

UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO
ODDELEK ZA FIZIKO

PRAKTIKUM STROJNEGA UČENJA V FIZIKI

3. naloga: Določanje elastičnih konstant v nematskem tekočem kristalu

Žiga Šinigoj, 28222025

Ljubljana, november 2023

1 Uvod

V nalogi se bomo ukvarjali z določanjem elastičnih konstant tekočega kristala z metodami strojnega učenja. Ob izbrani orientaciji tekočega kristala in postavitvi eksperimenta ter koordinatnega sistema v celici ima Frankova elastična energija dve relevantni elastični konstanti K_{11} in K_{33} , ki ustrezata pahljačasti in upogibni deformaciji. Cilj naloge je napovedati konstanti iz podanih intenzitet $I(t)$, ki jih tekoči kristal prepusti tekom relaksacije v ravnovesno stanje. Tekoči kristal je dvolomen, kar nam podaja različno prepustnost svetlobe glede na različne orientacije molekul.

Najprej bomo poiskali napovedi za eno konstantni približek $K_{11} = K_{33}$. Vhodni podatki so diskretizirane intenzitete v 400 točkah na intervalu $[0 \text{ s}, 1,2 \text{ s}]$. Vsaki intenziteti pripada vrednost elastičnih/e konstant/e. Podatke razdelimo na tri sete, set za treniranje, validacijo in test. Na testnem setu nato napovemo elastične konstante. V zaključku naloge je potrebno napovedati elastične konstante iz eksperimentalnih intenzitet, ki bodo v tem primeru naša testna množica. Pri učenju enega modela brez navzkrižne validacije je potrebno določiti velikost seta za trening in validacijo, v primeru Kfold-a pa določimo samo celotno velikost seta, ki se bo potem delil na trening in validacijo.

Za večjo preglednost v poročilu bom uvedel nekaj notacij. Vsaki nevronske mreži pripada set hiperparametrov, da dovolj dobro opišem dano mrežo bom njene parametre podal v seznamu kot:

- $[(400,200,100,1), \text{relu}, \text{adam}, \text{MSLE}, \text{MSE}, \text{LR}(10^{-5}), \text{E150}, \text{BS20}]$

kjer prvi oklepaj označuje arhitekturo nevronske mreže (1 plast 400 nevronov, 2 plast 200, ...), aktivacijsko funkcijo - relu, ki je načeloma enaka za vse plasti, razen za zadnjo, adam - optimizator, MSLE- loss funkcijo, metriko-MSE, hitrost učenja - LR(vrednost), število epoh-E(vrednost), velikost batch-ev - BS(vrednost).

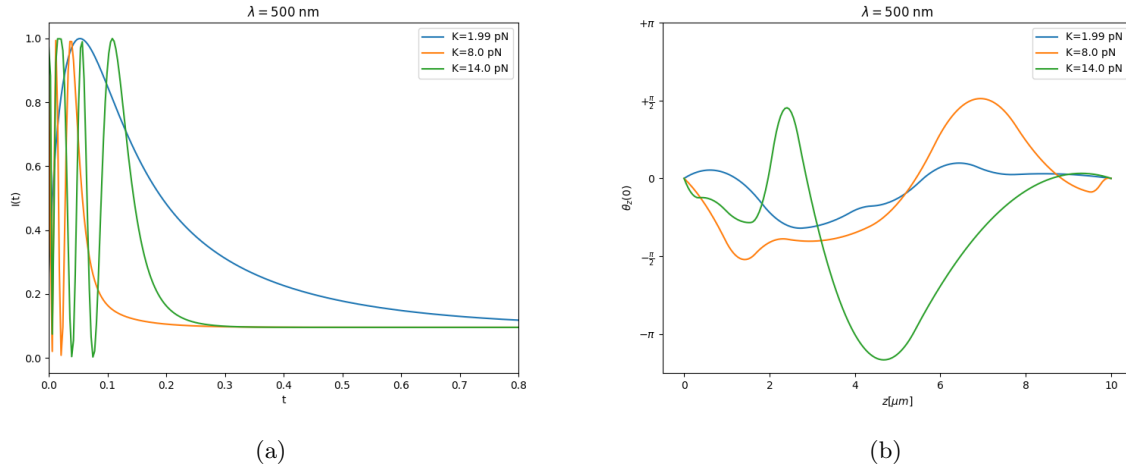
Pri učenju modelov sem uporabil EarlyStopping, ki poskrbi, da se model ustavi učiti, ko dosežeta validacijska in funkcija izgube za trening minimum. Poleg tega sem uporabil še funkcijo *ReduceLROnPlateau*, ki upočasni hitrost učenja modela, ko se metrika ne izboljšuje več.

Napovedi elastičnih konstant bom prikazoval v 2D histogramih, kjer barva označuje število napovedi elastične konstante v danem območju (N).

2 Približek ene konstante

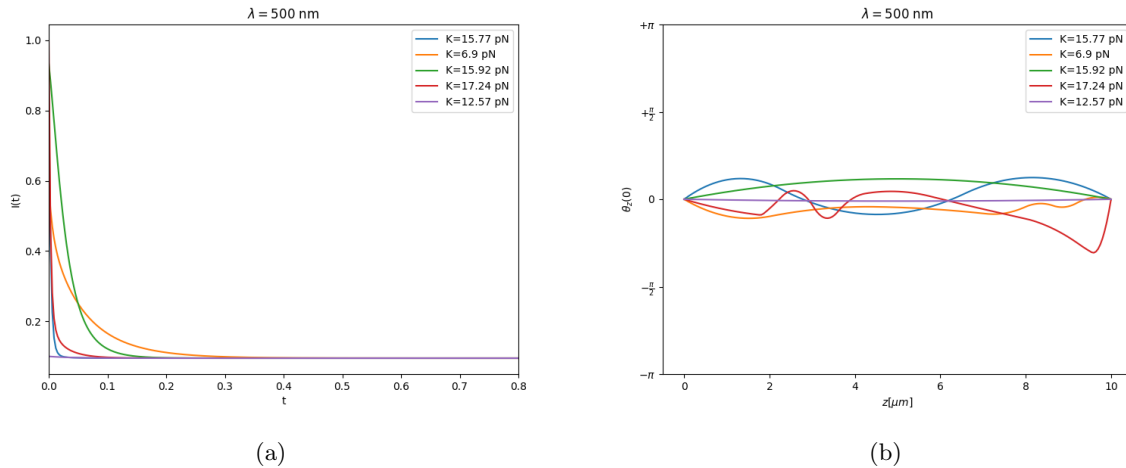
2.1 Podatkovni set

Simuliran podatkovni set intenzitet uporabimo za učenje modela. Pri eno-konstantnem približku sem uporabljal intenzitete valovne dolžine $\lambda = 500 \text{ nm}$. Da se spoznamo s podatkovnim setom sem najprej narisal nekaj tipičnih in ekstremnih intenzitet v množici (slika 1, 2, 3). Tipična intenziteta je taka, da vsebuje nekaj nihajev na začetku in nato relaksacijo v ravnovesje. Pri intenzitetah z več nihajih je tudi začetna odvisnost $\theta(z)$ bolj valovita. Ekstremni primeri prikazujejo dve limiti, zelo hitro relaksacijo modela v ravnovesje (slika 2) in zelo počasno relaksacijo (slika 3).



Slika 1: a) Nekaj tipičnih intenzitet v podatkovnem setu b) Pripadajoči začetni profil $\theta(z)$.

Ekstremne intenzitete ustrezajo velikim in majhnim elastičnim konstantam. Začetni kotni profil dosega pri velikih konstantah majhne vrednosti zasuka, pri majhnih pa precej večje. Ker pri majhnih K ne dosežemo relaksacije v danem časovnem intervalu bo mogoče nevronska mreža težje določila elastično konstanto. V primeru velikih elastičnih konstant bodo za nevronske mreže zelo pomembne začetne točke oz. odvisnosti, saj vsebujejo večino informacije.

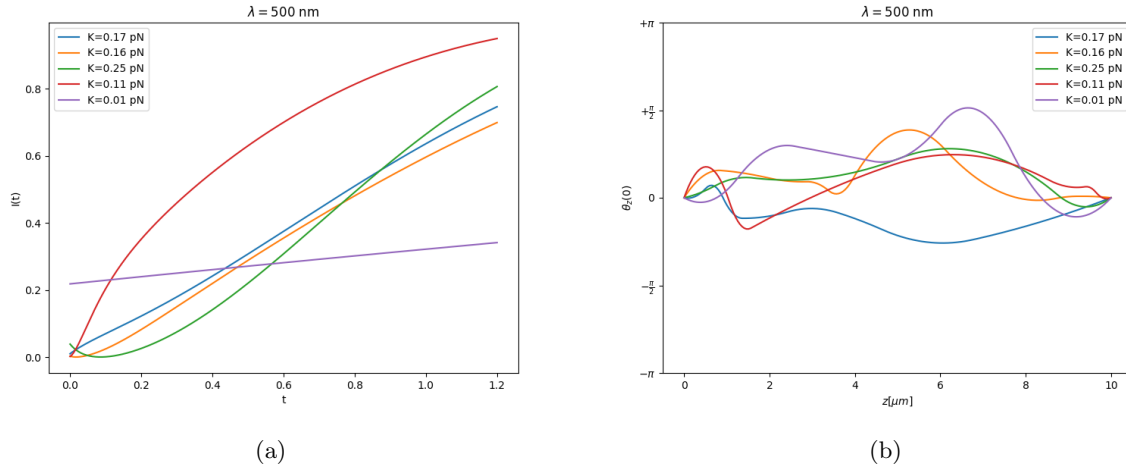


Slika 2: a) Intenzitete, ki ustrezajo hitri relaksaciji kristala. b) Pripadajoči začetni profil $\theta(z)$.

2.2 Odvisnosti modela od hiperparametrov

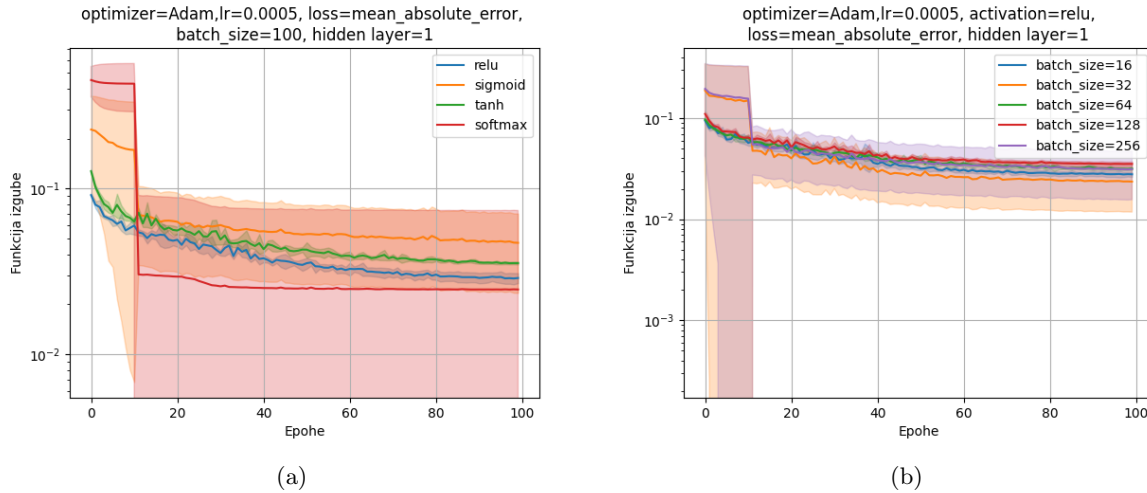
Pri konstrukciji nevronske mreže sem uporabil knjižnico Tensorflow in Scikit-learn. Uspešnost nevronske mreže določa pravilno izbran set hiperparametrov, zato sem najprej pogledal, kako različni hiperparametri minimizirajo funkcijo izgube. Ker gre samo za grobo oceno sem modele učil na 50% podatkov z navzkrižno validacijo. Na koncu sem povprečil po rezultatih navzkrižne validacije in izračunal standardni odklon. Iz tega sem nato določil *najboljši* model. Težava takega določanja hiperparametrov je ta, da jih obravnavamo kot neodvisne, kar pa načeloma ne drži. Za boljše določevanje bi bilo potrebno pogledati vse možne kombinacije hiperparametrov ("grid search"), kar pa je časovno zelo potratno.

Najprej sem pogledal funkcijo izgube pri različnih aktivacijskih funkcijah (slika 4a). V tem primeru ReLu najbolj minimizira aktivacijsko funkcijo. Kot optimalna velikost batch-ev (slika 4b) sem dobil $BS = 32$, ampak ima ta precej večjo nedoločenoost kot $BS = 16$, zato sem izbral kot optimalen $BS = 16$. Manjša velikost batch-ev pomeni propagacijo manjše množice skozi nevronske mreže naenkrat, kar poveča čas učenja modela, ampak načeloma tudi izboljša model.

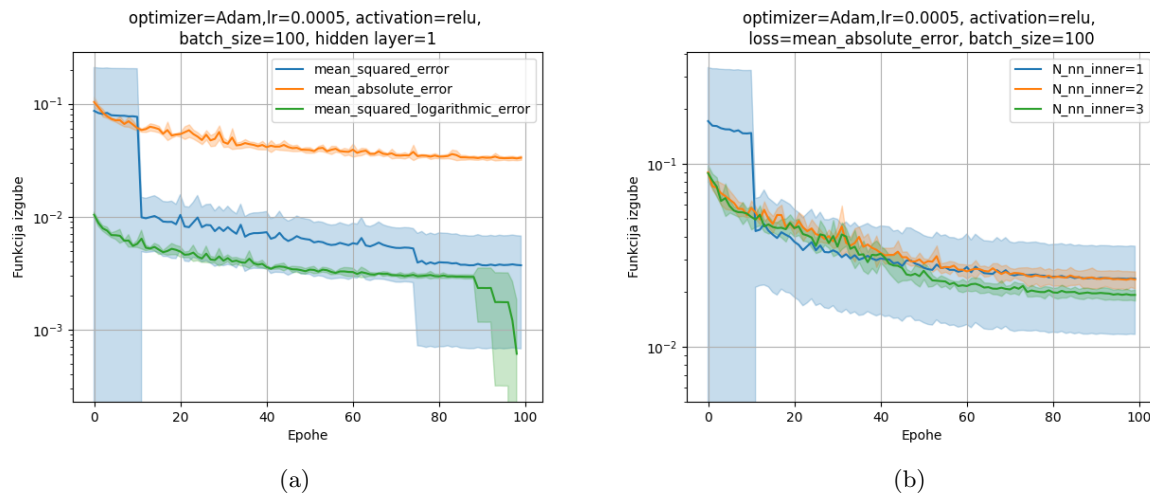


Slika 3: a) Intenzitete, ki ustrezajo počasni rekalsaciji kristala v ravnovesno stanje. b) Pripadajoči začetni profil $\theta(z)$.

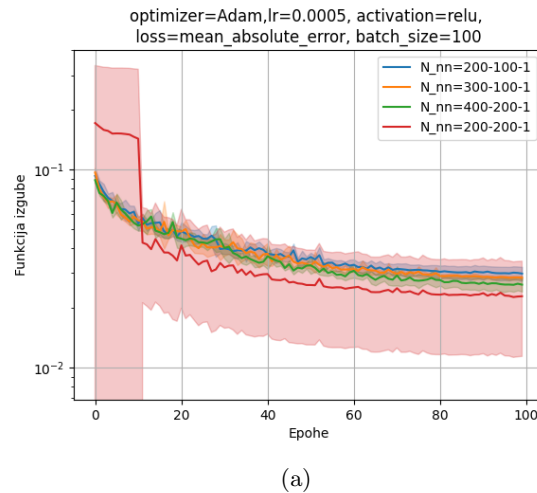
Pomembni parametri nevronske mreže so tudi funkcija izgube, število notranjih plasti in velikost posamezne plasti. Njihovo odvisnost od funkcije izgube prikazujejo slike 5a,b, 6. Za najboljšo funkcijo izgube se izkaže *MSLE* - mean squared logarithmic error. Tri notranje plasti najbolj optimizirajo funkcijo izgube, ker pa število plasti najbolj poveča časovno zahtevnost od vseh hiperparametrov, bom za moj model izbral raje dve notranji plasti. Pomemben hiperparameter je tudi število nevronov v posamezni plasti. Nekaj funkcij izgub pri različnih kombinacijah sem pogledal na sliki 6. Arhitektura (200,200,1) najbolj optimizira funkcijo izgube, a ima obenem največjo nedoločenoost, zato sem raje izbral arhitekturo (400,200,1).



Slika 4: (200,100,1) a) Povprečna funkcija izgube v odvisnosti od Epoch za različne aktivacijske funkcije, za aktivacijsko funkcijo zadnje plasti sem vedno uporabil ReLu. b) Povprečna funkcija izgube v odvisnosti od Epoch za različne velikosti batch-ev. Obarvano območje označuje standardni odklon K-fold modela.



Slika 5: (200,100,1) a) Povprečna funkcija izgube v odvisnosti od Epoch za različne funkcije izgube. b) Povprečna funkcija izgube v odvisnosti od Epoch za različno število notranjih plasti velikosti 100. Obarvano območje označuje standardni odklon K-fold modela.



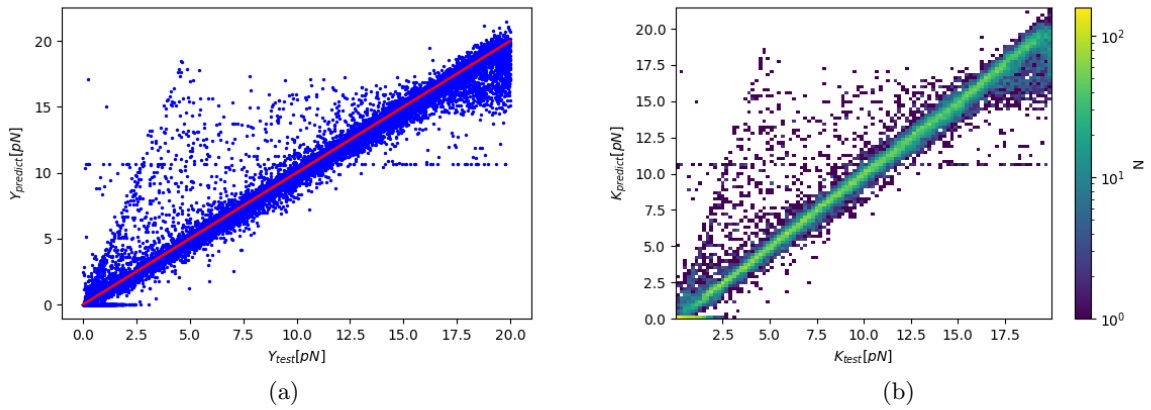
Slika 6: a) Povprečna funkcija izgube v odvisnosti od Epoch za različne arhitekture nevronske mreže. Obarvano območje označuje standardni odklon K-fold modela.

Da se iskanje hiperparametrov res izplača, bom preveril s primerjavo napovedi elastične konstante na preprosti mreži, brez navzkrižne validacije (slika 7). Opazim lahko, da večina meritev leži na premici z naklonom 1, vendar je varianca okrog premice precej velika.

Zaenkrat imamo naslednje hiperparametre:

- funkcija izgube: MSLE
- metrika: MSE
- aktivacijska funkcija: ReLU, na vseh plasteh
- velikost batch-ev: 16
- število notranjih plasti: 2
- arhitektura: (200,200,1)
- optimizator: Adam
- hitrost učenja: ?

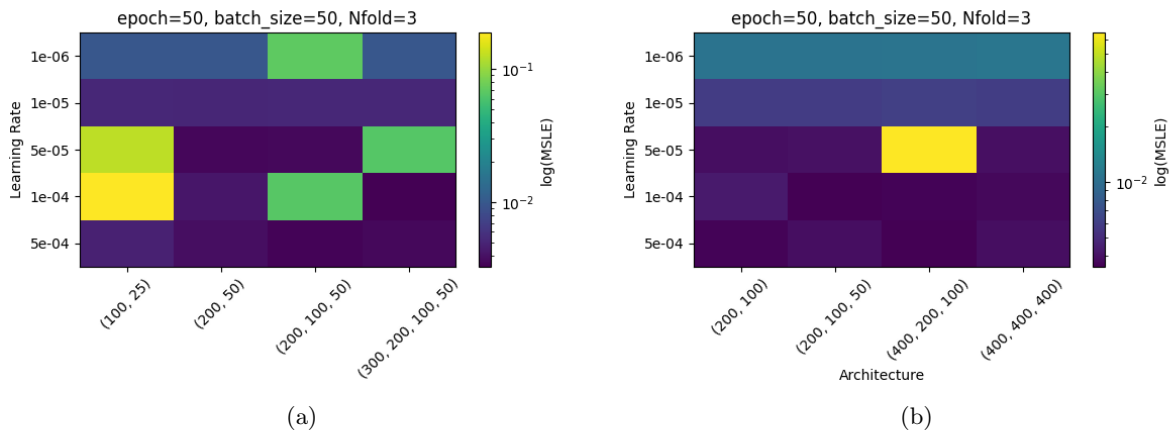
Potrebno je določiti še hitrost učenja optimizatorja Adam in arhitekturo, če dodamo še eno notranjo plast.



Slika 7: [(200,100,1), relu, adam, MSLE, MSE, $\text{LR}(10^{-5})$, E150, BS20, KF0] a) Napovedana elastična konstanta, pravilna napoved leži na rdeči premici ($Y=K$). b) Histogram napovedi elastičnih konstant, kjer barva prikazuje število napovedi v nekem območju na grafu.

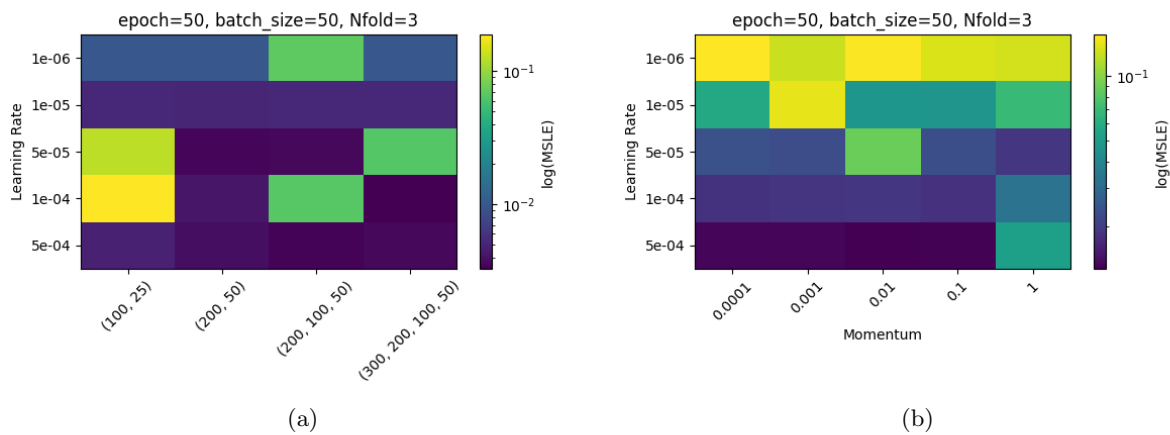
2.3 Mrežno iskanje hiperparametrov

Z metodo *GridSearchCV* poiščemo optimalno hitrost učenja in arhitekturo modela. Vrednosti funkcije izgube pri različnih parametrih prikazuje slika 8. Premajhna hitrost da slabe rezultate. Arhitektura (200,100,1) dobro minimizira funkcijo izgube, a ker sem se odločil za 2 notranji plasti, izbiram med (200,100,50,1) in (400,200,100,1). Odločil sem se za arhitekturo (400,200,100,1), saj je arhitektura (400,200,1) bolj minimizirala funkcijo izgube kot (200,100,1) (slika 6).



Slika 8: [relu, adam, MSLE, MSE] a) Mrežno iskanje arhitekture in hitrosti učenja mreže. b) Mrežno iskanje arhitekture in hitrosti učenja mreže.

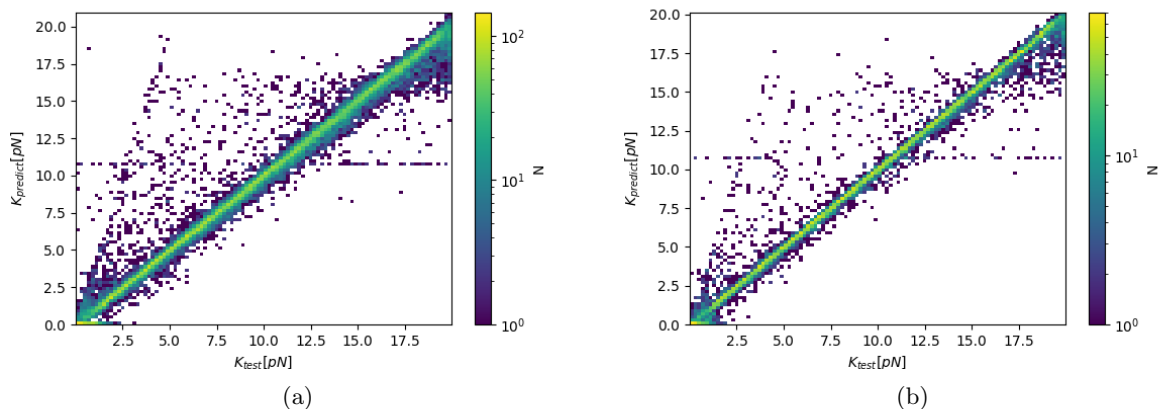
Pogledal sem tudi ali se lahko optimizator *SGD*-stochastic gradient descent približa optimizatorju *Adam*. *SGD* optimizator ima dva pomembna parametra: hitrost učenja in začetni moment, ki omogoči hitrejši spust v minimum aktivacijske funkcije. Na sliki 9a je prikazana odvisnost od arhitekture in hitrosti učenja pri začetnem momentu 0. Vidimo, da *SDG* približno enako optimizira funkcijo izgube kot optimizator *Adam*. V primeru ko dodam še začetni moment (slika 9b) pa rezultati v tem primeru niso boljši. Na rezultate precej vpliva začetni moment, vendar bi v tem primeru moral pogledati za manjše začetne momente. Zdi se, da gre optimalni parameter momenta proti 0, ko se hitrost učenja večja, kar se zdi smiselno.



Slika 9: [relu, SGD, MSLE, MSE] a) Mrežno iskanje arhitekture in hitrosti učenja mreže. b) Mrežno iskanje momenta SGD optimizatorja in hitrosti učenja mreže.

2.4 Napoved elastične konstante

Sedaj ko imamo določene hiperparametre lahko napovemo elastično konstanto. Napoved sem naredil najprej na eni množici podatkov, kjer sem vzel 70 % podatkov za trening, 20 % za validacijo in 10 % za test (histogrami 10a, 11a, 12a). Drugo napoved sem naredil po povprečenju napovedi navzkrižne validacije, kjer sem uporabil 10 % podatkov za test (histogrami 10b, 11b, 12b). Navzkrižna validacija zelo izboljša napoved elastične konstante. Napoved je večinoma natančna, nekoliko slabše pa so napovedi na robih. Ravno te ustrezajo ekstremnim primerom, ki sem jih predstavil na začetku naloge.



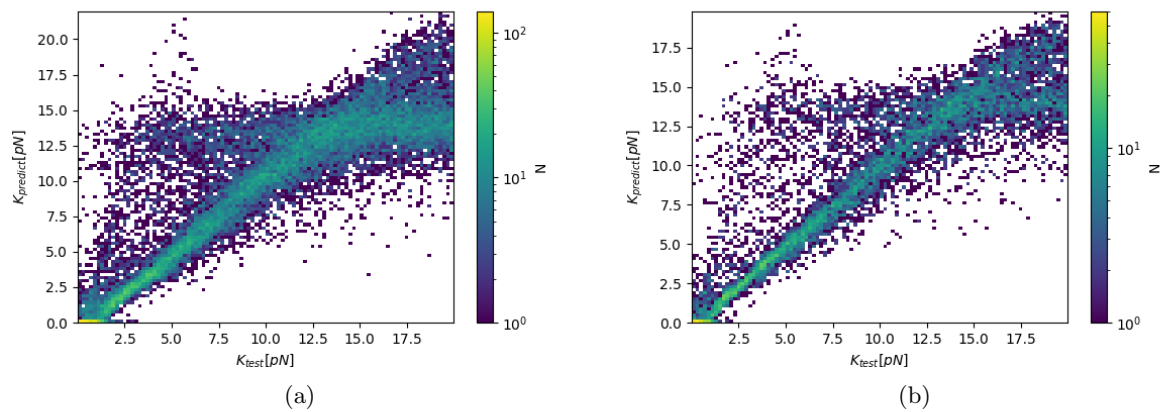
Slika 10: [(400,200,100,1), relu, adam, MSLE, MSE, $\text{LR}(10^{-5})$, E150, BS20] a) Histogram napovedi elastične konstante. b) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji, KF5.

2.5 Vpliv šuma, zamika in pomanjkanja podatkov na napoved konstante

Zanimivo je tudi pogledati vpliv šuma, zanima intenzitet in redčenja podatkov v intenzitetnem profilu na napoved elastične konstante. Pri zašumljenih podatkih (slika 11) pride do največjega odstopanja napovedi pri velikih elastičnih konstantah, kar je smiselno, saj imajo te hitro časovno odvisnost intenzitete in jim šum precej spremeni odvisnost, kar se odraža v večji spremembi konstante.

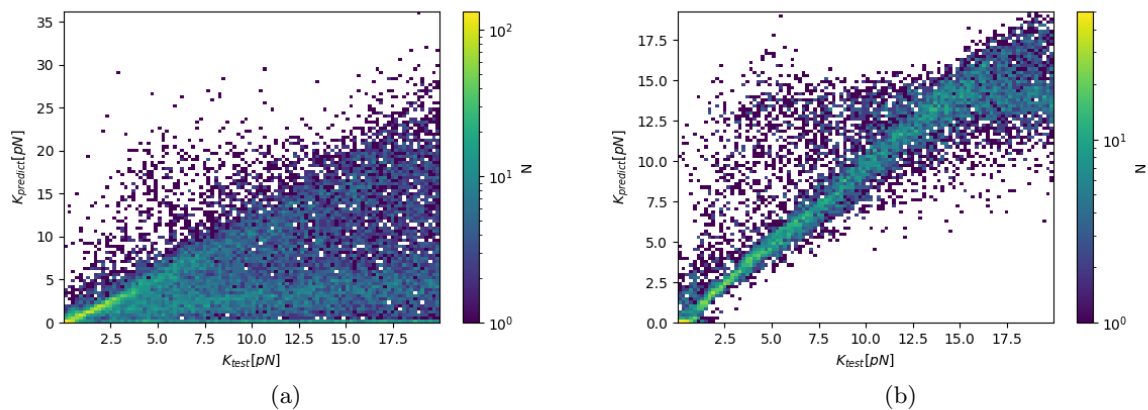
Z naučenim modelom na nezašumljenih podatkih lahko poskusimo napovedati elastične konstante na zašumljenih intenzitetah (12). Rezultat brez navzkrižne validacije je pričakovano slab, saj nevronska mreža ni nikoli videla zašumljenih podatkov. Če se nisem zmotil pri označevanju histogramov, je napoved zašumljenih podatkov z navzkrižno validacijo presenetljivo dobra.

Napovedi elastičnih konstant ob redukciji podatkov iz intenzitetnega profila in zamikom profila na interval [0.1 s, 1.2 s] prikazuje slika 13 in 14. V obeh primerih pričakujemo, da bo napoved slaba za večje konstante,

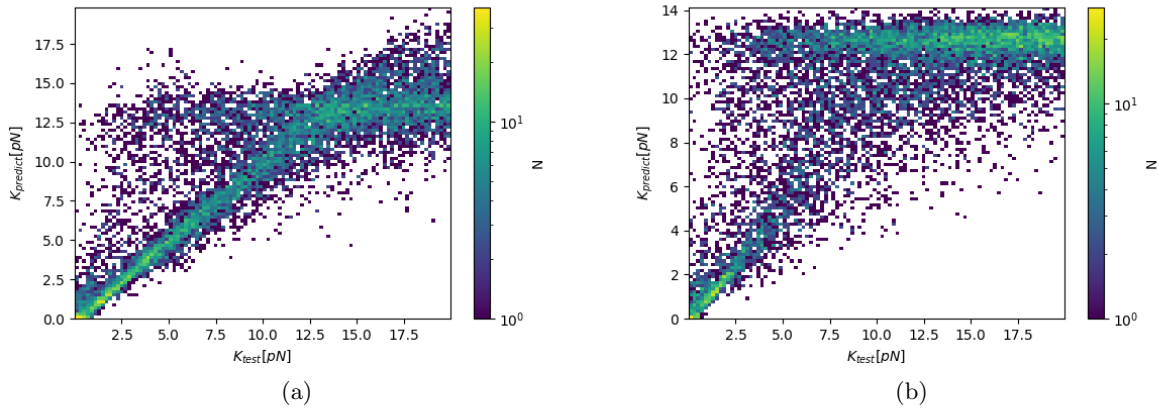


Slika 11: [(400,200,100,1), relu, adam, MSLE, MSE, $\text{LR}(10^{-5})$, E150, BS20] a) Histogram napovedi elastične konstante na zašumljenih vhodnih podatkih. b) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji na zašumljenih vhodnih podatkih, KF5.

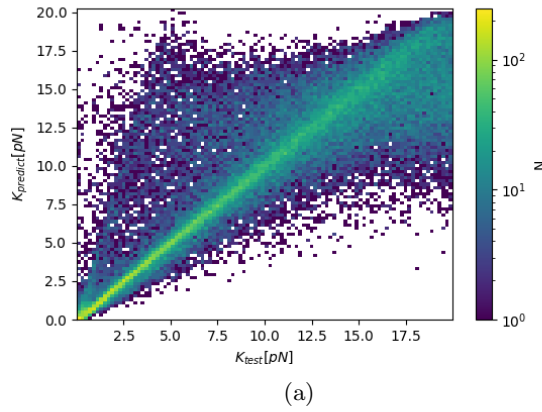
saj je tam vsa odvisnost skrita v začetnih časih in na zelo ozkem časovnem območju.



Slika 12: [(400,200,100,1), relu, adam, MSLE, MSE, $\text{LR}(10^{-5})$, E150, BS20] a) Histogram napovedi elastične konstante pri čemer model napoveduje zašumljene podatke. b) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji, pri čemer model napoveduje zašumljene podatke, KF5.



Slika 13: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji, pri čemer so vhodne intenzitete 4 krat redkejš, KF5. b) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji, pri čemer so vhodne intenzitete 40 krat redkejš, KF5.

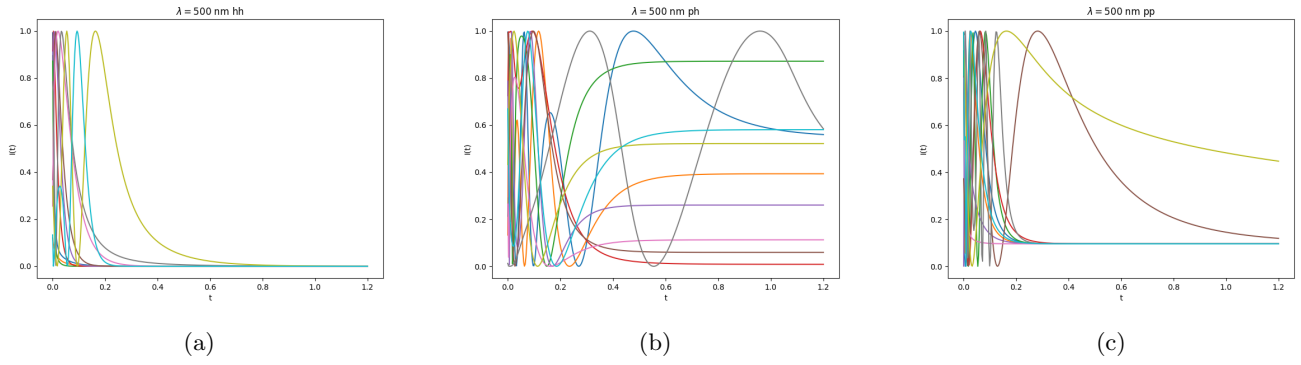


Slika 14: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Histogram napovedi elastične konstante s povprečenjem po K-fold navzkrižni validaciji, pri čemer so vhodne intenzitete zamaknjene - I([0,1,1.2]), KF5.

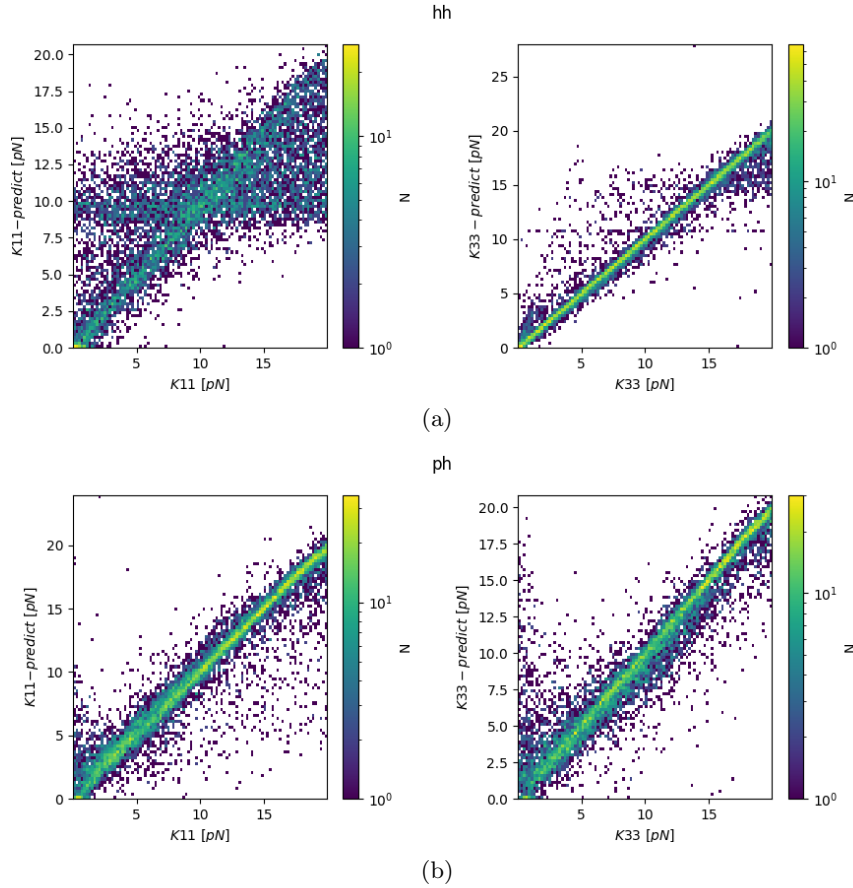
3 Dve elastični konstanti

V drugem delu obravnavamo primer, ko $K_{11} \neq K_{33}$. V ta namen smo imeli nov set podatkov, v katerem smo k vsaki simulirani intenziteti dobili tudi pripadajoč vektor dveh konstant. Dani so bili podatki za primere treh sidranj na steno celice - hometropno-homeotropno (hh), planarno-homeotropno (ph) in planarno-planarno (pp). Najprej si oglejmo nekaj tipičnih intenzitetnih profilov za vse tri primere sidranja (slika 15). Sidranje (hh) in (pp) imata precej napovedljivo intenziteto, s to razliko, da pri sidranju (pp) intenziteta ne gre proti 0. Pri mešanju sidranju (ph) pa je intenzitetni profil precej nepredvidljiv.

Podobno kot v prvem delu lahko sedaj napovemo konstanti K_{11} in K_{33} za vse tri primere sidranja (slika 16, 17). Opazimo lahko, da je napoved v primeru sidranja (hh) in (pp) dobra samo za eno elastično konstanto K_{33} oz. K_{11} . V primeru (ph) sidranja pa je napoved nekoliko slabša kot boljša napoved v primeru (hh, pp), vendar je podobno dobra za obe konstanti K_{11} in K_{33} . V primeru (ph) sidranja je deformacija sestavljena iz približno enakih prispevkov upogiba in pahljačne deformacije, zato je napoved obeh konstant približno enaka. V primeru (hh) oz. (pp) sidranja pa prevladuje ena izmed deformacij in zato je ena konstanta dobro določena, druga pa ne, saj malo prispeva k deformaciji oz. deformacijski energiji.



Slika 15: Tipični poteki intenzitet za vsakega od načinov sidranja (hh, ph, pp).

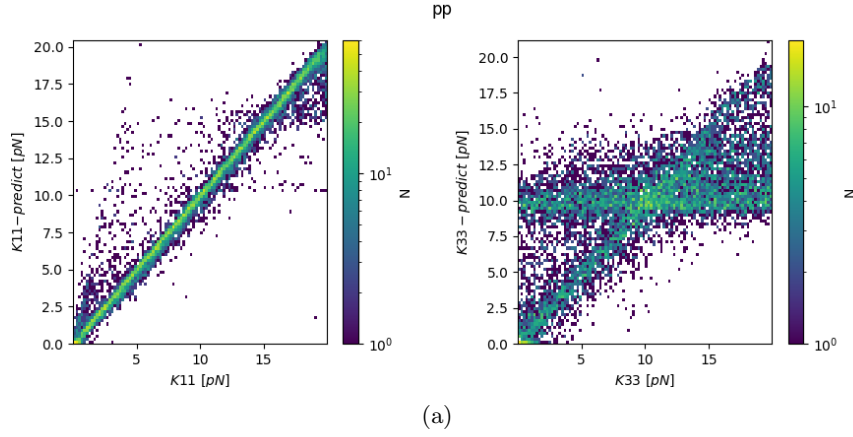


Slika 16: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Povprečna napoved (po Kfold) modela elastičnih konstant za hh sidranje. b) Povprečna napoved (po Kfold) elastičnih konstant za ph sidranje.

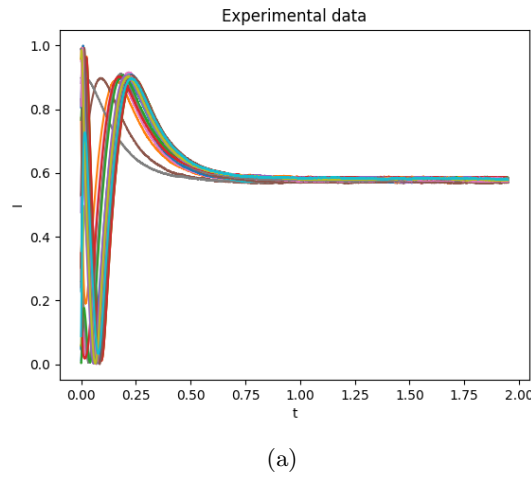
3.1 Napoved elastičnih konstant na podlagi eksperimentalnih podatkov

V zadnjem delu naloge je bilo potrebno napovedati elastične konstante eksperimentalnih intenzitet. Mreže smo učili na generiranih podatkih in nato predvideli elastične konstante na eksperimentalnih intenzitetah. Eksperimentalne intenzitete prikazuje slika 18. Ker je ima svetloba pri eksperimentalnih intenzitetah valovno dolžino $\lambda = 505 \text{ nm}$ sem uporabil za učni set intenzitete z enako valovno dolžino. Format eksperimentalnih intenzitet se razlikuje od formata učenih nevronske mreže, a to popravim z linearno interpolacijo. Eksperimentalni podatki imajo malo šuma, zato sem se odločil da uporabim nezašumljeno množico generiranih podatkov. Lahko bi dodal še zašumljene intenzitete, vendar bi potem precej podaljšal učenje mreže.

Prave vrednosti elastičnih konstant danega vzorca so $K_{11} = 6.6 \text{ pN}$ in $K_{33} = 9.0 \text{ pN}$. Najprej sem pogledal napoved 30 eksperimentalnih intenzitet na eni nevronske mreže brez navzkrižne validacije. Izkazalo se je tudi,

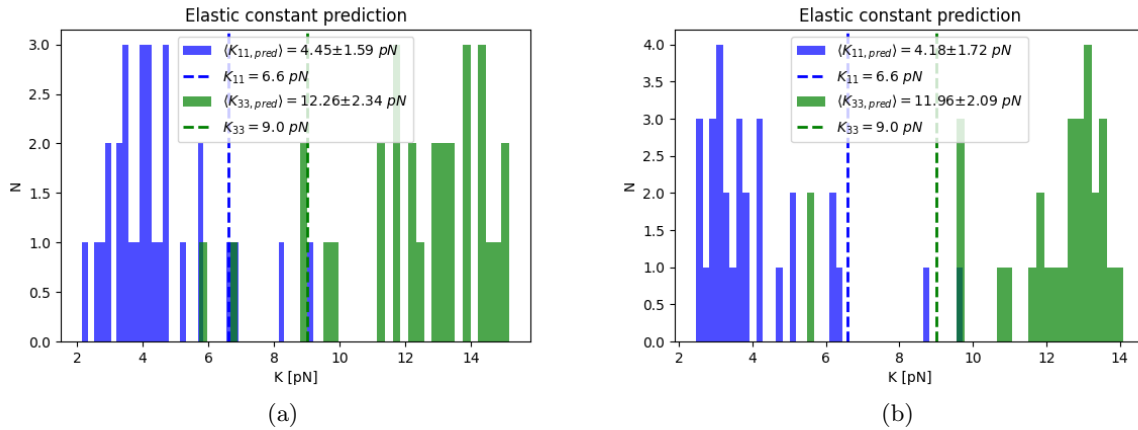


Slika 17: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Povprečna napoved (po Kfold) elastičnih konstant za pp sidranje.



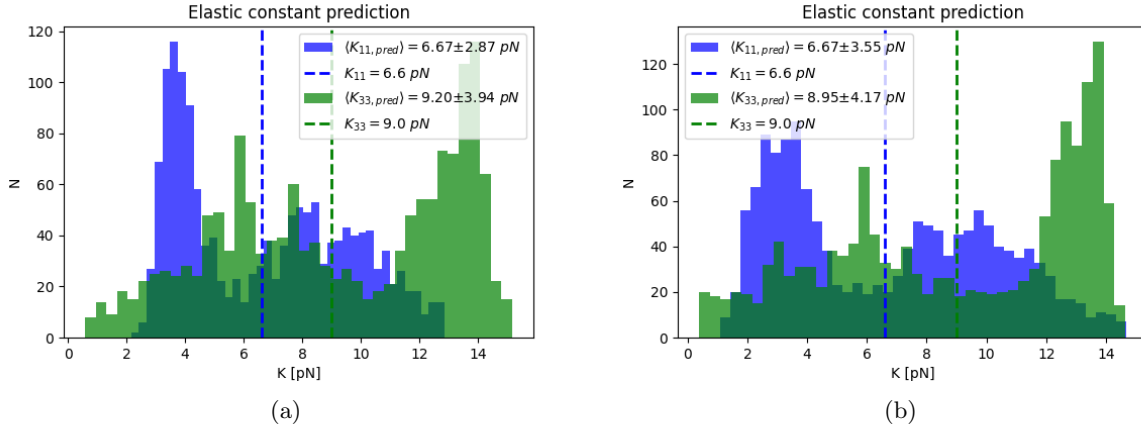
Slika 18: Eksperimentalno izmerjene intenzitete.

da povprečje navzkrižnih validacij mi ni dalo boljših napovedi, kar je mogoče malo nenavadno. Rezultati ene nevronske mreže (slika 19a) precej dobro napovejo elastično konstanto K_{11} v okviru napake. Pogledal sem tudi povprečje napovedi desetih nevronskih mrež, ki pa ne bistveni izboljšajo napovedi na tako majhni testni množici (slika 19b).



Slika 19: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Histogram napovedi elastičnih konstant K_{11} in K_{33} pri uporabi enega modela za napoved. b) Histogram napovedi elastičnih konstant K_{11} in K_{33} pri povprečenju napovedi desetih enakih modelov.

Napovedi lahko precej izboljšamo če umetno razširimo eksperimentalno testno množico intenzitet. To sem naredil tako, da sem vzel za vsako eksperimentalno intenziteto del intenzitete na intervalu $[dt\ s, 1.2 + dt\ s]$, kjer je dt iz enakomerno razdeljenega intervala 50 vrednosti $dt \in [0.0\ s, 0.1\ s]$. Na ta način sem generiral 1500 testnih intenzitet. Večje število testnih intenzitet ni izboljšalo rezultatov napovedi, saj preveč gosto vzorčenje povzroči koralcije intenzitet. Rezultati napovedi konstant z eno nevronske mreže in povprečjem desetih nevronske mreže prikazuje slika 20. Opazimo lahko, da z razširjenim testnim setom veliko bolje napovemo vrednosti konstant. Predvsem se napoved zelo izboljša pri povprečenju napovedi desetih nevronske mreže.



Slika 20: [(400,200,100,1), relu, adam, MSLE, MSE, LR(10^{-5}), E150, BS20] a) Histogram napovedi elastičnih konstant K_{11} in K_{33} pri uporabi enega modela za napoved in razširjenega seta eksperimentalnih podatkov. b) Histogram napovedi elastičnih konstant K_{11} in K_{33} pri povprečenju napovedi desetih enakih modelov in razširjenega seta eksperimentalnih podatkov.

4 Zaključek

S pomočjo nevronske mreže smo iz izmerjenih intenzitet poskušali napovedati elastične konstante tekočega kristala. Napoved nevronske mreže je z dovolj velikim učnim in testnim setom precej natančna. Največja težava je seveda izbrati optimalne parametre nevronske mreže, ki bodo optimizirali napoved elastičnih konstant. V poročilu sem izbral določeno množico hiperparametrov, vendar ta ni nujno optimalna.