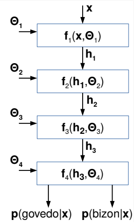
Konvolucijski modeli

Josip Krapac i Siniša Šegvić

Motivacija: razlučiti goveda od bizona



Duboki model ima šansu naučiti značajke koje reagiraju na dijelove objekata, npr: [grba?, mali rogovi?, divljina?, ...]

- bizoni: [DA, DA, DA, ...], goveda: [NE, NE, NE, ...]

Potpuno povezani modeli su u opasnosti da nauče šum jer:

- translatirana slika potpuno različita od originala
- ključne značajke određene lokalnim susjedstvima
- model treba odvojeno naučiti svaku translaciju

Pregled

- Što su konvolucijski modeli?
- Što je konvolucija?
- Zašto konvolucija?
- Sažimanje (*engl. pooling*) i nadopunjavanje.

Što su konvolucijski modeli?

Modeli specijalizirani za podatke s topologijom rešetke

- topologija: oblik strukture definiran relacijom susjedstva

Tipični primjeri:

- vremenski slijed (1D), slika (2D), volumen (3D)

Jednostavna definicija: konvolucijski model ima najmanje jedan konvolucijski sloj umjesto potpuno povezanog sloja.

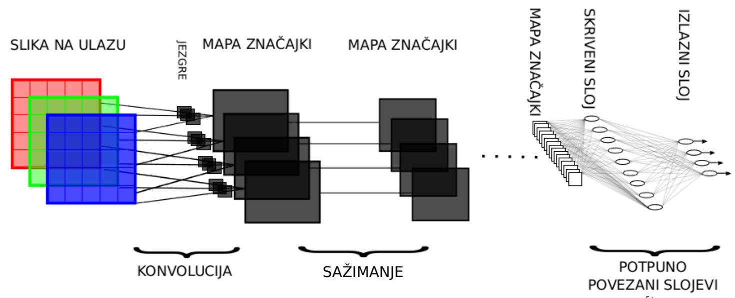
- pored konvolucijskih slojeva u pravilu koristimo slojeve sažimanja i aktivacijske funkcije (ReLU)

Pogledajmo značenje riječi convoluted (nomen est omen):

- extremely complex and difficult to follow
- intricately folded, twisted, or coiled

Što su konvolucijski modeli?

Klasična struktura konvolucijskog modela (LeNet-5):



[lecun89nc,vukotic14ms]

- konvolucijski slojevi transformiraju tenzore trećeg reda:
  - dvije prostorne, jedna "semantička" dimenzija
  - mi ćemo prvo pretpostaviti da je semantička dimenzija 1
- transformacije su lokalne: "pikseli" izlaza ovise o lokalnom susjedstvu piksela ulaza
- težine su tenzori četvrtog reda (!)

Što je to konvolucija?

Konvoluciju definiramo kao skalarni produkt jedne funkcije s obzirom na posmaknutu i reflektiranu drugu funkciju:

$$h(t) = (w * x)(t) = \int_{\mathcal{D}(w)} w(\tau)x(t-\tau)d\tau$$

U strojnom učenju, pod konvolucijom najčešće podrazumijevamo unakrsnu korelaciju:

$$h(t) = (w \star x)(t) = \int_{\mathcal{D}(w)} w(\tau)x(t+\tau)d\tau$$

Konvolucija (odnosno unakrsna korelacija) nam je zanimljiva kao diferencijabilna operacija sa slobodnim parametrima

Jednadžbe pokazuju da jezgru w možemo koristiti za ekstrakciju lokalnih značajki iz signala x

Što je to konvolucija?

Primjer: praćenje svemirskog broda laserskim senzorom koji daje izlaz $x(t)$, poziciju broda trenutku t.

- mjerenja su pokvarena šumom, želimo dobiti usrednjenu predikciju h

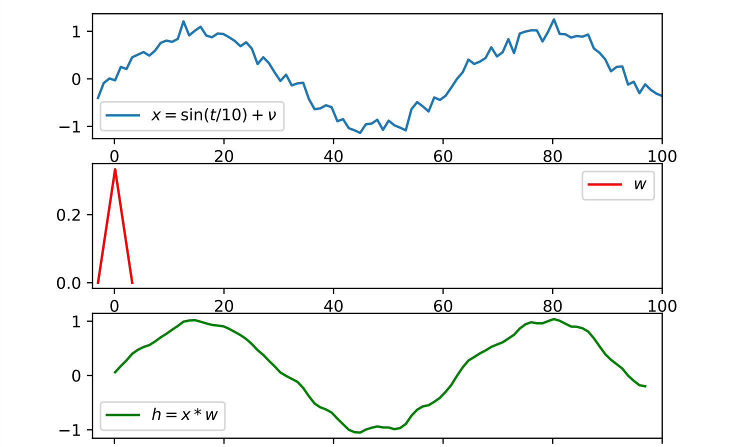
Parametrizirajmo postupak usrednjavanja funkcijom $w(\tau)$

- $w(\tau)$ kazuje koliki je doprinos mjerenja $x(t+\tau)$ filtriranom izlazu $h(t)$.

Filtrirani izlaz dobivamo skalarnim produktom funkcije w s posmaknutim signalom x:

$$h(t) = (w \star x)(t) = \int_{-\infty}^{\infty} w(\tau) x(t+\tau) d\tau$$

Što je to konvolucija?



$$h(t) = (w \star x)(t) = \int_{\mathcal{D}(w)} w(\tau) x(t+\tau) d\tau$$

Što je konvolucija?

Usrednjavanje signala postigli smo unakrsnom korelacijom ulaza $x(t)$ s prikladnom funkcijom $w(t)$:

$$h(t) = w(t) \star x(t)$$

U kontekstu ”konvolucijskih” modela:

- funkcija x (argument) je ulaz,
- funkcija w (argument) je jezgra, (slobodni parametri)
- funkcija h (rezultat) se naziva mapa značajki.

Što je konvolucija?

U diskretnom slučaju umjesto integrala koristimo zbroj:

$$h(t) = (w \star x)(t) = \sum_{\tau=-\infty}^{\infty} w(\tau)x(t+\tau)$$

Pretpostavljamo da je domena ulaza i jezgre konačan skup, tj. da su funkcije x i w izvan domene jednake 0:

$$h(t) = (w \star x)(t) = \sum_{\tau=\tau_{\text{min}}}^{\tau_{\text{max}}} w(\tau)x(t+\tau)$$

U primjenama je su x i w obično višedimenzionalne funkcije, tj. $\boldsymbol{x}(t), \boldsymbol{w}(t) \in \mathbb{R}^d$.

Što je konvolucija?

Korelacija se može primjenjivati kroz više dimenzija, npr. ako je argument ulaza dvodimenzionalan, kao u slučaju slika:

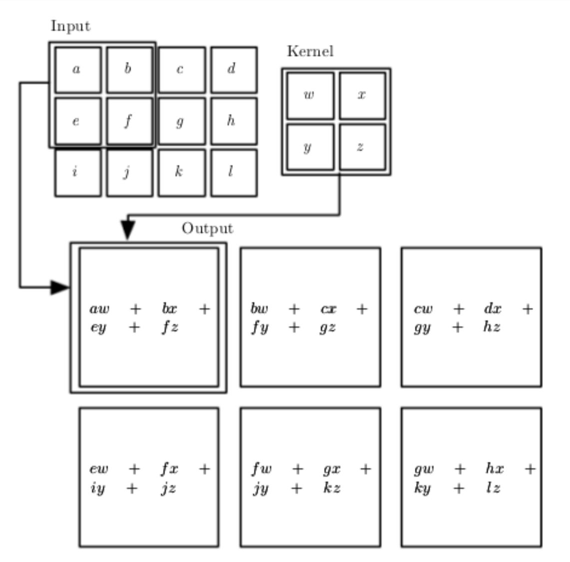
$$S(i,j) = (K \star I)(i,j) = \sum_{m=m_{\min}}^{m_{\max}} \sum_{n=n_{\min}}^{n_{\max}} K(m,n) \cdot I(i+m, j+n)$$

.

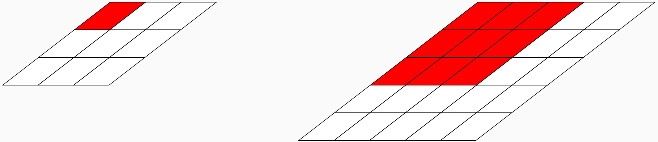
- rasponi min i max određeni vrijednostima na kojima su I i K definirani (tj. $\neq 0$).

Jezgra (3x3 - 7x7) je tipično manja od slike (MPx)

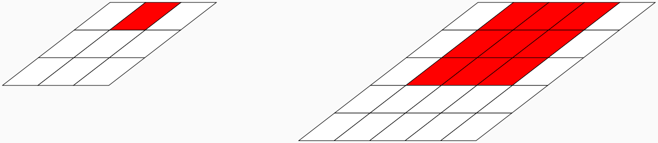
Što je to konvolucija?



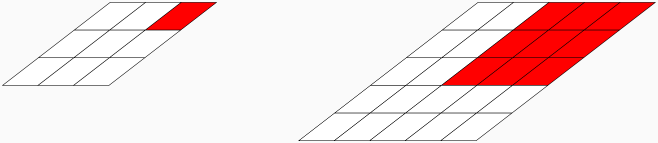
Što je to konvolucija?



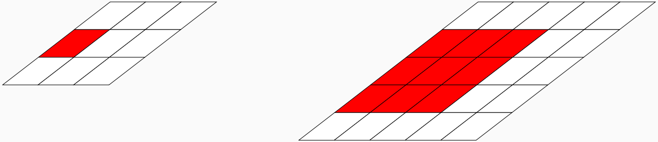
Što je to konvolucija?



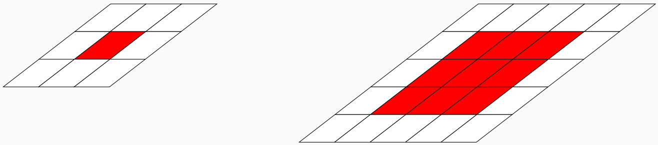
Što je to konvolucija?



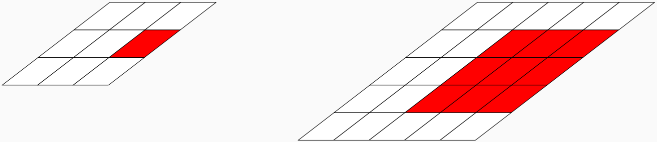
Što je to konvolucija?



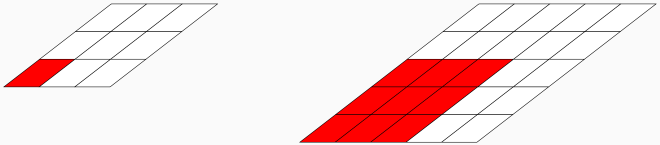
Što je to konvolucija?



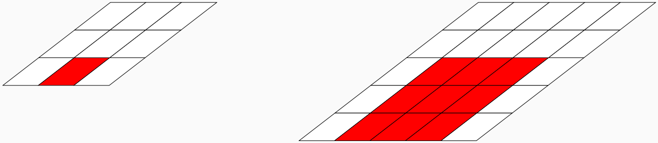
Što je to konvolucija?



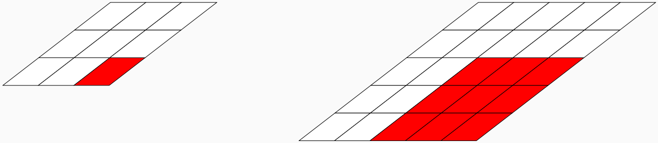
Što je to konvolucija?



Što je to konvolucija?

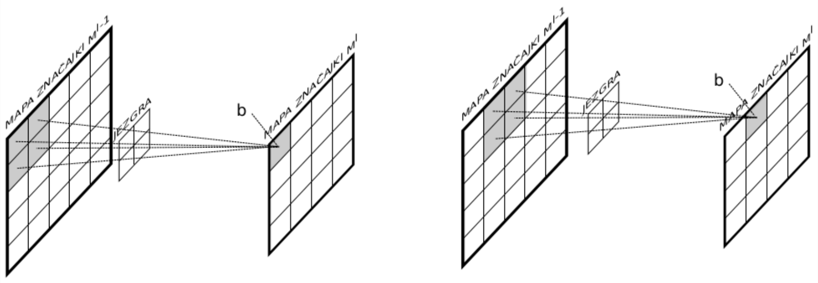


Što je to konvolucija?



Što je to konvolucija?

Konvolucija modelira lokalne interakcije i dijeli parametre:



Veza između ulaza i izlaza je linearna, ali:

- elementi izlaza M^l ovise o **lokalnom** susjedstvu ulaza M^{l-1}
- svi elementi mape značajki se računaju uz pomoć **istog** (dijeljenog) skupa parametara

Zašto konvolucija?

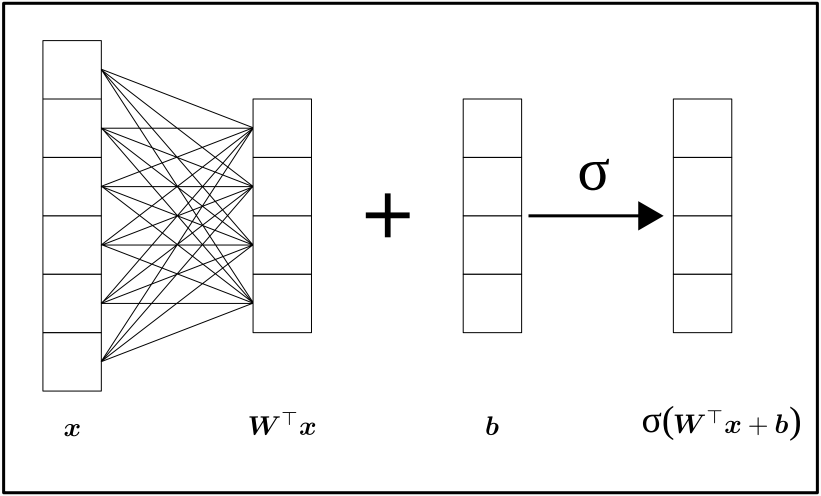
Konvolucija je slična potpuno povezanom sloju, ali postoje razlike:

- možemo modelirati samo lokalne interakcije.
- dijeljenje parametara $\rightarrow$ izlazna reprezentacija je **ekvivarijantna** s obzirom na pomak.

Svaka konvolucija može se predstaviti odgovarajućom afinom transformacijom:

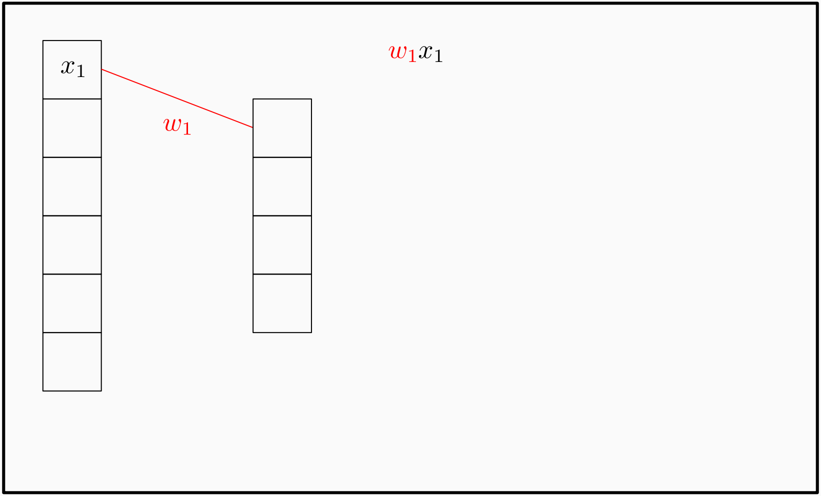
- konvolucijski sloj je regularizirana specijalizacija potpuno povezanog sloja.

Potpuno povezani sloj

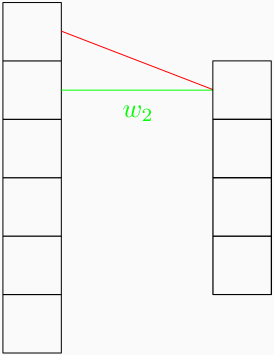


$$f(x, \boldsymbol{\Theta} = (W, b)) = Wx + b$$

Konvolucijski sloj (linearni dio)

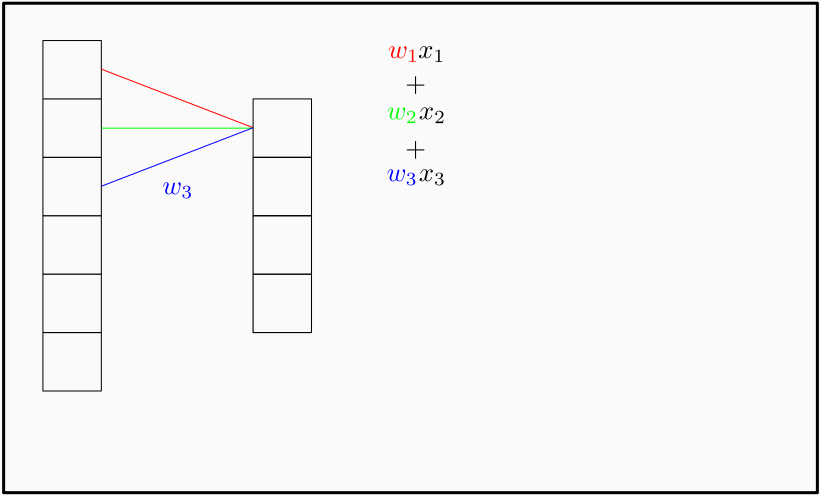


Konvolucijski sloj (linearni dio)

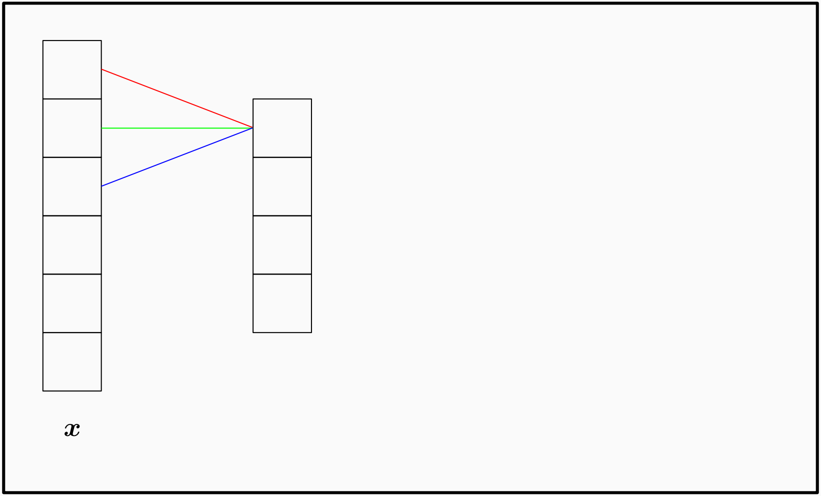


$$w_1 x_1 + w_2 x_2$$

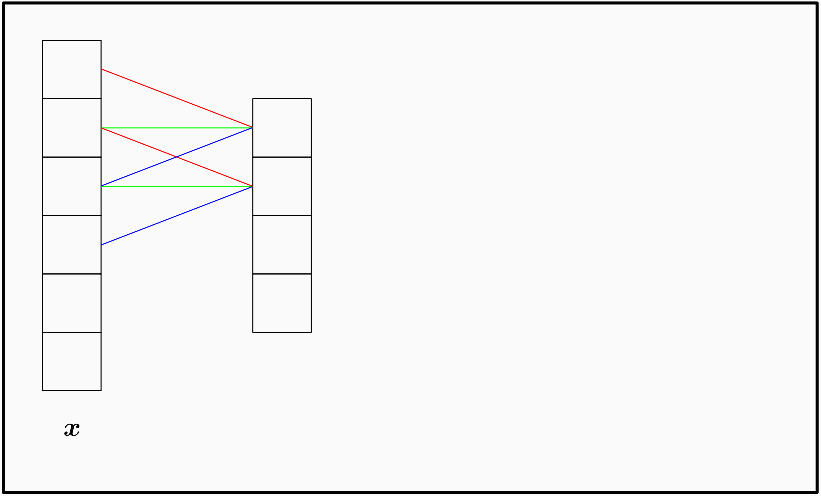
Konvolucijski sloj (linearni dio)



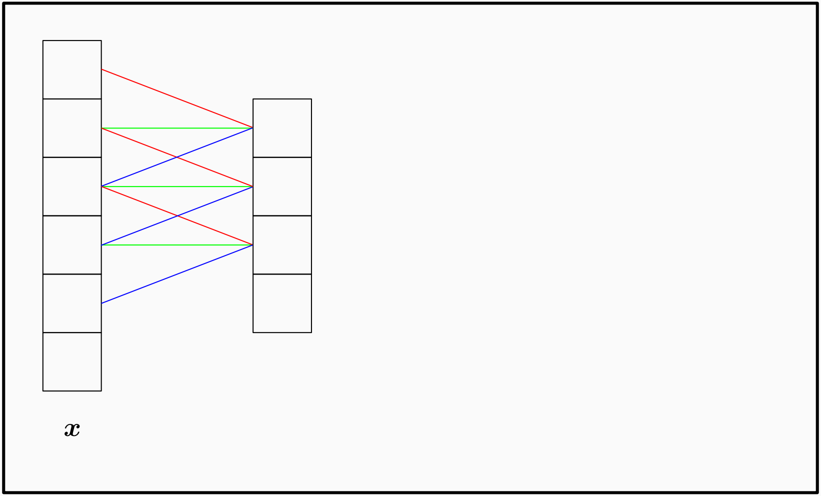
Konvolucijski sloj (linearni dio)



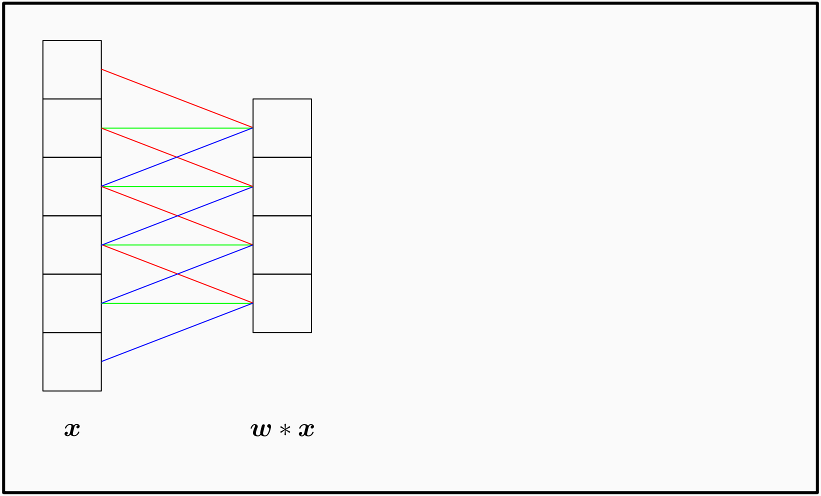
Konvolucijski sloj (linearni dio)



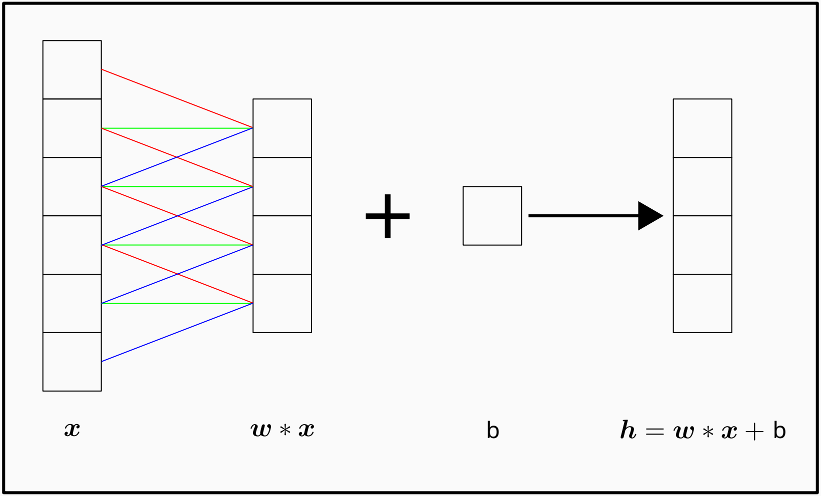
Konvolucijski sloj (linearni dio)



Konvolucijski sloj (linearni dio)



Konvolucijski sloj (linearni dio)

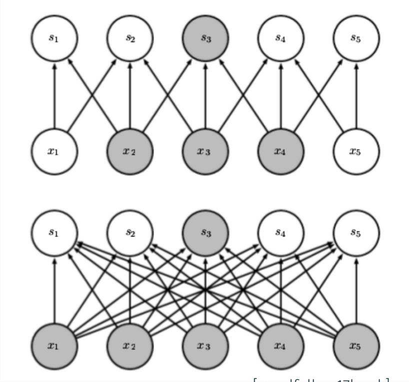


$$f(x, \boldsymbol{\Theta} = (w, b)) = w * x + b \cdot \mathbf{1}$$

Lokalne interakcije

Usporedba konvolucije s potpuno povezanim slojem:

- konvolucijske aktivacije (gore) "vide" samo mali broj ulaza
- aktivacije potpuno povezanog sloja (dolje) "vide" sve ulaze
- konvolucijski sloj ima manje veza i manje parametara.

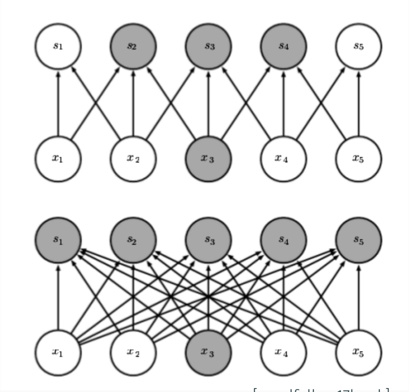


[goodfellow17book]

Lokalne interakcije (2)

Usporedba konvolucije s potpuno povezanim slojem (2):

- ulazi konvolucije (gore) utječu na samo mali broj izlaza
- ova spoznaja sugerira da ćemo širenje gradijenata unatrag također izražavati konvolucijom



[goodfellow17book]

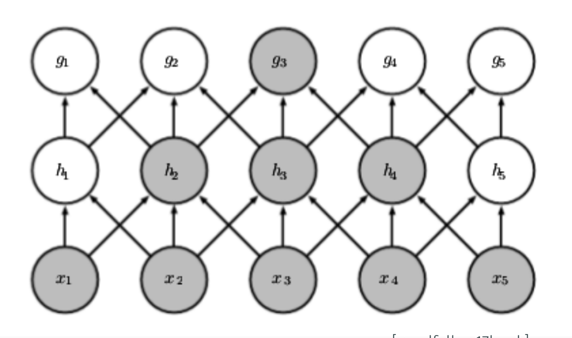
Lokalne interakcije (3)

Prednosti konvolucije pred potpuno povezanim slojem:

- brža evaluacija $O(m \cdot n)$ *vs* $O(k \cdot n)$
    - k širina jezgre
    - m dimenzija ulazne latentne reprezentacije
    - $k \ll m$.
- manji model: $k \cdot n$ *vs* $m \cdot n$ parametara
    - ovdje razmatramo samo lokalnost, a zanemarujemo dijeljenje parametara.
- manje parametara za naučiti, s istom količinom označenih podataka.

Ipak, značajke dubokog konvolucijskog modela indirektno mogu modelirati interakciju velike regije ulaznih značajki:

- receptivno polje značajke: skup svih elemenata ulaznog sloja koje mogu utjecati na tu značajku.
- receptivno polje konvolucijskih aktivacija raste s dubinom.



[goodfellow17book]

Dijeljenje parametara (ili povezivanje težina)

Sve izlazne značajke računaju se s obzirom na isti skup težina:

- skup težina za računanje značajke h_{00} isti je kao i skup težina za računanje značajke $h_{kl} \forall k, l$

Umjesto da učimo odvojen skup parametara za svaki od n izlaza, svi izlazi dijele jedan te isti skup parametara:

- više signala za učenje
- manja opasnost da će model biti prenaučen

Dijeljenje parametara (2)

Prednosti pred potpuno povezanim slojem:

- još manje parametara ($m \cdot n$ *vs* k): bolja statistička efikasnost modela.
- računska složenost evaluacije: ista kao i za model koji ima samo lokalne interakcije ali ne dijeli težine $O(nk)$.

Ekvivarijantnost na pomak

$f(x)$ je ekvivarijantna s obzirom na g ako vrijedi:

$$f(g(x)) = g(f(x))$$

Konvolucija (f) je ekvivarijantna s obzirom na pomak (g):

- izlaz konvolucije prikazuje prostornu mapu značajki ulaznog tenzora (vremenskog slijeda, matrice, volumena)
- ako pomaknemo ulaz, pomaknut će se i mapa značajki.
- konvolucijski modeli su prikladni za slike, govor, jezik, bioinformatiku, ...

Konvolucija nije ekvivarijantna s obzirom na neke druge transformacije ulaza, npr. skaliranje ili rotaciju.

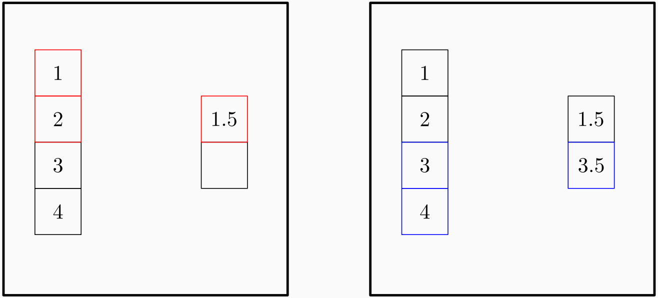
Sloj sažimanja

Funkcija sažimanja (*eng. pooling function*) mapira skup prostorno bliskih značajki na ulazu u jednu značajku na izlazu.

Obično se računa statistički pokazatelj ulaznih značajki, npr. srednja ili maksimalna vrijednost.

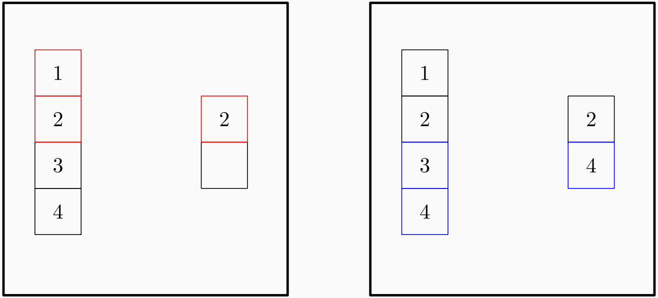
Sažimanje maksimalnom i srednjom vrijednošću

Sažimanje srednjom vrijednošću



Sažimanje maksimalnom i srednjom vrijednošću

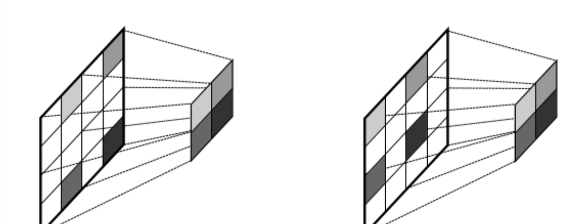
Sažimanje maksimalnom vrijednošću



Sloj sažimanja: motivacija

Povećanje invarijantnosti na pomak

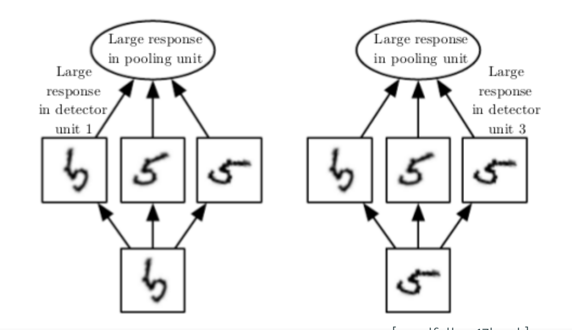
- $f(x)$ je invarijantna s obzirom na g ako: $f(g(x)) = f(x)$
- posebno korisno ako je za raspoznavanje važnje detektirati prisutnost koncepta nego lokaciju
  - npr. kod detekcije lica: pomak očiju u odnosu na nos varira od osobe do osobe
- veličina regije sažimanja regulira dozu invarijantnosti: veća regija $\rightarrow$ invarijantnost na veće pomake
  - npr. kod kategorizacije slika: objekt koji definira razred može biti bilo gdje u slici



Sloj sažimanja: motivacija (2)

Sažimanje se može provesti ne samo preko susjednih značajki iste mape nego i preko različitih mapa.

Tada model može naučiti invarijantnost i na druge transformacije.

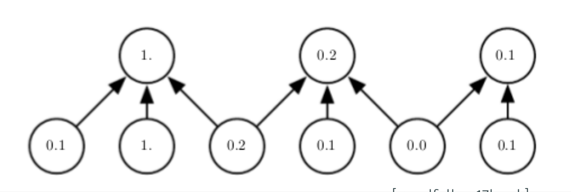


[goodfellow17book]

Sloj sažimanja: lokalna primjena

Sažimanje tipično provodimo na nepreklapajućim oknima k×k

- mapa značajki se podijeli na regije
- svaka regija se sažima u jednu značajku nepromijenjene semantičke dimenzionalnosti
- izlazni korak k: mapa značajki se smanjuje k puta
    - najčešće: $k = 2$, izlaz (H/2,W/2)
    - ponekad ciljamo izlaz q×q: (k_h, k_w)=$(H/q, W/q)$
    - moderne modele tipično zaključujemo **globalnim sažimanjem**: (k_h, k_w)=(H, W), izlaz 1×1.

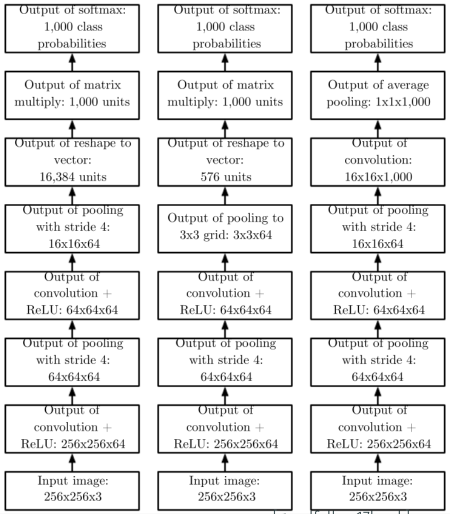


[goodfellow17book]

Sloj sažimanja: primjena preko rešetke

Ponekad veličina regije sažimanja nije fiksna

- time omogućavamo procesiranje ulaza različitih veličina

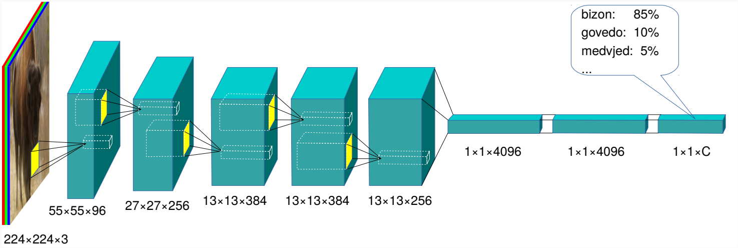


[goodfellow17book]

Sloj sažimanja: globalna primjena

Globalno sažimanje koristimo kada latentnu mapu značajki prevodimo u simboličku kategoriju:

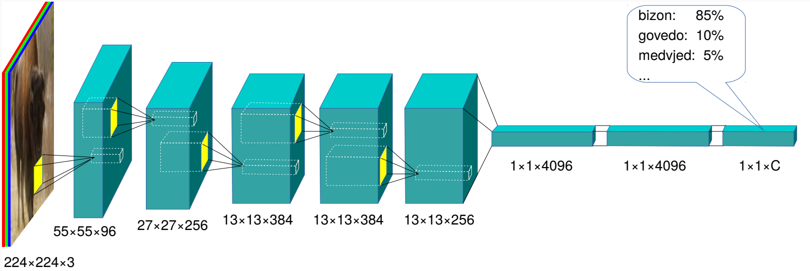
- tijesto za pizzu trebamo premijesiti u tijesto za francuz
- javlja se u najdesnijem stupcu prethodne stranice



Sloj sažimanja: pitanja

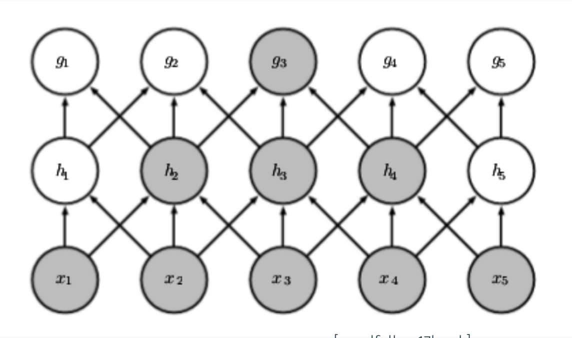
Što se zbiva s receptivnim poljem značajki dobivenih sažimanjem (pretp. izlazni korak k)?

kako sažimanje utječe na broj parametera modela?



Učinak konvolucije s jezgrom veličine k:

- uvećanje receptivnog polja za k-1 (ako nije bilo sažimanja)



[goodfellow17book]

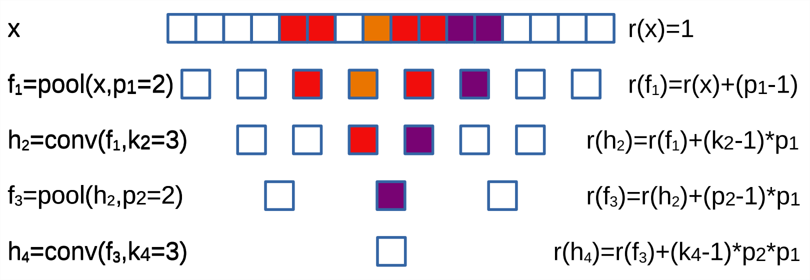
Učinak poduzorkovanja s jezgrom veličine k i korakom k:

- uvećanje svog receptivnog polja za k-1
- umnažanje receptivnog doprinosa svih sljedbenika k puta!

Receptivno polje (2)

Veličinu receptivnog polja možemo odrediti analiziranjem modela unaprijed sloj po sloj:

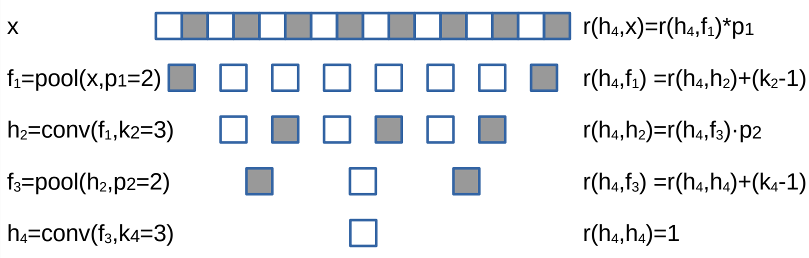
- boje označavaju ukupno receptivno polje odgovarajućih aktivacija
- moramo pamtiti ukupni faktor poduzorkovanja
- tim faktorom množimo receptivni doprinos sloja: k - 1



Receptivno polje (3)

Veličinu receptivnog polja možemo odrediti i analizom unatrag (pretpostavljamo da model raste prema prednjoj strani):

- sive aktivacije označavaju uvećanje receptivnog polja u odnosu na sljedeći sloj
- prednost: ne moramo pamtiti ukupni faktor poduzorkovanja (teže je pogriješiti)



Konvolucija i sažimanje: pristranost

Postavljanjem konvolucija i sažimanja u model unosimo sljedeće oblike pristranosti:

- konvolucija: interakcije su lokalne s obzirom na topologiju $\rightarrow$ pretpostavljamo topologiju podataka
- konvolucija: predikcija je ekvivarijantna s pomakom
- sažimanje: predikcija je invarijantna na male pomake

Te pretpostavke povećavaju pristranost i smanjuju varijancu

- teorija: to može dovesti do podnaučenosti i bolje generalizacije
- praksa: nema podnaučenosti, konvolucijski modeli bolje generaliziraju od potpuno povezanih

Konvolucija i sažimanje: pristranost (2)

Konvolucija je kao potpuno povezani sloj u kojem smo težine izvan područja jezgre dekretom postavili na 0:

- gubitak dobivenog modela biti će veći od gubitka odgovarajućeg modela sa slobodnim težinama

Ako konvolucijski sloj dovodi do podnaučenosti (loših rezultata na skupu za učenje):

- možda nisu samo lokalne interakcije važne $\rightarrow$ povećati veličinu receptivnog polja

Ako sloj sažimanja dovodi do podnaučenosti:

- zadatak ovisi o preciznim lokacijama značajki $\rightarrow$ smanjiti veličinu regije sažimanja

Nadopunjavanje (*eng. padding*)

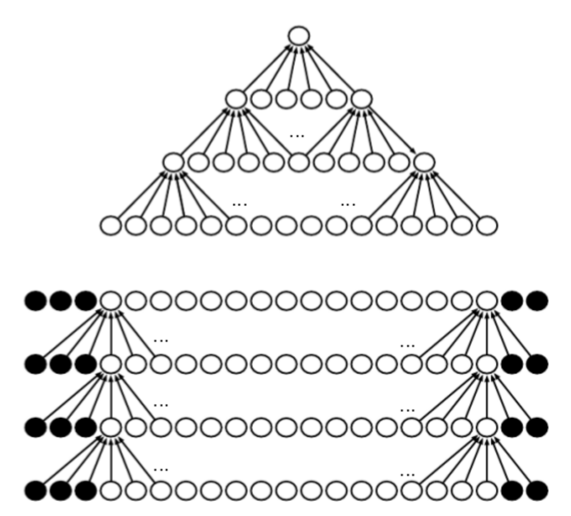
Bez nadopune: reprezentacija se smanjuje s dubinom

- za ulaz veličine m izlaz je $m - k + 1$, gdje je k veličina jezgre
- nedostatak: elementi na rubovima manje utječu na izlaz od elemenata u sredini.
- dubina ograničena veličinom ulaza i konvolucijske jezgre
- u programskim bibliotekama ovakvu konvoluciju označavamo s `"VALID"`.

Nadopuna nulama na rubovima: neograničena dubina mreže

- ulaz iste veličine kao i izlaz pod uvjetom da se doda $k - 1$ nula na rubove
- primjer: za $k = 5$ dodamo po dvije 0 sa sve četiri strane (lijevo, desno, gore, dolje)
- ovakve konvolucije označavamo s `"SAME"`

Nadopunjavanje (primjer)



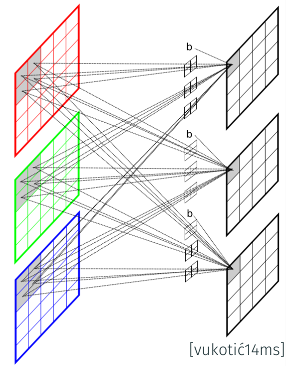
Višekanalna 2d konvolucija

Proširujemo definiciju operatora $\star$ prema slici i zadržavamo sintaksu:

$$\mathbf{q} = \mathbf{w} \star \mathbf{p}$$

Izlaz $\mathbf{q}^{(g)}$ zbraja konvolucije odgovarajućih kriški ulaza i g-te jezgre:

$$\mathbf{q}^{(g)} = \sum_f \mathbf{w}^{(g,f)} \star \mathbf{p}^{(f)}$$



[vukotić14ms]

Ista operacija može se izraziti i kroz umnoške skalara:

$$q_{ij}^{(g)} = \sum_{fuv} p^{(f)}_{i-o_k+u,j-o_k+v} \cdot w_{uv}^{(g,f)}$$

Konvolucije često imaju i pomak: $\mathbf{q} = \mathbf{p} \star \mathbf{w} + \mathrm{broadcast}(\mathbf{b})$

- komponente pomaka odgovaraju kanalima izlaza: $\mathbf{b} = [b_g]$

Konvolucija: zadatak

Razmatramo klasifikacijski konvolucijski model za sive slike dimenzija 28x28. Arhitektura modela je:

- dva konvolucijska sloja bez nadopunjavanja: jezgra 5×5 s pomakom; aktivacija ReLU; sažimanje maksimumom 2×2; korak 2;
  - prvi sloj: 16 kanala, drugi sloj: 32 kanala;
- potpuno povezani sloj dimenzije 512 s pomakom + ReLU;
- potpuno povezani sloj D=10 s pomakom + softmaks.

Zadatci:

1. Odredite dimenzije aktivacija, broj parametara te veličinu receptivnog polja u svim slojevima.
2. Natipkajte implementaciju u PyTorchu s metodama `__init__`, `fwd`, i `loss`.

Konvolucija: zadatak 2

Natipkajte svoju implementaciju 1D konvolucije pod Numpyjem (jedna for-petlja, samo unaprijedni prolaz).

```
import numpy as np

def conv1d_my(vector, kernel):
  n = vector.shape[0]
  k = kernel.shape[0]
  out = np.zeros((n-k+1,), dtype=np.float32)
  kernel = np.flip(kernel)
  for i in range(n-k+1):
    out[i] = np.sum(vector[i:i+k] * kernel)
  return out
```

Konvolucija: zadatak 3

Usporedite svoju implementaciju konvolucije s odgovarajućm implementacijama iz torcha i scipyja.

```
import torch
from torch.nn.functional import conv1d as conv1d_torch
from scipy.ndimage import convolve1d as conv1d_scipy

x = np.array([2.0]*3 + [6.0]*4)
w = np.array([-1.0, 1.0])
print(conv1d_my(x,w))
print(conv1d_scipy(x, w, mode='nearest'))
print(conv1d_torch(torch.tensor(x).reshape([1,1,-1]),
  torch.tensor(w).reshape([1,1,-1])).numpy().squeeze())

# [ 0.  0. -4.  0.  0.  0.]
# [ 0.  0. -4.  0.  0.  0.  0.]
# [ 0.  0.  4.  0.  0.  0.]
```