# POVRATNA PROPAGACIJA

https://deeplizard.com/

Ovo poglavlje je posvećeno povratnoj propagaciji (eng. *backpropagation*) i njenoj ulozi u procesu treniranja neuralne mreže.

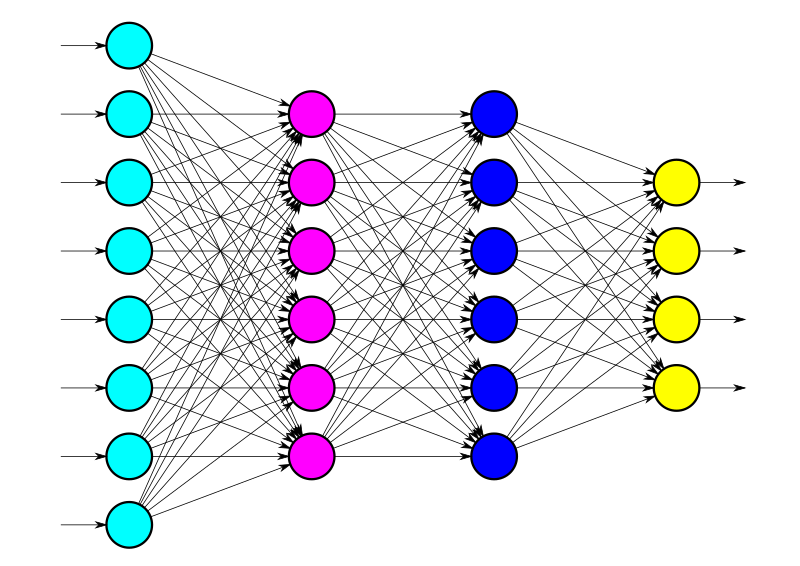
Prvo će se ukratko ponoviti neke bitne točke u vezi stohastičkog gradijentnog spusta koje su se spominjale u prethodnim poglavljima. Nakon toga će se pokazati gdje točno pripada povratna propagacija te će se veliki dio poglavlja baviti intuicijom što povratna propagacija zapravo radi.

## PREGLED STOHASTIČKOG GRADIJENTNOG SPUSTA (SGD)

Kao što je već spomenuto, tijekom procesa treniranja, stohastički gradijentni spust radi na tome da minimizira funkciju gubitka na način da ažurira težine veza nakon svake epohe. Također je spomenuto kako se to radi na način da se prvo izračuna gradijent, ili derivacija, funkcije gubitka u odnosu na težine u modelu. Međutim, ništa detaljnije nije bilo opisano. Ovaj se čin računanja gradijenata, s ciljem da se ažuriraju težine, odvija u procesu koji se naziva 'povratna propagacija'.

## PROPAGACIJA PREMA NAPRIJED

Neka je dana proizvoljna neuralna mreža koja sadrži dva skrivena sloja. Zbog jednostavnosti, primjer će se temeljiti na jednom uzorku podatka koji se prosljeđuje mreži.



Slika 1: neuralna mreža s dva skrivena sloja

Kao podsjetnik, tijekom procesa učenja, svaki put, kada se podatak proslijedi modelu, podatak se prosljeđuje kroz mrežu prema naprijed dok ne stigne do njenog izlaznog sloja. Isto tako, svaki čvor u modelu prima ulaz iz prethodnog sloja i ulaz čini ponderirana suma koju čini težina svake veze pomnožena s izlaznom vrijednošću svakog pojedinog čvora u prethodnom sloju. Ta ponderirana suma se, kao ulazna vrijednost, prosljeđuje aktivacijskoj funkciji. Vrijednost, koju vraća aktivacijska funkcija, je izlazna vrijednost određenog čvora koja se zatim prosljeđuje kao dio ulazne vrijednosti u čvor koji se nalazi u sljedećem sloju. Ovaj se proces ponavlja za svaki sloj u mreži dok se ne stigne do izlaznog sloja i on se naziva propagacija 'prema naprijed'.

Kada se dosegne izlazni sloj, dobiveni rezultat predstavlja izlaznu vrijednost, odnosno predviđanje modela, za danu ulaznu vrijednost.

## RAČUNANJE GUBITKA

Kada su dobiveni izlazni rezultati, računa se gubitak modela. Način, na koji se računa gubitak, ovisi o izabranoj funkciji gubitka koja će se koristiti. Neka, radi jednostavnosti, funkcija gubitka samo izračunava koliko je model udaljen od ispravne klasifikacije za danu ulaznu vrijednost. O gubitku se može misliti i kao o razlici između onoga što je model predvidio i onoga što je zapravo ulazna vrijednost.

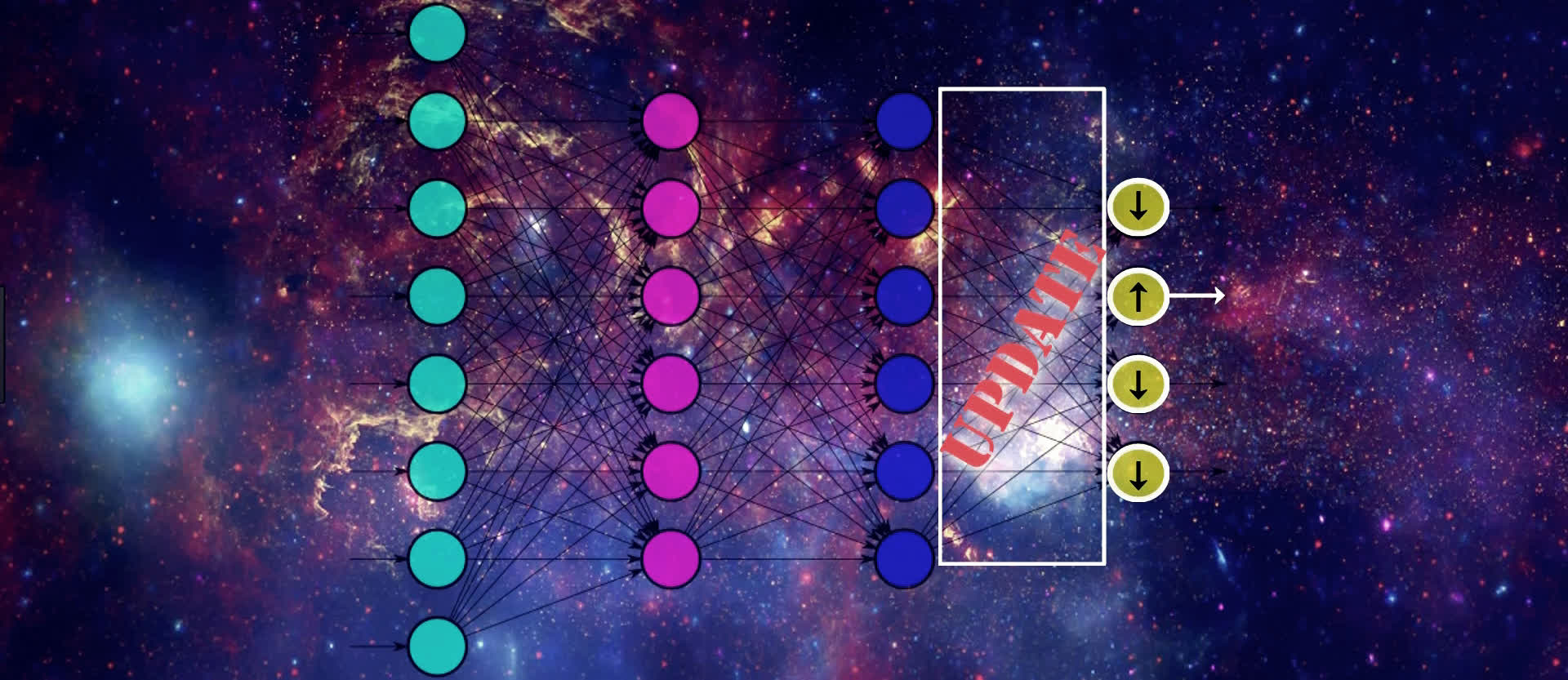
Kao što je spomenuto, cilj gradijentnog spusta jest minimizirati funkciju gubitka. To se radi uzimanjem derivacije, odnosno gradijenta, funkcije gubitka u odnosu na težine u modelu.

Povratna propagacija jest alat koji koristi gradijentni spust kako bi izračunao gradijent funkcije gubitka.

## INTUICIJA POVRATNE PROPAGACIJE

Kako bi ažurirao težine veza, gradijentni spust prvo gleda izlazne vrijednosti aktivacijskih funkcija u izlaznom sloju.

Neka izlazni čvor, kojem je pridružena strelica koja pokazuje prema gore, predstavlja ispravnu izlaznu vrijednost za danu ulaznu vrijednost. U tom slučaju, gradijentni spust 'razumije' da se vrijednost tog čvora treba povećati, a vrijednosti ostalih čvorova se moraju smanjiti. Na taj način se smanjuje gubitak za danu ulaznu vrijednost.

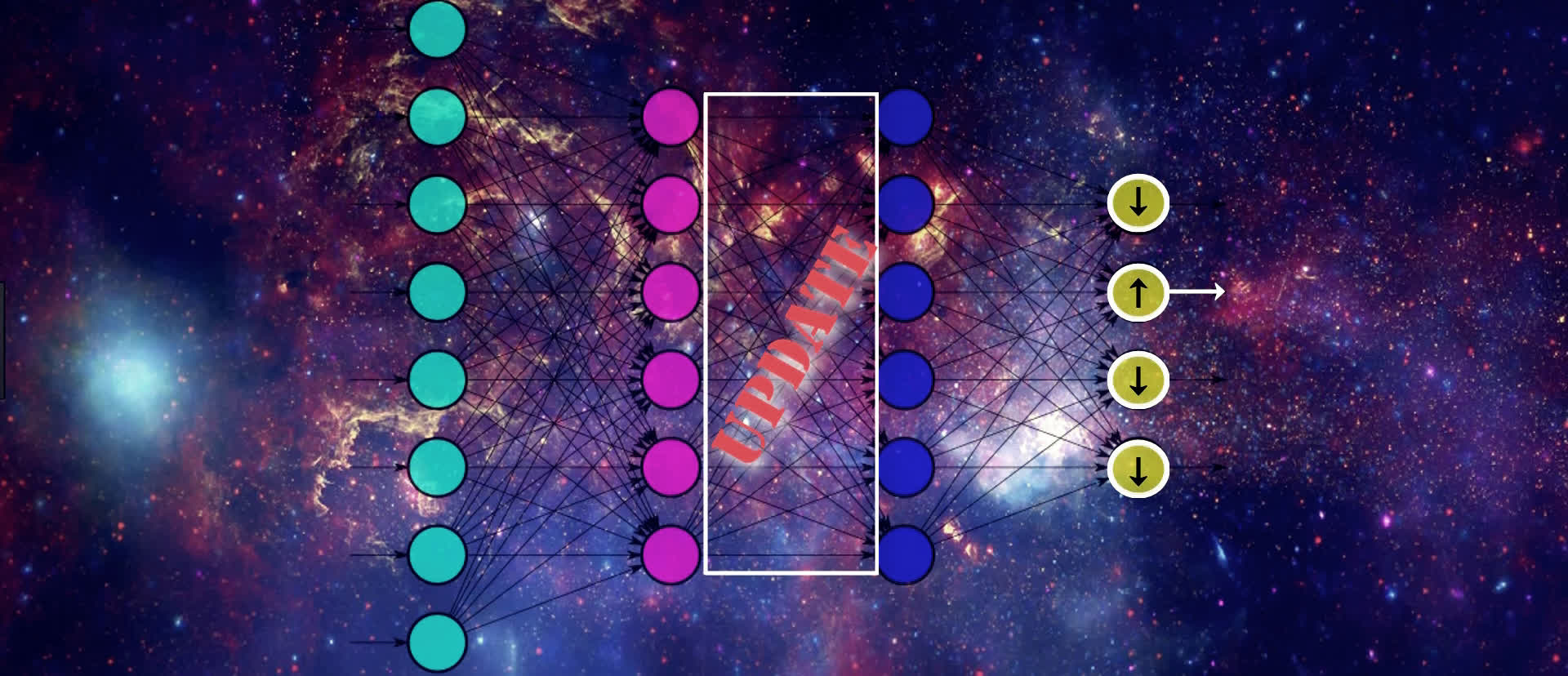


Slika 2: cilj gradijentnog spusta je povećati vrijednost ispravnog izlaznog čvora

Zna se da vrijednosti izlaznih čvorova dolaze od ponderiranih suma težina, koje su pomnožene s određenim vrijednostima čvorova iz prethodnog sloja, i da se ta ponderirana suma prosljeđuje aktivacijskoj funkciji za svaki određeni čvor u izlaznom sloju.

Kada bi se htjele promijeniti vrijednosti izlaznih čvorova, jedan način kako bi se to moglo napraviti jest da se promijene težine veza koje spajaju izlazni sloj s njemu prethodnim slojem. Drugi način na koji bi se to mogli napraviti jest da se promijeni izlazna vrijednost aktivacijskih funkcija koje se nalaze u sloju koji prethodi izlaznom sloju.

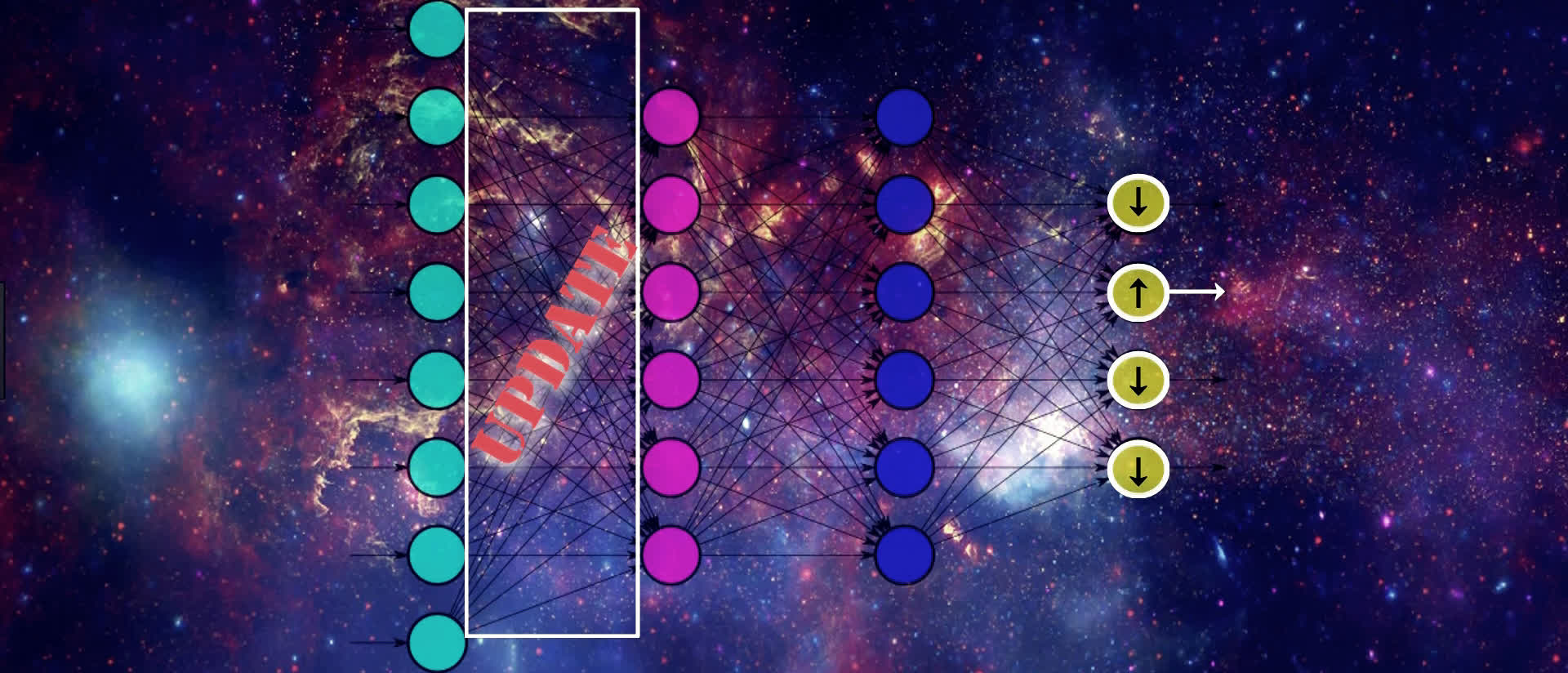
Nije moguće direktno promijeniti izlaz iz aktivacijskih funkcija jer on ovisi o težinama koje prethode predzadnjem sloju u neuralnoj mreži. Međutim, moguće je indirektno utjecati na promjenu tog izlaza tako da se cijeli izračun prebaci za još jedan sloj unatrag (sada se govori o težinama koje spajaju predzadnji sloj s njemu prethodnim slojem) i tako da se promijene težine veza na isti način koji je bio opisan za izlazni sloj.



Slika 3: kako bi se promijeno izlaz iz aktivacijske funkcije, potrebno je promijeniti težine tog sloja

Cijeli se proces pomicanja unatrag, kako bi se promijenile težine i na taj način utjecalo na izlaz aktivacijskih funkcija, ponavlja sve dok se ne dosegne ulazni sloj modela. Vrijednosti ulaznih čvorova se ne mogu promijeniti jer one sadrže prave podatke koji se prosljeđuju neuralnoj mreži.

Ukratko, gradijentni spust se pomiče natrag kroz mrežu, mijenja težine veza s desna na lijevo kako bi pomakao vrijednosti izlaznih čvorova u onom smjeru koji bi smanjio gubitak. To znači da, za jedan primjerak, stohastički gradijentni spust nastoji povećati vrijednost ispravnog čvora i smanjiti vrijednosti neispravnih čvorova na onaj način koji će najefikasnije smanjiti gubitak.



Slika 4: težine se mijenjaju s desna na lijevo sve dok se ne dosegne ulazni sloj

Cijeli se ovaj proces, koji je pokazan na primjeru jednog ulaznog podatka, odvija i u slučaju više ulaznih podataka u seriji. Razlika je u tome što će konačne promijene težina biti jednake prosječnoj izračunatoj promjeni težina za svaki pojedini ulaz.

## MATEMATIČKA NOTACIJA POVRATNE PROPAGACIJE

Sljedeća tablica definira matematičke notacije koje će se koristiti kako bi se matematički opisao proces povratne propagacije.

|  |  |
| --- | --- |
| **simbol** | **definicija** |
| L | broj slojeva u mreži |
| l | indeks (broj) sloja |
| j | indeks čvora u sloju l |
| k | indeks čvora u sloju l-1 |
|  | vrijednost čvora j u izlaznom čvoru L kod jednog uzorka za treniranje |
|  | funkcija gubitka kod jednog uzorka za treniranje |
|  | vektor težina koje povezuju čvorove iz l-1 a čvorom j u sloju l |
|  | težina koja povezuje čvor k u sloju l-1 s čvorom j u sloju l |
|  | ulazna vrijednost u čvor j koji se nalazi u sloju l |
|  | aktivacijska funkcija koja se koristi u sloju l |
|  | izlaz iz aktivacijske funkcije čvora j koji se nalazi u sloju l |

Kao što se može primijetiti, sve definicije iz prethodne tablice ovise o sljedećim indeksima:

|  |  |
| --- | --- |
| **simbol** | **definicija** |
| l | indeks sloja |
| j | indeks čvora u sloju l |
| k | indeks čvora u sloju l-1 |

## MATEMATIČKE OBZERVACIJE ZA POVRATNU PROPAGACIJU

U ovom poglavlju će se prikazati kako se funkcija gubitka može matematički opisati, a nakon toga će se prikazati kako se matematički mogu prikazati ulazi i izlazi za bilo koji dani čvor. Na kraju će se definirati metoda diferenciranja funkcije gubitka preko povratne propagacije.

### FUNKCIJA GUBITKA

Funkcija

Je kvadrat razlika izlazne vrijednosti aktivacijske funkcije i željenog izlaza za čvor *j* u izlaznom sloju *L*. Ovo se može protumačiti i kao gubitak čvora *j* u sloju *L*. Kako bi se izračunao ukupni gubitak, potrebno je zbrojiti rezultat kvadrata razlike za svaki čvor *j* u izlaznom sloju *L*.

Matematički to izgleda na sljedeći način:

### ULAZNA VRIJEDNOST

Ulazna vrijednost čvora *j*, koji se nalazi u sloju *l*, jest ponderirana suma izlaza aktivacijskih funkcija iz prethodnog sloja *l-1*.

Individualna ulazna vrijednost izgleda ovako:

Tako, za dani čvor *j* u sloju *l*, izlaz izgleda ovako:

### IZLAZ IZ AKTIVACIJSKE FUNKCIJE

Izlaz iz aktivacijske funkcije za dani čvor *j* u sloju *l* je rezultat ulazne vrijednosti koja se prosljeđuje aktivacijskoj funkciji po vlastitom izboru .

Matematički izraz za izlaz iz aktivacijske funkcije za dani čvor *j* u sloju *l* je:

### FUNKCIJA GUBITKA KAO KOMPOZICIJA FUNKCIJA

Kao što je već napisano, definicija funkcije gubitka izgleda ovako:

Gubitak jednog čvora *j* u izlaznom sloju *L* se matematički izražava na sljedeći način:

Kako je funkcija funkcija za jedan izlazni čvor *j* u sloju *L*, funkcija se može opisati kao funkcija od :

.

Promatranjem definicije za , može se primijetiti da ­­­­­ ovisi i o . Kako je konstanta, ­­­­­ se promatra samo kao funkcija od , dok se smatra parametrom koji pomaže definirati ovu funkciju.

Aktivacijski izlaz čvora *j* u izlaznom sloju *L* je funkcija izlaza za čvor *j*. Na temelju jedne od prethodnih obzervacija, to se sada može izraziti kao:

.

Ulazna vrijednost čvora *j* je funkcija svih težina koje su vezane za čvor *j*. Tako se može izraziti kao funkcija od :

.

Tako da vrijedi sljedeći izraz:

.

Ovaj izraz je pokazao da je kompozicija funkcija.

Nadalje, vrijedi:

To znači da je i ukupan gubitak mreže za jedan ulaz kompozicija funkcija. Ovo je korisno znati kako bi se razumjela derivacija . Kako bi se derivirala kompozicija funkcija, koristi se takozvano 'lančano pravilo'.

## RAČUNANJE GRADIJENTA

Kao podsjetnik, kako bi stohastički gradijentni spust promijenio težine u mreži, prvo treba izračunati gradijent gubitka u odnosu na te težine.

Težina koja spaja čvor *2* u sloju *L-1* sa čvorom *1* u čvoru *L* se označava na sljedeći način:

.

Derivacija funkcije gubitka u odnosu na ovu konkretnu težinu se izražava kao:

S obzirom da ovisi o , ovisi o i ovisi o , lančano pravilo kaže da, kako bi se derivirala u odnosu na , uzima se derivacija kompozicije funkcija.

To je matematički izraženo na sljedeći način:

Neka se promotri prvi izraz, .

Kao što je već spomenuto, zna se da vrijedi:

Stoga vrijedi:

Kada se proširi suma, može se vidjeti da vrijedi:

=

To znači da će gubitak mreže, u slučaju jedne ulazne vrijednosti, reagirati malom promjenom u aktivacijskom izlazu čvora 1 u sloju *L*. Količina promjene jednaka je razlici aktivacijskog izlaza za čvor 1 i željenog izlaza za čvor 1 koja je potom umnožena s brojem 2.

Neka se promotri drugi izraz:

Zna se da za svaki čvor *j* u izlaznom sloju *L* vrijedi:

i, s obzirom da je j = 1, vrijedi:

Stoga vrijedi:

Dakle, to je samo direktna derivacija od jer je direktna funkcija od .

Neka se, konačno, promotri treći izraz: .

Zna se da za svaki čvor *j* u izlaznom sloju *L* vrijedi:

S obzirom da je j = 1, vrijedi:

Stoga vrijedi:

= .

Kada se proširi suma, dobije se:

To znači da će ulazni čvor 1 u sloju *L* reagirati na promjenu težine količinom koja je jednaka aktivacijskom izlazu čvora 2 u prethodnom sloju *L-1*.

Udruživanjem svih izraza, konačni rezultat je:

Opisano je kako se računa derivacija funkcije gubitka u odnosu na individualnu težinu kada je u pitanju jedan ulaz za treniranje. Kako bi se izračunala derivacija funkcije gubitka u odnosu na ovu istu težinu za svih *n* ulaznih vrijednosti iz skupa za treniranje, računa se prosječna derivacija funkcije gubitka u odnosu na svih *n* ulaza iz skupa za treniranje.

Matematički se to može izraziti kao:

Tada bi se cijeli ovaj proces ponovio za svaku težinu u mreži kako bi se izračunala derivacija funkcije gubitka u odnosu na svaku pojedinu težinu veze u mreži.

## POVRAT U POVRATNOJ PROPAGACIJI

Do sada je bilo opisano kako se može izračunati gradijent funkcije gubitka korištenjem povratne propagacije. Međutim, nije spomenuto gdje točno dolazi taj pokret prema natrag u povratnoj propagaciji.

U prethodnom potpoglavlju je prikazano kako se računa gradijent funkcije gubitka u odnosu na bilo koju težinu u mreži. Primjer je dan na jednoj konkretnoj težini u izlaznom sloju u mreži, . Nakon toga se taj proces poopćio jer se on može primijeniti na svaku težinu u vezi pojedinačno.

Za ovu konkretnu težinu, pokazalo se da je derivacija gubitka u odnosu na tu težinu jednaka sljedećem izrazu

Kada je u pitanju težina koja se ne nalazi u izlaznom sloju, na primjer , formula za gradijent gubitka, koja je ponovljena maloprije, može se iskoristiti i za ovu težinu. Formula tada glasi:

Jedina razlika je ta što se razlikuju natpisi, jer se u ovom slučaju radi s trećim slojem u mreži, koji se označava s *L-1,* i indeksi jer se radi s težinom koja povezuje drugi čvor u drugom sloju s drugim čvorom u trećem sloju.

Međutim, iako su drugi i treći izraz na desnoj strani formule isti i računat će se na već definirani način, prvi izraz na desnoj strani jednakosti jest derivacija gubitka u odnosu na aktivacijski izlaz koji će se trebati izračunati na drugačiji način.

Kada se računala derivacija gubitka u odnosu na težinu u izlaznom sloju, prvi izraz je činila derivacija gubitka u odnosu na aktivacijski izlaz čvora koji se nalazi u izlaznom sloju. Kao što je već objašnjeno, gubitak je direktna funkcija aktivacijskog izlaza za sve čvorove u izlaznom sloju. Gubitak je zbroj kvadriranih pogreški između pravih oznaka za podatke i aktivacijskih izlaza čvorova u izlaznom sloju.

Tako, kada se računa derivacija gubitka u odnosu na težinu koja se nalazi u sloju *L-1*, prvi izraz s desne strane jest derivacija gubitka u odnosu na težinu koja se ne nalazi u izlaznom sloju *L*, već u sloju *L-1*. U ovom slučaju, za razliku od aktivacijskog izlaza čvora u izlaznom sloju, funkcija gubitka nije direktna funkcija izlaza čvora koji se nalazi u sloju *L-1*. Potrebno je uzeti u obzir gdje se nalazi aktivacijski izlaz u mreži i gdje se računa gubitak na kraju mreže. Izlaz se ne prosljeđuje direktno gubitku.

Neka se usporede sljedeće funkcije:

i

Kako bi se prikazao način na koji se može izračunati derivacija funkcije u gubitka u odnosu na aktivacijski izlaz čvora koji se nalazi bilo gdje u mreži, a ne samo u izlaznom sloju, radit će se s jednim aktivacijskim izlazom. Konkretno, aktivacijskim izlazom čvora 2 u sloju *L-1*.

Aktivacijski izlaz je označen kao . Parcijalna derivacija funkcije gubitka u odnosu na ovaj aktivacijski izlaz označena je kao .

Neka se promotri da, za svaki čvor *j* u sloju *L*, funkcija gubitka C0 ovisi o , ovisi o . ovisi o svim težinama, koje čvor *j* spajaju s prethodnim slojem *L-1*, kao i svim aktivacijskim izlazima iz sloja *L-1*. To onda znači da ovisi o .

Sada aktivacijski izlaz za svaki od čvorova u izlaznom sloju ovisi o ulazu u svaki od tih čvorova. Nadalje, ulaz u svaki od čvorova u izlaznom sloju ovisi i o težinama koje povezuju svaki od tih čvorova s prethodnim slojem *L-1* i o aktivacijskim izlazima čvorova koji se nalaze u tom sloju *L-1*.

Vraćajući se na konkretni primjer, može se vidjeti kako ulaz u svaki čvoru u izlaznom sloju ovisi o aktivacijskom izlazu čvora 2 u sloju *L-1*.

Tako je funkcija gubitka opet kompozicija funkcija te se računa derivacija u odnosu na aktivacijski izlaz s kojim se radi. Koristi se lančano pravilo koje kaže da, kako bi izračunali derivaciju u odnosu na , uzima se derivacija kompozicije funkcija koja sada izgleda na sljedeći način:

Ovo je suma za svaki čvor j u izlaznom sloju L. Formula izgleda skoro identično kao i ona za izračun derivacije funkcije gubitka u odnosu na određenu težinu s najvećom razlikom da ova formula sadrži izraz za sumu. Razlog zašto nova formula u svom izrazu sadrži sumu je taj što će aktivacijski izlaz prethodnog sloja utjecati na svaki pojedini ulaz u svaki pojedini čvor j u sljedećem sloju L.

Još jedna od razlika je treći izraz u formuli koji računa derivaciju ulaza u bilo koji čvor j u izlaznom sloju L u odnosu na aktivacijski izlaz iz čvora 2 u sloju L-1, .

Za svaki čvor j u sloju L vrijedi:

Stoga se izraz može zamijeniti izrazom s desne strane jednakosti i uvrstiti u izraz koji računa derivaciju ulaza u bilo koji čvor j u izlaznom sloju L u odnosu na aktivacijski izlaz iz čvora 2 u sloju L-1 te se dobije sljedeće:

Proširenjem sume se dobije:

Zbog toga što je operacija sume linearna operacija, može se uzetu derivacija svakog pojedinog izraza u odnosu na . Međutim, samo jedan izraz sadrži izraz . To znači da, kada se računaju derivacije koje ne sadrže izraz , vrijednosti tih izraza će biti jednaka 0. Rezultat će se dobiti deriviranjem izraza koji sadrži izraz .

Taj rezultat govori da će ulaz u svaki čvor j u sloju L odgovarati promjeni aktivacijskog izlaza čvora 2 u sloju L-1 količinom jednakom težini veze koja povezuje čvor 2 u sloju L-1 s čvorom j u sloju L.

### SPAJANJE IZRAZA

Tako konačna formula za izračun derivacije funkcije gubitka u odnosu na aktivacijski izlaz za dani primjer izgleda sljedeće:

Ukratko, potrebno je prvo izračunati derivacije koje ovise o komponentama koje se nalaze kasnije u mreži i onda iskoristiti te derivacije kod računanja gradijenta funkcije gubitka u odnosu na težine koje se nalaze kasnije u mreži. To se postiže tako što se lančano pravilo primjenjuje unatrag kroz mrežu.

### PROSJEČNA DERIVACIJA FUNKCIJE GUBITKA

Kako bi se pronašla derivacija funkcije gubitka u odnosu na isti aktivacijski izlaz za svih n uzoraka za treniranje, računa se prosječna derivacija funkcije gubitka za svih n uzoraka za treniranje. To se matematički može izraziti na sljedeći način:

## NESTAJUĆI I EKSPLODIRAJUĆI GRADIJENT

Nažalost, tijekom treniranja se mogu javiti dva česta problema. Prvi je problem nestajućeg gradijenta (eng. *vanishing gradient problem*). Drugi je problem eksplodirajućeg gradijenta (eng. *exploding gradient problem*).

Općenito, problem nestajućeg gradijenta uzrokuje velike poteškoće tijekom treniranja neuronske mreže. To je problem koji vezan za težine koje se nalaze između ranijih slojeva u mreži.

Kao što je spomenuto u prethodnom poglavlju, stohastički gradijentni spust računa vrijednost gradijenta u odnosu na težine u neuronskoj mreži s ciljem da smanji vrijednost funkcije gubitka. U nekim situacijama se dogodi da vrijednost gradijenta u odnosu na težine, koje se nalaze u prvim slojevima u mreži, postane jako malena, gotovo nestajuća. Pod pojmom male vrijednosti se misli na one vrijednosti koje su manje od 1. Nadalje, ako je vrijednost gradijenta tako mala, onda će i promjena težina biti jako mala.

Tako mala promjena težine neće dovoljno utjecati na mrežu kako bi se smanjio gubitak jer se mreža jedva pomaknula s mjesta na kojem je bila prije promjene težina. Posljedica je da je težina određene veze na neki način 'zapela' na mjestu i jako će sporo mijenjati svoju težinu kako bi se približila svojoj optimalnoj vrijednosti. Tako se smanjuje i sama sposobnost mreže da uči.

Vraćajući se na matematičke izvode za propagaciju prema natrag, može se primijetiti da, što se veza nalazi ranije u mreži, bit će potrebno više parametara kako bi se izračunao njen gradijent. Stoga, ako više tih parametara ima vrijednost manju od 1, umnožak tih parametara će biti još manji. Dobivena mala vrijednost se zatim koristi kako bi se promijenila težina određene veze. Ali, prije nego se dobivena vrijednost iskoristi za promjenu težine, ona se množi sa stopom učenja koja sama po sebi ima malu vrijednost, između 0.01 i 0.0001. Tako je ukupan rezultat još manji broj. Onda se taj broj oduzima od stare težine kako bi se dobila nova težina.

Ovako, sama težina veze postaje na neki način 'zaglavljena'. Drugim riječima, jako se sporo odmiče od svoje stare vrijednosti i na taj način ne pomaže u minimiziranju gubitka.

S druge strane postoji problem eksplodirajućeg gradijenta. Taj problem se javlja kada su parametri potrebni za izračun gradijenta u odnosu na određenu težinu, koja se nalazi rano u mreži, puno veći od jedan.

Kao i kod problema nestajućeg gradijenta, kada se množi veliki broj parametara, čija je vrijednost puno veća od 1, vrijednost gradijenta će 'eksplodirati'. Međutim, umjesto da se težina jedva pomakne s mjesta, njena će se vrijednost jako promijeniti. Toliko će se promijeniti da može lako 'preskočiti' svoju optimalnu vrijednost.