaza u drugi konvolucijski sloj je 3 jer prethodni sloj sadrži 3 filtera. Broj izlaza iz drugog konvolucijskog sloja jednak je . Umnožak ulaza i izlaza daje broj . Ukupan broj parametara drugog konvolucijskog sloja je .
4. Neka se koristi *zero padding* kako bi se sačuvale dimenzije ulaznog kanala. Onda je broj ulaza u izlazni sloj jednak . Broj izlaza jednak je broju neurona u izlaznom sloju, 2. Konačan broj parametara za izlazni sloj jednak je .

Ukupan broj parametara koje konvolucijska mreža mora naučiti tijekom procesa učenja je .

## HIPERPARAMETRI

Hiperparametri su vanjski parametri neuronske mreže koje postavlja operator neuronske mreže. Oni imaju veliki utjecaj na preciznost neuronske mreže i njenu sposobnost učenja. Iako postoji više načina pronalaska optimalnih vrijednosti hiperparametara, najjednostavniji način je metoda pokušaja i pogreške.[2]

Hiperparametri su sljedeći:

1. Broj skrivenih slojeva – dodavanjem više skrivenih slojeva se povećava preciznost predviđanja, ali se isto tako povećava rizik od prenaučenosti.
2. *Dropout* – koliki će se postotak nasumično odabranih neurona ignorirati tijekom svake epohe kako bi se spriječila prenaučenost. Ali s druge strane, preveliki postotak ignoriranih neurona može dovesti do podnaučenosti.
3. Aktivacijska funkcija – koja se aktivacijska funkcija koristi za obradu ulaznih primjera. Aktivacijska funkcija utječe na preciznost i brzinu učenja mreže.
4. Inicijalizacija težina – prije samog učenja neuronske mreže, potrebno je njenim vezama pridijeliti neke težine. Dvije su osnovne opcije, postavljanje svih težina na nulu ili postavljanje nasumičnih vrijednosti. S obje opcije postoji opasnost od problema nestajućeg i eksplodirajućeg gradijenta. Jedna od alternativa takvih inicijalizacija težina jest Xavier-ova heuristička inicijalizacija težina.
5. Stopa učenja neuronske mreže – koliko se brzo algoritam propagacije prema natrag približava minimalnom gubitku. Niža stopa učenja produljuje vrijeme učenja, ali zato izbjegava mogućnost dobivanja pretjeranog rezultata (eng. *overshooting*).
6. Broj epoha, veličina grupe – ovi parametri određuju kojom će se brzinom primjeri iz skupa za učenje prosljeđivati mreži.
7. Algoritam optimizacije – osnovna opcija jest korištenje stohastičkog gradijentnog spusta. Alternative su algoritmi kao što su Momentum, Nester-ov akcelerirani gradijent, Adam ili AdaDelta.[2]

[1] [www.deeplizard.com](http://www.deeplizard.com)

[2] <https://missinglink.ai/guides/neural-network-concepts/hyperparameters-optimization-methods-and-real-world-model-management/>