Procjena kreditnog rizika malih i srednjih poduzeća probranim metodama strojnog i dubokog učenja

Krizmanić, Antonio

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Department of Mathematics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Odjel za matematiku**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:126:197359

Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku
Sveučilišni diplomski studij matematike
Smjer: Financijska matematika i statistika

Antonio Krizmanić

Procjena kreditnog rizika malih i srednjih poduzeća probranim metodama strojnog i dubokog učenja

Diplomski rad

Osijek, 2022.

Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku
Sveučilišni diplomski studij matematike
Smjer: Financijska matematika i statistika

Antonio Krizmanić

Procjena kreditnog rizika malih i srednjih poduzeća probranim metodama strojnog i dubokog učenja

Diplomski rad

Mentorica: prof. dr. sc. Nataša Šarlija
Komentor: doc. dr. sc. Slobodan Jelić

Osijek, 2022.

Sadržaj

Uvodna riječ

Ekonometrija, definirana kao kvantitativna analiza stvarnih ekonomskih pojava temeljena na istodobnim razvojima teorije i zapažanja, te uz iste vezana prikladnim metodama zaključivanja [21], kao poveznica između sfere ekonomije i sfere matematike u svojem razvoju i sazrijevanju zrcali evolucijsku trajektoriju statističke metode. Kako bismo poduprijeli navedenu tvrdnju, dovoljno je proučiti sinopsis povijesnog razvoja jedne od fundamentalnih sastavnica ekonometrije - kreditnog skoringa. Sa ciljem razrješavanja brojnih nedostataka kvalitativnim metodama utvrđenih ocjena kreditnog rizika, financijske institucije sredinom prošloga stoljeća u proces odobravanja kreditnih kartica uvode kredit skoring, koristeći pri tome primitivne skor-kartice. Neupitna uspješnost navedenoga trenda (u ranoj se fazi kredit skoring iskazao smanjenjem stope defaulta među pojedincima kojima je odobrena kreditna kartica za 50% uz grešku tipa II u iznosu od 7% [17]) rezultira migracijom kreditnog skoringa u sferu procjene rizika odobravanja zajmova, prvo za stanovništvo, a kasnije i za poduzeća. Jednostavna diskriminacijska analiza korištena pri izgradnji primitivnih skor-kartica osamdesetih godina prošloga stoljeća zamijenjena je logističkom regresijom i linearnim programiranjem - metodama koje, zahvaljujući svojoj kvaliteti i interpretabilnosti, i danas dominiraju područjem analize kreditnog rizika [25]. Proces ubrzane digitalizacije koji je obilježio 21. stoljeće i (in)direktno prožeo svako područje ljudske djelatnosti, kroz kanale porasta računalne snage, dostupnosti podataka i tehnološke pismenosti, u sferu statistike te posljedično i ekonometrije uveo je široki dijapazon metoda strojnoga učenja (engl. *machine learning*, nadalje ML) i umjetne inteligencije općenito[1]. Unatoč izrazitoj diskriminacijskoj snazi ML modela, kao i njihovoj sposobnosti otkrivanja zamršenih podatkovnih uzoraka, zbog generalno otežane interpretacije navedenim modelima generiranih predikcija zamjena standardnih regresijskih modela ML metodama i korištenje istih za možebitno neodobravanje zajma zajmotražitelju nailazi na poteškoće prouzrokovane zakonskim regulativama [12].

Time vođeni, cilj ovoga istraživanja raščlanili smo na dva dijela: usporediti performanse probranih algoritama strojnoga i dubokoga učenja izgrađenih nad podatcima o poduzećima-zajmoprimcima i njihovu statusu ispunjavanja kreditnih obaveza te provjeriti učinke selekcije prediktora na kvalitetu pojedinih modela kao jedne od fundamentalnih metoda poboljšanja transparentnosti ML modela.

Svrha istraživanja stvaranje je okosnice za možebitni nastavak analize performansi modela nad istim podatcima. Bitno je istaknuti kako je misao vodilja pri donošenju odluka glede metoda modeliranja i izbora modela poželjne performanse bila umanjiti njihovu vremensku kompleksnost – ograničeni resursi neosporno su utjecali na kvalitetu istraživanja te je zaključke rada

[1] Strojno učenje obuhvaćeno je širim područjem umjetne inteligencije, dok su metode dubokog učenja i njime obuhvaćenih neuronskih mreža podskup metoda strojnoga učenja. Kroz rad često koristimo frazu „strojno i duboko učenje“ iako bi, sintaktički gledano, bilo dovoljno sve u radu korištene metode nazivati metodama strojnoga učenja.

primjerenije uzeti za smjernice pri možebitnoj podrobnijoj analizi nego za konačan sud o performansi modela nad podatcima u pitanju. Strukturom i metodologijom, rad približno prati prijašnja istraživanja provedena u području upravljanja kreditnim rizikom u poduzetničkome kreditiranju.

Nastavak rada strukturiran je na sljedeći način. U prvome su poglavlju prezentirane osnovne karakteristike materijala, odnosno skupa podataka nad kojom je istraživanje provedeno skupa s metodologijom korištenom pri izboru predstavnika familija metoda strojnog i dubokog učenja, inženjeringu prediktora te formiranju konačnih zaključaka. U drugome poglavlju predstavljena je objedinjena analiza nekolicine sustavnih pregleda literature u području primjene metoda strojnoga učenja pri upravljanju kreditnim rizikom te prije provedenih istraživanja ciljem i sadržajem sličnih ovome. Teorijska osnova u prvome poglavlju sistematiziranih metoda korištenih pri provođenju istraživanja, među kojima je i matematička pozadina izgradnje i formiranja predikcija pojedinim ML metodama, prezentirana je u trećem poglavlju. U četvrtome poglavlju predstavljeni su rezultati istraživanja, dok je u petome poglavlju provedena rasprava glede spomenutih rezultata.

1. Materijali i metodologija

Predstavljanju metodologije istraživanja i zaključivanja prethodi prezentacija osnovnih svojstava skupa podataka nad kojem je analiza provedena. Metode ćemo po svojoj prirodi raščlaniti na metode statističke analize, metode strojnoga učenja i selekcije predstavnika familija klasifikatora[2] te metode inženjeringa prediktora.

Istraživanje smo proveli programskim jezikom Python, inačici 3.9.10, pri tome koristeći pakete Category Encoders (2.4.0), Matplotlib (3.5.1), NumPy (1.23.1), pandas (1.4.2), SciPy (1.8.0), scikit-learn (1.0.2), seaborn (0.11.2), TensorFlow (2.7.0), XGBoost (1.6.1).

1.1. Materijali

Skup podataka na kojemu je provedeno istraživanje (nadalje *uzorak*) prezentirano u ovome radu obuhvaća 3.658 jedinki, svaka od kojih predstavlja malo ili srednje poduzeće s područja Republike Hrvatske. Proces selekcije, odnosno uzorkovanja kojim je kreiran skup podataka izvršen je u pet koraka:

i) Iz baze podataka (izvor: Hrvatska gospodarska komora) svih malih i srednjih poduzeća u Hrvatskoj izabrani su oni koji su u 2019. godini bili insolventni, njih 3.207. Dostupna su bila osnovna financijska izvješća svih hrvatskih malih i srednjih poduzeća.

ii) Iz ostatka baze, slučajnim izborom odabran je isti broj solventnih poduzeća.

iii) Za svako od u koraku i) i ii) uzorkovanih 6.414 malih i srednjih poduzeća kreirana je varijabla stope rasta/smanjenja prihoda od prodaje u 2019. godini relativno na 2018. godinu. Poduzeća za koja je novokreirana varijabla bilježila nedostajuće vrijednosti su ispuštena iz uzorka. Među ostalim 4.271 poduzećem, njih 1826 je zabilježilo rast prodajne aktivnosti, odnosno pozitivnu realizaciju novokreirane varijable.

iv) Izračunati su financijski indikatori poduzeća za 2018. godinu.

v) Kako su neki od prediktora bili od većeg interesa u istraživanju, jedinke koje su za iste bilježili nedostajuće vrijednosti ispušteni su iz uzorka. Time je dobivena finalna verzija uzorka s 3.658 jedinki od kojih je 2019. godine njih 1.924 slovilo za solventna i 1.734 za insolventna poduzeća.

[2] Iako se pojam familija ML metoda u literaturi koristi pri njihovoj razdiobi temeljem tipova učenja (npr. familije metoda nadziranog, podržanoga i nenadziranog učenja) i principa učenja (npr. familije metoda učenja na temelju sličnosti, na temelju informacija, na temelju vjerojatnosti i na temelju greške), u okvirima ovoga rada ju koristimo kako bismo jasnije razlikovali tvrdnje vezane uz konkretne modele, opisane konkretnim vrijednostima hiperparametara, te skupa modela koji dijele princip izgradnje i strukturu, ali bez konkretizacije vrijednosti hiperparametara.

Uzorak smo podijelili na trening-skup i test-skup u omjeru 4:1. Trening-skup smo u sklopu algoritma peterostruke unakrsne validacije (engl. *five-fold cross validation*) te za potrebe implementacije metode ranoga zaustavljanja pri izgradnji XGBoost klasifikatora dodatno razdijelili na podatke za izgradnju klasifikatora (trening-podskup) te podatke za validaciju (validacijski skup), također u omjeru 4:1.

Svako od poduzeća okarakterizirano je pomoću 56 varijabli – 55 nezavisnih varijabli, u radu nazivanih prediktorima, te jedne zavisne varijable, odnosno varijable cilja. Pet je kvalitativnih prediktora, od kojih su tri dobivena kategorizacijom neprekidnih ekvivalenata, dok je među kvantitativnim prediktorima (od kojih glavninu čine financijski omjeri) tek jedan diskretne prirode. Financijski pokazatelji kojima je svako od poduzeća opisano navedeni su i po svojoj prirodi grupirani u *Tablici D1* i *Tablici D2* navedenima u *Dodatku*.

Makroekonomski faktori, posebice tržišni faktori, kao čimbenici koji u isti mah zahvaćaju sve zajmoprimce (ili barem znatan dio njih), bitna su sastavnica procjene poduzetničkog kreditnog rizika [14]. Tržišne promjene uzrokovane egzogenim šokovima unazad dvije godine, među kojima su najutjecajniji pandemija SARS-CoV-2 virusa i Rusko-Ukrajinski sukob, nedvojbeno su rezultirale fluktuacijama u poduzetničkim aktivnostima, pa neposredno i financijskim omjerima korištenim pri izgradnji modela. Performanse u okvirima ovoga rada analiziranih modela, pa samim time i u njemu prezentirane zaključke, stoga valja utvrditi nad ažurnijim uzorcima.

1.2. Statistička analiza

U radu korištene statističke metode svode se na procjene srednjih vrijednosti mjera performanse unutar pojedinog skupa najboljih predstavnika te jednostranog i dvostranog t-testa za testiranje pretpostavki o odnosu srednjih vrijednosti mjera performanse među skupovima najboljih predstavnika familija klasifikatora.

Srednje vrijednosti mjera performanse najboljih predstavnika familija modela procijenjene su intervalno s

$$\left(\overline{X}_n - t_{n-1,\alpha/2} \cdot \frac{S_n}{\sqrt{n}},\ \overline{X}_n + t_{n-1,\alpha/2} \cdot \frac{S_n}{\sqrt{n}}\right) \tag{1.1}$$

pri čemu je n broj klasifikatora u skupu klasifikatora za koji procjenjujemo srednju vrijednost, $\alpha = 0,05$ razina značajnosti, $\overline{X}_n$ uzoračka srednja vrijednost, S_n uzoračka standardna devijacija te $t_{n-1,\alpha/2}$ $\frac{\alpha}{2}$-kvantil studentove t-distribucije s $n-1$ stupnjeva slobode [24]. U nastavku ćemo procjenu (1.1) sažeto zapisivati u obliku $\overline{X}_n \pm t_{n-1,\alpha/2} \cdot \frac{S_n}{\sqrt{n}}$.

Razina značajnosti za koju će se prihvaćati ili odbacivati nul-hipoteze jednostranih i dvostranih t-testova (u njihovoj standardiziranoj formi, v. [9]) iznosi 0,05. Odnose među srednjim vrijednostima mjera performansi utvrđivat ćemo isključivo nad test-podatcima.

Mjere performanse modela (klasifikacijske metrike) podijelili smo na mjere diskriminacijske snage i mjere klasifikacijske točnosti. Diskriminacijsku smo snagu mjerili površinom ispod ROC

krivulje (engl. *area under the ROC curve*, nadalje AUC vrijednost). Klasifikacijsku točnost najboljih modela mjerili smo pozitivnom i negativnom prediktivnom vrijednošću (eng. *positive and negative predictive value*, nadalje redom PPV i NPV) te točnošću (eng. *accuracy*, nadalje ACC). Vjerojatnosti prag (eng. *cutoff value*) s obzirom na kojeg smo zajmoprimce klasificirali kao dobre ili kao loše odredili smo minimizacijom udaljenosti između osjetljivosti (engl. *sensitivity*, nadalje O) i specifičnosti (engl. *specificity*, nadalje S) pojedinog modela nad trening-podatcima.

1.3. Binarni klasifikatori i selekcija kandidata

Pri provođenju istraživanja uspoređene su performanse binarnih klasifikatora iz četiri familije ML modela: klasifikacijskih i regresijskih stabala (engl. *classification and regression trees*, nadalje CART), slučajne šume (engl. *random forest*, nadalje RF), algoritama podizanja stabala, konkretno ekstremnog podizanja gradijenta (engl. *extreme gradient boosting*, nadalje XGBoost) te višeslojnih perceptron neuronskih mreža (engl. *multilayer perceptron artificial neural network*, nadalje ANN). Ne bismo li osigurali što veću reprezentativnost performansi finalnih verzija modela, za svaku smo familiju proveli pretragu mreže hiperparametara s peterostrukom unakrsnom validacijom (engl. *parameter grid-search with five-fold cross-validation*) kreirajući pri tome 4.608 različitih CART, 11.520 RF, 4.500 XGBoost te 3.888 ANN binarnih klasifikatora. Ispod je opisana trodijelna evaluacija performansi pretragom mreže hiperparametara izgrađenih modela – unutar familije modela i među familijama modela.

- Iz mreže modela smo za svaku familiju ML algoritama izdvojili vodećih 1% modela (nadalje *najboljih 1% modela*), odnosno modele čija je pretragom mreže hiperparametara zabilježena prosječna AUC vrijednost veća od ili jednaka s 0.99-kvantilom prosječnih AUC vrijednosti modela iz iste familije. Njihove smo performanse potom statističkim testovima usporedili nad test-podatcima.

- Iz mreže modela smo za svaku familiju ML algoritama izdvojili najbolji model, odnosno model čija je pretragom mreže hiperparametara zabilježena prosječna AUC vrijednost jednaka maksimalnoj vrijednosti prosječnih AUC vrijednosti modela iz iste familije. Njihovu smo performansu potom usporedili nad test-podatcima.

- Sekvencijskom selekcijom prediktora unaprijed (engl. *forward sequential feature selection*) smo izdvojili deset prediktora koji najviše doprinose kvaliteti najboljih predstavnika četiriju familija modela. Njihovu smo performansu potom usporedili nad test-podatcima, usredotočivši se isključivo nad odabrane najrelevantnije prediktore.

Diskretnu mrežu vrijednosti hiperparametara analiziranih algoritmom pretrage mreže smo izgradili kroz četiri koraka:

i) **Za familije modela definirali smo inicijalne skupove vrijednosti hiperparametara.** Iako je njihova definicija vođena praksom, ona je i dalje suštinski arbitrarna – za

odabir skupa vrijednosti pojedinih hiperparametara postoje određene smjernice, ali zbog stohastičke prirode problema optimizacije, one nisu konkretizirane.

ii) **Za svaku familiju modela smo probrali hiperparametre koji najviše utječu na vremensku složenost pretrage mreže.**

iii) **Skupove vrijednosti probranih hiperparametara sveli smo na pripadajuće podskupove za koje je volumenom manjom pretragom mreže zabilježen najbolji prosjek AUC vrijednosti.** Konkretno, utjecaji fluktuacija svakog od probranih hiperparametara na performansu modela razmotrene su zasebno. Ostali su hiperparametri, izuzev mjera nečistoće u CART i RF klasifikatoru te aktivacijskih funkcija u ANN klasifikatoru čijim bismo fiksiranjem uzrokovali značajnu pristranost u zaključcima, fiksirani na medijan skupova vrijednosti u slučaju numeričkih te u praksi češće korištene vrijednosti u slučaju kvalitativnih hiperparametara.

iv) **Proveli smo finalnu pretragu mreže hiperparametara nad pripadnim novodefiniranim skupovima vrijednosti.**

Osim hiperparametara čiji su skupovi vrijednosti apriorno analizirani sa ciljem umanjenja vremenske složenosti algoritma pretrage mreže hiperparametara, analiziran je i parametar rezidbe CART klasifikatora nc bismo li utvrdili kojc njcgovc vrijcdnosti rczultiraju podnaučcnošću (cngl. *underfitting*) modela nad trening-podatcima. U tom smo slučaju, za razliku od 3. koraka iznad opisanog procesa, fiksirali vrijednosti svih hiperparametara isključujući maksimalnu dubinu, parametar rezidbe i hiperparametar mjere nečistoće. U *Tablici 1.2* su navedeni hiperparametri čiji su skupovi vrijednosti apriorno analizirani, bilo zbog njihova utjecaja na vremensku složenost algoritma pretrage mreže ili zbog prevencije podnaučenosti.

Familija klasifikatora	**Apriorno analizirani hiperparametri**
CART	Maksimalna dubina stabla, parametar rezidbe
RF	Maksimalna dubina stabala, broj stabala u šumi
XGBoost	Maksimalna dubina stabala
ANN	Struktura mreže: broj skrivenih slojeva i broj čvorova po sloju

Tablica 1.1: Hiperarametri čije su vrijednosti analizirane prije provođenja algoritma pretrage mreže hiperparametara.

1.4. Inženjering prediktora

XGBoost klasifikator može biti izgrađen te kasnije formirati predikcije i kada prediktori bilježe nedostajuće (izostavljene) vrijednosti. Izgradnja i korištenje ostalih klasifikatora zahtijevaju zamjenu nedostajućih vrijednosti konkretnim numeričkim vrijednostima, što smo u okvirima

ovoga rada izbršili metodom imputacije srednjom vrijednošću (engl. *mean imputation*).

Realizacije kvalitativnih prediktora praksa je preslikati u prostor realnih brojeva, iako neki od razmatranih klasifikatora dopuštaju izgradnju i formiranje predikcija i kada su prediktori kategorički. Postupak kojim se realizacije kvalitativnih prediktora preslikavaju u konkretne numeričke ekvivalente u praksi se naziva kodiranje (engl. *encoding*). U okvirima ovoga rada korišteno je ordinalno kodiranje (poznato i kao kodiranje oznaka kategorije, engl. label *encoding*) te kodiranje srednjom vrijednošću (engl. *mean target encoding*).

2. Pregled literature

Postojeća literatura igrala je dvojaku ulogu, kako pri selekciji familija modela koje smo međusobno suprotstavili u okvirima ovoga rada te formiranju slutnji glede njihovih performansi, tako i pri uspostavljanju ograničenja i nedostataka same analize. Kako bi navedeni fundamenti istraživanja bili što kvalitetniji, formirani su na temelju sinteze zaključaka sustavnih pregleda literature iz područja upravljanja kreditnim rizikom različitim klasičnim statističkim i ML metodama te konkretnih istraživanja ciljem i sadržajem sličnih ovome.

2.1. Sustavni pregledi literature o kreditnim rizicima

Kreditiranje, jedna od primarnih usluga bankarskog sektora, primarni je izvor prihoda bankarskih i srodnih institucija. Protutežu nastojanju maksimizacije profitabilnosti djelovanja financijskih institucija kroz odobravanje što je moguće većeg volumena zajmova čine rizici kojima se institucije izlažu pozajmljivanjem vlastitoga kapitala zajmoprimcima od kojih je, u smislu financijskih gubitaka, najznačajniji rizik neizvršavanja obaveza od strane zajmoprimca, u literaturi poznat kao kreditni rizik [14]. Interes financijskih institucija za inkorporacijom metoda strojnoga učenja u njihovo šire djelovanje kao i interes podatkovnih znanstvenika za analizom kvalitete strukturalno različitih algoritama strojnoga učenja u izvršavanju pojedinih zadataka unazad tri desetljeća kontinuirano raste [2].

Međutim, kvaliteta provedenih studija kojima se algoritmi strojnoga učenja u kreditnome skoringu nastoje evaluirati i međusobno usporediti znatno varira. *Baesens, Lessmann, Seow i Thomas* 2015. godine ukazuju na pet krucijalnih nedostataka među istraživanjima provedenim u području upravljanja potrošačkim kreditnim rizikom korištenjem metoda strojnoga učenja. Pregledom 48 znanstvenih radova objavljenih u razdoblju od 2003. do 2014. godine autori utvrđuju kako su, u prosjeku, performanse suprotstavljenih modela evaluirane i uspoređene nad nešto manje od dva različita skupa podataka (1,9 različitih skupova), da prosječni skup podataka nad kojim je pojedini model izgrađen i evaluiran čini 6.167 jedinki te da prosječan broj varijabli kojima su zajmoprimci opisani iznosi 24 [5]. *Ara, Louzada i Fernandes* su 2016. zabilježili sličnu vrijednost prosječnog broja skupova podataka korištenih pri evaluaciji performansi modela (2,18 skupova podataka po članku) ističući kako je ista rasla kroz vrijeme [2]. *Baesens*

et al. (2015) navode kako su iznad navedene vrijednosti nezadovoljavajuće velike - evaluacija performansi klasifikatora nad većim brojem skupova podataka omogućuje provjeru robusnosti istoga dok veći broj jedinki u uzorku kao i veći broj prediktora kojima je svaka jedinka opisana osiguravaju vanjsku valjanost empirijskih rezultata. Ističemo kako istraživanje stoga valja proširiti dodatnim skupovima podataka veće dimenzije vektora prediktora.

Autori ukazuju i na važnost korištenja suštinski različitih mjera kvalitete modela pri evaluaciji klasifikacijske performanse. Konkretnije, autori mjere performanse dijele na mjere diskriminacijske snage (npr. AUC i KS statistika), mjere kojima ocjenjujemo preciznost predikcija vjerojatnosti skor-kartica (npr. Brier score) te mjere kojima ocjenjujemo točnost kategoričkih predikcija skor-kartice, odnosno točnost klasifikacije (npr. greška klasifikacije i točnost). Time vođeni smo performanse modela odlučili evaluirati mjerama diskriminacijske snage i klasifikacijske točnosti.

Stohastička narav izgradnje modela za predikciju otežava formiranje kvalitetnih zaključaka glede generalne kvalitete modela, odnosno kvalitete modela izvan okvira pojedine analize slučaja. U jednu ruku, kompleksnost proizlazi iz same prirode problema, ilustrirana činjenicom kako neznatna promjena u prostoru hiperparametara i načinu njegova pretraživanja ili primjena dodatnih regularizacijskih metoda mogu znatno poboljšati modele koji su prije bilježili lošu performansu. U drugu ruku, ista je dijelom rezultat načina na koji se istraživanja provode - kako ističu *Baesens et al.* (2015), autori radova nerijetko arbitrarno izabiru modele koje međusobno suprotstavljaju, pri selekciji ignorirajući najsuvremenije metode, ili pak suprotstavljaju probrane metode vlastitim metodama, uz određene pristranosti koje autori ne uzimaju u obzir. Dodatnu dimenziju problemu daje činjenica kako su metode u glavnini slučajeva uspoređene nad privatnim bazama podataka (*Ara et al.*, 2016) – kako isključivo autori imaju pristup podatcima, na temelju kojih nerijetko provode tek jednu studiju, obogaćivanje analize dodatnim metodama i mjerama performansi u tom je slučaju gotovo pa nemoguća. Konsenzusi glede kvalitete pojedinih modela, barem u kontekstu klasifikacijske snage, u stručnoj literaturi stoga dolaze tek u generalnim smjernicama formiranih nad skupovima podataka učestalo korištenima u srodnim znanstvenim radovima[3]. Među smjernicama, dvije su igrale ključnu ulogu pri selekciji klasifikatora razmotrenih u okvirima ovoga rada, kao i formiranju slutnji glede konačnih rezultata.

Prva se smjernica tiče jaza u performansama pojedinačnih algoritama i ansambla algoritama. *Celik, Dastile i Potsane* [12] 2020. godine te *D'Addona, Luo, Pau, Shi i Tse* [15] 2022. godine objavljuju sustavne preglede literature prikupljene iz istih pet repozitorija znanstvenih i stručnih radova, pri njezinu prikupljanju koristeći različite ključne riječi. Usprkos činjenicama kako nad istim skupovima podataka bilježe generalno drugačije hijerarhije performanse modela te kako u oba slučaja poredak modela ovisi o izboru metode kvantifikacije performansi klasifika-

[3]U ovome potpoglavlju razmotreni sustavni pregledi literature pri statističkoj analizi u obzir uzimaju (isključivo ili tek dijelom) znanstvene radove koji su kvalitete metoda evaluirali nad barem jednom od dva učestalo korištena javno dostupna skupa podataka – *German Credit Dana* (nadalje Njemački podatci) i *Australian Credit Approval* (nadalje Australski podatci) dostupnima u UC Irvine repozitoriju za strojno učenje.

tora, autori zaključuju kako ansambl metode uvijek nadmašuju pojedinačne algoritme binarne klasifikacije. Isti su trend zabilježili i *Baesens et al* (2015).

Druga se smjernica tiče jaza u performansama klasičnih statističkih klasifikacijskih metoda, od kojih je u sferi upravljanja kreditnim rizikom neupitno najrasprostranjeniji logistički regresijski model (nadalje LR model), i metoda strojnoga učenja. *Baesens et al.* (2015) utvrđuju kako LR bez regularizacije ostvaruje bolju prosječnu performansu od pojedinačnih ML algoritama, izuzev ANN, ali kako ansambl metode generalno bilježe bolju performansu. Za kontekst našega rada je bitno istaknuti kako je prosječna performansa LR modela veća od prosječne performanse CART klasifikatora i klasifikatora iz familije stohastičkog podizanja gradijenta, kojima je srodan XGBoost klasifikator, te manja od prosječne performanse ANN i RF klasifikatora.

2.2. Srodna istraživanja - poduzetnički kreditni rizici

Sustavni pregledi literature mahom obuhvaćaju istraživačke radove u čijem je fokusu analiza i(li) usporedba ML metoda u području analize kreditnog rizika za potrošače, odnosno stanovništvo. Opservacije glede značajnog povećanja kvalitete zaključaka povećanjem broja različitih skupova podatak i njihove dimenzionalnosti nadilaze sferu upravljanja potrošačkim kreditnim rizikom – kako iste općenito vrijede za analizu tabelarnih podataka, trivijalno se prevode i u problem upravljanja poduzetničkim kreditnim rizikom. Međutim, specifičnosti upravljanja poduzetničkim kreditnim rizikom moguće povlače promjene u hijerarhiji kvalitete među binarnim klasifikatorima predstavljenoj u prošlome potpoglavlju. Kako glavninu prediktora kojima su opisana poduzeća čine financijski omjeri koji pak zahtijevaju kreiranje novih prediktora transformacijama bazom inicijalno obuhvaćenih prediktora, za očekivati je znatno veća multikolinearnost no što je slučaj u bazama kojima je opisano stanovništvo. Dodatno, kako se poduzeće pretežno karakterizira informacijama sadržanim u pripadnim financijskim izvješćima, udio kvantitativnih varijabli veći je no što je to slučaj sa stanovništvom. Način oblikovanja prediktora također otvara mogućnost većeg udjela nedostajućih vrijednosti no što je to slučaj za analizu rizika za potrošače. *Baesens et al.* (2015) tako naglašavaju da „razlike u tipovima varijabli sugeriraju da se specifični izazovi pri modeliranju javljaju u potrošačkom i korporativnom kreditnom skoringu".

Ne bismo li upotpunili pregled literature te ga ujedno učinili relevantnijim za kontekst ovoga rada, analizirat ćemo konkretne volumenom i resursima bogatije studije performansi ML metoda pri analizi poduzetničkog kreditnog rizika. Osnovne informacije o analiziranim studijama navedene su u *Tablici 2.1.* Istaknimo kako autori nerijetko razmatraju i dodatne ML metode, ali da smo se odlučili usredotočiti samo na one ključne za potrebe ovoga istraživanja.

Rezultati u tablici navedenih triju članaka idu podruku sa zaključcima predstavljenim u prethodnom potpoglavlju – LR modeli i ostale klasične statističke metode bilježe lošiju performansu od glavnine ML metoda dok se među spomenutima svojom performansom ističu ansambl metode i kompleksne neuronske mreže. *Vidovic et al.* (2020) navode da, usprkos njegovoj jed-

Članak	Godina objave	Veličina uzorka	Broj prediktora	Lokacija uzorkovanja	Uzorkovani period	Modeli
Klamargias, Petropoulos, Siakoulis, Stavroulakis [20]	2018.	200.000	65	Grčka	2005. - 2015.	LR, XGBoost, ANN
Addo, Guegan, Hassani [1]	2018.	117.019	181	Nepoznato	2016. - 2017.	LR, RF, MPG, ANN
Vidovic, Yue [26]	2020.	52.500	11	Globalno	2002. - 2016.	LR, CART

Tablica 2.1: Osnovne informacije o istraživanjima sličnima ovome čiji su zaključci sintetizirani u nastavku. *MPG = metoda podizanja gradijenta*

nostavnosti, regularizirani CART klasifikator diskriminacijskom snagom (AUC = 94,80% nad test-podatcima) nadilazi LR model (AUC = 93,60% nad test-podatcima), ali kako isti bilježi značajnu prenaučenost (AUC = 99,80% nad trening-podatcima). Autori su pokazatelje financijskih rizika upotpunili pokazateljima poslovnih rizika u koje su uključene i apriorno utvrđene mjere rizičnosti kreditiranja na nacionalnoj razini (*Country Risk Score*) i razini branše (*Industry Risk Score*) te su selekcijom po jednog predstavnika svake grupe financijskih pokazatelja ograničili multikoreliranost među prediktorima, što je zauzvrat rezultiralo poboljšanjem performanse modela. *Addo et al.* (2018) bilježe znatno veći jaz između performanse LR modela i performansi ML klasifikatora. Performansa LR modela (AUC = 87,63% nad test-podatcima) nadišla je tek neke od strukturom, odnosno brojem parametara, jednostavnijih višeslojnih perceptrona. Performansa ANN modela ovisi o njegovoj kompleksnosti, točnije broju parametara – razmotreni model s najmanjim brojem parametara lošije je diskriminacijske snage od LR modela (AUC = 84,12% nad test-podatcima) dok je model s najvećim brojem parametara nad test-podatcima postigao AUC vrijednost od 97,53%. Kompenzacija za razmjerno velik broj parametara u razmotrenim višeslojnim perceptronima dolazi u obliku trojake regularizacije – metodom ranog zaustavljanja, L1 regularizacijom i L2 regularizacijom. RF klasifikator i metoda podizanja gradijenta ostvaruju dovoljno blisku performansu da ih autori smatraju jednako kvalitetnima (AUC nad test-podatcima redom 99,31% i 99,48%). Nakon uspostave odnosa u performansama modela nad cijelim skupom prediktora, autori za svaki od modela provode selekciju deset prediktora najveće prediktivne snage te ponovo evaluiraju modele nad test-podatcima. Selekcija prediktora najviše je utjecala na performansu LR modela i prosječnu performansu perceptrona (ΔAUC $> 20\%$) dok su slučajna šuma i metoda podizanja gradijenta zabilježile neznatno pogoršanje diskriminacijske snage (ΔAUC $< 1\%$). Isti odnos u performansama analiziranih metoda zabilježili su i *Klamargias et al.* (2018) – nad test podatcima, LR bilježi AUC u vrijednosti od 66,00%, ANN u vrijednosti od 72,00% te XGBoost u vrijednosti od 78,00%.

Sinteza u ovome potpoglavlju prezentiranih zaključaka istraživanja srodnih istraživanju pre-

zentiranome u ovome radu, doduše volumenom i heterogenošću podataka znatno većih, ide podruku sa zaključcima iz prethodnog potpoglavlja – logistička regresija, iako i dalje zbog svoje robusnosti i interpretabilnosti najkorištenija metoda upravljanja poduzetničkim i potrošačkim kreditnim rizikom, bilježi znatno lošije rezultate kada joj suprotstavimo performansu ML metoda, posebice kada istu mjerimo diskriminacijskom snagom. S obzirom na to da volumen ovoga rada nije dopustio podobniju analizu LR modela, isti smo ispustiti i prednost dati probranim ML metodama različite složenosti. Napomenimo kako provedeno istraživanje valja proširiti dodatnim ML metodama, među kojima naglasak stavljamo na SVM - *Ara et al* (2016) navode kako se „među razmotrenim metodama SVM ističe kao metoda visoke prediktivne performanse i niske računalne složenosti", te klasičnim statističkim metodama, među kojima naglasak stavljamo na logističku regresiju.

U studijima sličnim ovoj formiranje hipoteza od upitnog je značaja. Tvrdnju podupiremo u ovome poglavlju prezentiranom raspravom o nepostojanju konsenzusa o hijerarhiji kvalitete među ML metodama te kako čak i nad istim podatcima autori dobivaju različite, izrazito varijabilne rezultate. Usprkos tome, oslanjajući se na generalne smjernice glede veće kvalitete ansambl metoda u odnosu na pojedinačne metode te rezultata istraživanja koje su proveli *Vidovic et al.* (2020) i *Addo et al.* (2018), formiramo hipoteze glede relativne performanse pri ovome istraživanju suprotstavljenih četiriju (familija) klasifikatora:

- CART klasifikator će, zbog svoje izrazite jednostavnosti, ostvariti najlošiju performansu.
- ANN klasifikator će ostvariti bolju performansu od CART model, ali će ista i dalje biti lošija od one RF i XGBoost klasifikatora, kako zbog malenog, šumnog uzorka, tako i zbog činjenice kako su kasnije spomenuti, za razliku od ANN klasifikatora, ansambl metode.
- RF klasifikator i XGBoost klasifikator će ostvariti najbolje performanse. Odnos među njima ne možemo unaprijed odrediti.

3. Teorijska pozadina

U ovome je poglavlju predstavljena teorijska pozadina iza metoda transformacije i selekcije prediktora te iza izgradnje (treniranja, često nazivanog i *fittingom*) i regularizacije klasifikacijskih metoda kao i načina na koje istima procjenjujemo vjerojatnost da je konkretan zajmoprimac loš. Notacija i matematička nomenklatura povremeno je pojednostavljena, ponekad narušavajući matematičku preciznost tvrdnji, ali odlučili smo prednost dati interpretabilnosti i razumljivosti terminologije kako bi čitatelji koji se po prvi put susreću sa sadržajem u pitanju lakše razumjeli njegovu srž.

Označimo s $\mathbb{X} = (X_1, X_2, \dots X_D)$ slučajni vektor prediktora kojima su opisane jedinke u uzorku i s Y ciljanu varijablu. Neka je za $j \in \{1, 2, \dots N\}$ s $\mathbf{a}^{(j)} = \left(x_1^{(j)}, x_2^{(j)}, \dots, x_D^{(j)}, y^{(j)}\right) =$

$\left(\mathbf{x}^{(j)}, y^{(j)}\right)$ označena realizacija vektora prediktora i ciljane varijable za j-to poduzeće. Varijabla Y poprima vrijednosti iz skupa $\{0, 1\}$, pri čemu njezina realizacija jediničnom vrijednošću implicira kako je zajmoprimac loš. Kada budemo predstavljali principe treniranja, odnosno izgradnje pojedinih klasifikatora, pretpostavit ćemo kako iste izgrađujemo nad cijelim skupom zajmoprimaca $\mathbf{a}^{(1)}, \ldots, \mathbf{a}^{(N)}$.

3.1. Pokazatelji klasifikacijske točnosti

Kako je istaknuto u poglavlju *Pregled literature*, performansu modela teoretski je neopravdano i praktički neadekvatno kvantificirati isključivo jednom mjerom performanse modela. Kako AUC vrijednost dopušta uvid isključivo u diskriminacijsku snagu modela, istu valja nadopuniti mjerama klasifikacijske točnosti ne bismo li dobili potpuniji uvid u kvalitetu modela. Najčešće korišteni pokazatelji mjere točnosti izvedenice su informacija sadržanih u matrici zabune (engl. *confusion matrix*) – tabelarne organizacije binarnim klasifikatorom formiranih predikcija $\hat{y}^{(j)}$ u četiri kategorije, ovisno o tome podudara li se pojedina predikcija s korespondentnom u skupu podataka zabilježenom realizacijom varijable cilja $y^{(j)}$ te ovisno o tome koja je kategorija dodijeljena zajmoprimcu. Kako pri usporedbi najboljih predstavnika razmotrenih familija klasifikatora nećemo bilježiti pripadnu matricu zbunjenosti, korištene mjere klasifikacijske točnosti ćemo definirati bez oslanjanja na njihovu tabličnu formu (v. [22]). Definirajmo sljedeće vrijednosti:

- $TP \coloneqq \left|\left\{j \in \{1, \ldots, N\} : y^{(j)} = 1, \hat{y}^{(j)} = 1\right\}\right|$ kao broj loših zajmoprimaca koji su klasifikatorom kategorizirani kao loši (engl. *true positive*),
- $TN \coloneqq \left|\left\{j \in \{1, \ldots, N\} : y^{(j)} = 0, \hat{y}^{(j)} = 0\right\}\right|$ kao broj dobrih zajmoprimaca koji su klasifikatorom kategorizirani kao dobri (engl. *true negative*),
- $FP \coloneqq \left|\left\{j \in \{1, \ldots, N\} : y^{(j)} = 0, \hat{y}^{(j)} = 1\right\}\right|$ kao broj dobrih zajmoprimaca koji su klasifikatorom kategorizirani kao loši (engl. *false positive*) te
- $FN \coloneqq \left|\left\{j \in \{1, \ldots, N\} : y^{(j)} = 1, \hat{y}^{(j)} = 0\right\}\right|$ kao broj loših zajmoprimaca koji su klasifikatorom kategorizirani kao dobri (engl. *false negative*).

Pokazatelji klasifikacijske točnosti korišteni u ovome radu su točnost (ACC), osjetljivost (O), specifičnost (S) te pozitivna i negativna prediktivna vrijednost (redom PPV i NPV).

Točnost predstavlja udio točno klasificiranih jedinki, neovisno o kategoriji koja im je dodijeljena, te se određuje kao

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}.$$

Kako je točnost izuzetno osjetljiva na neuravnoteženost u podatcima, praksa je zasebno se usredotočimo na točnost kategorizacije za loše i dobre zajmoprimce kroz tzv. uparene mjere.

Prvi par uparenih mjera čine osjetljivost i specifičnost koje možemo interpretirati kao mjere

točnosti uočavanja dobrih i loših zajmoprimaca u populaciji. Dok osjetljivost predstavlja udio točno kategoriziranih loših zajmoprimaca među svim lošim zajmoprimcima, osjetljivost analogno predstavlja udio točno kategoriziranih dobrih zajmoprimaca među svim dobrim zajmoprimcima. Konkretnije,

$$O = \frac{TP}{TP + FN}, \quad S = \frac{TN}{TN + FP}.$$

Drugi par uparenih mjera čine pozitivna i negativna prediktivna vrijednost koje možemo interpretirati kao mjeru točnosti razvrstavanja zajmoprimaca na dobre i loše. Dok pozitivna prediktivna vrijednost predstavlja udio točno kategoriziranih loših zajmoprimaca među svim zajmoprimcima koji su kategorizirani kao loši, negativna prediktivna vrijednost analogno predstavlja udio točno kategoriziranih dobrih zajmoprimaca među svim zajmoprimcima koji su kategorizirani kao dobri. Konkretnije,

$$PPV = \frac{TP}{TP + FP}, \quad NPV = \frac{TN}{TN + FN}.$$

Za razliku od AUC vrijednosti, pri čijoj se procjeni koriste predikcije vjerojatnosti da su zajmoprimci loši (predikcije kakve su razmotrene pri raščlambi binarnih kategorizacijskih metoda u kasnijim potpoglavljima ovoga poglavlja), pokazatelji kategorizacijske točnosti zahtijevaju diskretizaciju neprekidne procjene vjerojatnosti insolventnosti u skup s dvije vrijednosti. U okvirima ovoga istraživanja, spomenuta diskretizacija provedena je na temelju granične vrijednosti $\xi \in [0, 1]$ – zajmoprimci za koje se procjena vjerojatnosti insolventnosti realizirala vrijednošću iz intervala $[0, \xi\rangle$ kategorizirani su kao dobri, dok su ostali zajmoprimci kategorizirani kao loši.

Sve vrijednosti obuhvaćene tablicom zabune kao i izvedene mjere kvalitete kategorizacije osjetljive su na izbor (hiper)parametra granične vrijednosti, odnosno iste mogu biti razmatrane kao funkcije parametra ξ. Kompleksnost problema određivanja optimalne granične vrijednosti detaljnije je prezentirana u referiranoj literaturi, kao i pojašnjenja kako se iznad navedeni pokazatelji mogu učiniti robusnijima na izbor parametra ξ. U okvirima rada parametar ξ izabrali smo kao vrijednost koja minimizira $|O(\xi) - S(\xi)|$ nad trening-podatcima.

3.2. Kodiranje podataka

Odluka o metodama kodiranja kategoričkih varijabli donesena je kako na temelju prirode razmatranih kvalitativnih prediktora, tako i na temelju ML metoda koje smo u okvirima istraživanja suprotstavili jednu drugoj. Binarni klasifikatori temeljeni na stablima predikcije formiraju biparticijom područja vrijednosti slučajnih varijabli tako da maksimum jednog skupa bude manji od minimuma drugoga skupa. Korištenje neadekvatnih metoda kodiranja može rezultirati svojevrsnim forsiranjem uređaja među kategorijama kvalitativne varijable kada isti nije prisutan, uzrokujući pristranost koja narušava prediktivnu snagu modela i njegovu općenitu kvalitetu.

Neka je za neki $k \in \{1, \ldots, N\}$ odgovarajuća varijabla X_k kvalitativna. Pretpostavimo

kako ista vrši particiju skupa poduzeća na p disjunktnih skupova (u literaturi poznatijih kao kategorije) $C_1, \ldots, C_p$ na temelju nekog kvalitativnog svojstva. Po potrebi, jedna od p kategorija obuhvaća poduzeća za koja varijabla X_k bilježi nedostajuće vrijednosti. Pojednostavljenja radi, činjenicu da se varijabla X_k za j-to poduzeće realizirala kategorijom C_i, donosno kako je $x_k^{(j)} = C_i$, ćemo označavati s $\mathbf{a}^{(j)} \in C_i$ te ćemo reći kako j-to poduzeće pripada kategoriji C_i.

Ordinalno kodiranje, poznato i kao kodiranje po vrijednosti (engl. *value encoding*), jedan je od primitivnijih, ali i dalje u praksi korištenih principa kodiranja kvalitativnih varijabli. U okvirima ovoga rada je ordinalno kodiranje korišteno kada skup kategorija kvalitativne varijable dozvoljava intuitivnu, prirodnu definiciju relacije uređaja, odnosno kada možemo tvrditi kako *jedna kategorija po vrijednosti dolazi prije druge kategorije*. Valja uočiti kako bi ordinalno kodiranje varijabli koje usprkos prirodnom uređaju kategorija bilježe nedostajuće vrijednosti, a kakvih u razmatranome uzorku nije bilo, bilo nemoguće bez određene stope arbitrarnosti. Karakterističan primjer kvalitativnih varijabli nad kojima postoji prirodan uređaj su varijable nastale kategorizacijom neprekidnih varijabli – transformacijom neprekidnih varijabli kojima se iste diskretiziraju segmentacijom pripadnog skupa vrijednosti. Prirodan uređaj kategorija u tom je slučaju uspostavljen odnosom vrijednosti u jednom segmentu s vrijednostima u drugom segmentu.

Označimo relaciju uređaja na skupu kategorija $\{C_1, \ldots, C_p\}$ varijable X_k s $\preceq$ te definirajmo permutaciju $\pi : \{1, \ldots, p\} \to \{1, \ldots, p\}$ koja zadovoljava $C_{\pi(1)} \preceq C_{\pi(2)} \preceq \ldots \preceq C_{\pi(p)}$. Ako definiramo preslikavanje $Ord : \{C_1, \ldots, C_p\} \to \{1, \ldots, p\}$ s pravilom pridruživanja $Ord(C_i) = \pi(i)$ za $i \in \{1, \ldots, p\}$, ordinalno kodiranje varijable X_k možemo definirati kao njezinu transformaciju $Ord(X_k)$[4].

Kodiranje srednjom vrijednošću [23] omogućuje da pri samome procesu kodiranja utjecaj činjenice kako jedinka pripada kategoriji varijable na vjerojatnost insolventnosti u većoj ili manjoj mjeri generaliziramo vjerojatnošću insolventnosti za cijeli uzorak. U jednu ruku, kategorije koje obuhvaćaju tek nekolicinu jedinki generaliziramo, odnosno kodiramo vrijednošću karakterističnom za cijeli uzorak kako bismo onemogućili možebitnim ekstremnim realizacijama varijable da impliciraju uređaj među kategorijama. U drugu ruku, kategorije koje sadrže dovoljan broj jedinki kodiramo vrijednostima karakterističnim za tu konkretnu kategoriju. Formalno, kodiranje srednjom vrijednošću definirano kao transformaciju slučajne varijable X_k preslikavanjem $ksv : \{C_1, \ldots, C_p\} \to [0, 1]$, konveksnom kombinacijom stope insolventnosti poduzeća unutar kategorije i stope insolventnosti poduzeća u cijelome uzorku. Neka je N_i broj jedinki u i-toj

[4]Općenito, funkcija *Odj* iz definicije ordinalnog kodiranja kvalitativne varijable može biti bilo koje bijektivno preslikavanje $\{C_1, \ldots, C_p\} \to \{1, \ldots, p\}$. Naša definicija prilagođena je kontekstu u kojem je ordinalno kodiranje korišteno u radu: kada postoji suštinski uređaj među kategorijama.

kategoriji C_i kvalitativne varijable X_k. Pravilo pridruživanja funkcije ksv dano je s

$$ksv(C_i) = \alpha(N_i)\frac{1}{N_i}\sum_{\left\{j:\mathbf{a}^{(j)}\in C_i\right\}} y^{(j)} + (1-\alpha(N_i))\frac{1}{N}\sum_{j=1,\ldots,N} y^{(j)} \tag{3.1}$$

pri čemu je $\alpha : \mathbb{R} \to [0,1]$ funkcija koja u našem razmatranju ovisi isključivo o veličini pojedine kategorije. Idealno, funkcija α poprima zanemarive vrijednosti kada kategorija nije dovoljno zastupljena u uzorku - time funkcija ksv kategoriji dodjeljuje procjenu srednje vrijednosti insolventnosti na temelju cijelog uzorka. Ista idealno poprima vrijednosti zanemarivo manje od jedan kada je kategorija dovoljno zastupljena u uzorku da možemo procjenu pripadne uvjetne stope insolventnosti smatrati statistički kvalitetnom. Motivirani referiranom literaturom, funkciju α definirali smo pravilom pridruživanja

$$\alpha(x) = \left(1 + e^{-\frac{x-a}{b}}\right)^{-1}. \tag{3.2}$$

Praksa je koeficijente a i b u formuli (3.2) tretirati kao hiperparametre te ih također uštimavati algoritmom pretrage mreže hiperparametara, ne bi li se time postigla što bolja performansa modela. Zbog ograničenih resursa, za potrebe ovoga rada smo bili primorani koeficijente izabrati arbitrarno, temeljeći svoje odluke na činjenicama da *i*) slučaj kada je kategorija obuhvaća jedno poduzeće, doprinos njezine stope insolventnosti treba biti jednak doprinosu koje isto poduzeće ima u cijelome uzorku te *ii*) statistička struka uzorak od približno 30 jedinki smatra dovoljno dobrim za donošenje razmjerno kvalitetnih zaključaka o srednjoj vrijednosti uzorka. Stoga su koeficijenti odabrani kao rješenja sustava

$$\begin{cases} \alpha(1) = 1/3658 \\ \alpha(1\%N) = \alpha(36,58) = 1/3. \end{cases}$$

3.3. Imputacija srednjom vrijednošću

Imputacija srednjom vrijednošću transformacija je numeričkih varijabli koje bilježe nedostajuće vrijednosti. Aritmetička sredina realizacija jedna je od (moglo bi se reći i najintuitivnijih) karakterističnih vrijednosti varijable koja bilježi brojne poželjne karakteristike. Vođeni time, metodom imputacije srednjom vrijednošću nedostajuće vrijednosti u skupu realizacija slučajne varijable zamjenjujemo aritmetičkom sredinom ostalih, poznatih realizacija. Formalno, neka je X_i numerička varijabla koja bilježi nedostajuće vrijednosti, jednostavnosti radi označene s $\emptyset$, odnosno skup realizacija varijable X_i dan je s $\mathbb{R}\cup\{\emptyset\}$. Neka je $M = \left\{j : x_i^{(j)} \neq \emptyset\right\}$ indeksni skup poduzeća za koje varijabla X_i ne bilježi nedostajuće vrijednosti, odnosno za koje je vrijednost varijable X_i poznata. Imputacija srednjom vrijednošću transformacija je slučajne varijable X_i

preslikavanjem $isv : \mathbb{R} \cup \{\emptyset\} \to \mathbb{R}$ s pravilom pridruživanja

$$isv\,(x) = \begin{cases} x, & x \neq \emptyset \\ \frac{1}{M} \sum_{k \in M} x_i^{(k)}, & x = \emptyset \end{cases}. \tag{3.3}$$

Protuteža jednostavnosti implementacije i interpretacije spomenute metode dolazi u obliku velike pristranosti koju metoda uvodi u novodobivenu varijablu, u našem slučaju dodatno pogoršanom činjenicom kako nedostajuće vrijednosti *ne nedostaju nasumično* (engl. *missing not at random*) [19]. Međutim, njezino korištenje opravdavamo njezinom intuitivnošću te činjenicom kako broj prediktora koji bilježe nedostajuće vrijednosti u ne smatramo dovoljno velikim da bi rezultirajuća pristranost znatno naštetila kvaliteti zaključaka.

3.4. Sekvencijska selekcija prediktora

Sekvencijska selekcija prediktora pohlepni je algoritam korišten kako bi se polazni skup prediktora sveo na podskup relevantnih prediktora određene veličine, pri selekciji što je moguće manje naštetivši performansi modela. Za potrebe istraživanja korištena je pokretna sekvencijska selekcija prediktora unaprijed (engl. *Sequential Forward Floating Selection*, nadalje SFFS) pri kojoj, počevši od praznog skupa, iterativno dodajemo prediktor koji u najvećoj mjeri poboljšava performansu, uz možebitno isključivanje nekih od prethodno dodanih prediktora, ovisno o njihovu doprinosu kvaliteti modela. Iako selekcija unazad nerijetko rezultira drugačijim skupom relevantnih prediktora, zbog velike vremenske složenosti algoritma i činjenice da smo koristili njegovu pokretnu inačicu, odlučili smo se za potrebe istraživanja usredotočiti isključivo na SFFS. Performansu modela smo mjerili AUC vrijednošću te je bitno istaknuti kako bi izbor drugačije mjere performanse rezultirao drugačijim skupom relevantnih prediktora – dio je to analize koji ostavljamo za možebitno kasnije proširenje istraživanja.

Neka je $\mathcal{M}$ model u kojemu su fiksirane vrijednosti hiperparametara (izuzev skupa prediktora) i neka je $\widetilde{X} \in \mathcal{P}\left(\{X_1, ..., X_D\}\right)$ proizvoljan neprazan podskup skupa prediktora u uzorku. S $AUC(\mathcal{M}; \widetilde{X})$ označimo prosječnu AUC vrijednost peterostruke unakrsne validacije koju postiže model $\mathcal{M}$ kada je isti izgrađen isključivo na temelju prediktora iz skupa $\widetilde{X}$. SFFS algoritam, kojim od inicijalnog skupa prediktora dobivamo n-člani podskup relevantnih prediktora, možemo sročiti ispod navedenim pseudokodom [16].

Pojednostavljeno, nakon što u skup prije odabranih relevantnih prediktora dodamo prediktor koji u najvećoj mjeri poboljšava performansu modela, provjeravamo jesu li njegovim dodavanjem neki od prije dodanih prediktora postali irelevantni te iste isključujemo iz skupa odabranih prediktora. Samim time, SFFS algoritam preferira prediktore koji donose novi skup bitnih informacija o zajmoprimcima što zauzvrat (u teoriji) implicira nisku koreliranost među relevantnim prediktorima. Kada navedenome dodamo i njegovu visoku kvalitetu, zaključujemo kako SFFS predstavlja izvrstan algoritam selekcije prediktora u kontekstu razmatranog uzorka.

SFFS algoritam

Ulaz:	Skup prediktora $X = \{X_1, \ldots, X_D\}$; model $\mathcal{M}$; broj $n < D$
Izlaz:	Skupovi $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \in \mathcal{P}(X)$ pri čemu je $\left\lvert X^{(k)} \right\rvert = k$
Inicijalizacija:	$X^{(0)} \leftarrow \emptyset; k \leftarrow 0$
Terminacija:	Zaustaviti se kada je $k = n$

Korak 1) *(dodavanje)*

$x^+ \leftarrow \underset{x \in X \setminus X^{(k)}}{\operatorname{argmax}} AUC\left(\mathcal{M}; X^{(k)} \cup \{x\}\right)$

$X^{(k+1)} \leftarrow X^{(k)} \cup \{x^+\}$

$k \leftarrow k + 1$

Korak 2) *(isključivanje)*

$x^- \leftarrow \underset{x \in X^{(k)}}{\operatorname{argmax}} AUC\left(\mathcal{M}; X^{(k)} \setminus \{x\}\right)$

ako $AUC\left(\mathcal{M}; X^{(k)} \setminus \{x^-\}\right) > AUC\left(\mathcal{M}; X^{(k-1)}\right)$ **onda**

$X^{(k-1)} \leftarrow X^{(k)} \setminus \{x^-\}$

$k \leftarrow k - 1$

idi na **Korak 2)**

inače

idi na **Korak 1)**

3.5. Klasifikacijsko i regresijsko stablo

Klasifikacijsko i regresijsko stablo [10] [18] [6] neparametarska je metoda nadziranog strojnoga učenja zasnovana na principu particije prostora $\mathbb{R}^D$ na kvadre čiji su bridovi paralelni s koordinatnim osima. Kako je u okvirima rada razmotreno isključivo binarno klasifikacijsko stablo izgrađeno nad skupom numeričkih prediktora (nakon kodiranja kvalitativnih prediktora), u nastavku ćemo predstaviti proces izgradnje, formiranja predikcija i regularizacije isključivo za navedeni podskup CART modela, dok se generalizacije tvrdnji mogu pronaći u referiranoj literaturi. Klasifikacija jedinke reprezentirane odgovarajućom realizacijom vektora prediktora vrši se sustavno, spustom niz strukturu stabla od njegova korijena (čvora koji nema nadređeni čvor, često nazivan i čvor-roditelj) do odgovarajućeg terminalnog čvora, odnosno lista (čvora koji nema nasljednike, često nazivane i čvorove-djecu), pri čemu se na svakome usputnome čvoru jedinka raspoređuje u jedan od dva čvora-nasljednika temeljem realizacije točno jednog prediktora.

Izgradnja binarnog klasifikacijskog stabla. U praksi standardni pristup biparticiji podataka na čvorovima binarnom je kategorizacijom skupa vrijednosti jedne od numeričkih varijabli na skupove u kojima je maksimum jednog skupa manji od minimuma drugog skupa – drugim riječima, jedinki se pridružuje lijevi ili desni čvor-nasljednik ovisno o realizaciji identifikator-funkcije $\mathbb{1}_{\{X_i < \theta\}}$ za numeričku varijablu X_i i graničnu vrijednost $\theta \in \mathbb{R}$. Pojednostavljenja radi, s $\mathbf{a}^{(j)} \in v$ ćemo označavati činjenicu da se zajmoprimac $\mathbf{a}^{(j)}$, počevši od korijena stabla prateći

iznad opisan princip raspoređivanja na čvorove-nasljednike, nakon određenog broja koraka nađe na čvoru v. Nad svakim čvorom v klasifikacijskoga stabla razmatramo uvjetnu vjerojatnost

$$p_{v1} = P\left(Y = 1 \,|\, \mathbf{a} \in v\right) \tag{3.4}$$

da je zajmoprimac loš ako se isti pri spustu niz strukturu stabla nađe na čvoru v. Neka su $n_{v1} = \left|\left\{j : \mathbf{a}^{(j)} \in v,\, y^{(j)} = 1\right\}\right|$ i $n_v = \left|\left\{j : \mathbf{a}^{(j)} \in v\right\}\right|$ redom broj loših zajmoprimaca i broj zajmoprimaca općenito koji se pri spustu niz strukturu stabla nađu na vrhu v. Uz navedenu notaciju, uvjetnu vjerojatnost (3.4) možemo procijeniti s $\hat{p}_{v1} = n_{v1}/n_v$ te analogno definirati $\hat{p}_{v0} = 1 - \hat{p}_{v1}$. Želimo za svaki čvor u stablu kvantificirati kvalitetu odluke, odnosno formulirati mjeru sigurnosti u odluku da je zajmoprimac koji se pri spustu niz strukturu stabla našao na čvoru v dobar ili loš. Intuitivno, sigurnost u odluku time je veća što je $\hat{p}_{v1}$ bliža vrijednostima nula ili jedan dok sigurnost slabi što je $\hat{p}_{v0}$ bliža vrijednosti 0,5 (kada je naša odluka dobra koliko i slučajan odabir bacanjem novčića). Dodatno, s obzirom na binarnost naše odluke i činjenicu kako niti jednu od klasifikacijskih odluka ne preferiramo više od druge, poželjno svojstvo mjere sigurnosti u odluku je da ista poprima jednaku vrijednost kada je $\hat{p}_{v1} = a$ i kada je $\hat{p}_{v1} = a$ za $a \in [0, 1]$. Vođeni time definiramo funkciju (u literaturi nazivanom i mjerom) nečistoće kao funkciju $\Phi : \left\{(\hat{p}_{v0},\, \hat{p}_{v1}) \in [0,1]^2 : \hat{p}_{v0} + \hat{p}_{v1} = 1\right\} \to \mathbb{R}$ za koju vrijedi kako

i) Φ postiže maksimum ako i samo ako vrijedi $\hat{p}_{v0} = \hat{p}_{v1}$,

ii) Φ postiže minimum ako i samo ako vrijedi $\hat{p}_{v0} = 0$ ili $\hat{p}_{v1} = 0$,

iii) Φ je simetrična, tj. vrijedi $\Phi(\hat{p}_{v0},\, \hat{p}_{v1}) = \Phi(\hat{p}_{v1},\, \hat{p}_{v0})$.

Stabla su u okvirima ovoga istraživanja kreirana korištenjem dviju mjera nečistoće ne bismo li usporedili njihov utjecan ja performansu klasifikatora: Gini nečistoće $Gini(v) = 2\hat{p}_{v0}\hat{p}_{v1}$ te entropije[5] $Entropy(v) = -\hat{p}_{v0}\ln(\hat{p}_{v0}) - \hat{p}_{v1}\ln(\hat{p}_{v1})$.

Pretpostavimo kako na čvoru v spustom niz strukturu stabla imamo dovoljan broj jedinki da isti ima smisla granati na čvorove-nasljednike. Neka je $n_i \leq N$ broj različitih realizacija varijable X_i, neka je $\left\{z_i^{(1)}, z_i^{(2)}, \ldots, z_i^{(n_i)}\right\} \subseteq \left\{x_i^{(1)}, \ldots, x_i^{(N)}\right\}$ skup njezinih različitih realizacija te neka bez smanjenja općenitosti vrijedi

$$z_i^{(1)} < z_i^{(2)} < \ldots < z_i^{(n_i)}.$$

Za varijablu X_i definiramo niz graničnih vrijednosti $\theta_i^{(0)}, \theta_i^{(1)}, \ldots, \theta_i^{(n_i)}$ za koji vrijedi

i) $\theta_i^{(0)} = -\infty$ te $\theta_i^{(n_i)} = \infty$,

ii) za $j \in \{1, \ldots, n_i - 1\}$, $\theta_i^{(j)}$ je proizvoljna vrijednost iz (često definirana kao njegova aritmetička sredina) intervala $\left\langle z_i^{(j)},\, z_i^{(j+1)} \right\rangle$.

[5]U definiciji smo prešutno ispustili činjenicu kako se radi o proširenju po neprekidnosti navedene funkcije na segmentu $[0, 1]$ – u suprotnom pravilo pridruživanja nije definirano za $\hat{p}_{v0} = 0$ i $\hat{p}_{v1} = 0$.

Proces ponavljamo za svaki od prediktora te zatim razmotrimo binarnu particiju vrha v s obzirom na indikator-funkciju $\mathbb{1}_{\left\{X_i<\theta_i^{(j)}\right\}}$ za svaki $i \in \{1,\ldots,D\}$ i za svaki $j \in \{0,1,\ldots,n_i\}$.

Neke od polinomijalno rastućeg broja mogućih biparticija su redundantne – njihovom se implementacijom sigurnost u odluku da je zajmoprimac dobar ili loš, kada bi se isti s vrha v spustio na neki od čvorova-nasljednika, ne bi poboljšala, ili bi se u najgorem slučaju pogoršala. Neke biparticije su temeljene na varijablama koje u širem smislu nisu od velikog značaja te bismo njihovom implementacijom, iako eventualno poboljšali mjeru nečistoće čvorova-nasljednika čvora v, onemogućili relevantnijim prediktorima da ostvare svoj potencijal. Stoga za indikator-funkciju na temelju koje ćemo izvršiti biparticiju čvora v uzimamo onu koja maksimizira funkciju dobiti (engl. *gain function*). Funkciju dobiti biparticije čvora v na čvorove-nasljednike v_L i v_R na temelju indikator-funkcije $\mathbb{1}_{\{X<\theta\}}$ s obzirom na mjeru nečistoće Φ definiramo kao

$$Gain(v; X, \theta) = \Phi(v) - \left(\frac{n_L}{n_v}\Phi(v_L) + \frac{n_R}{n_v}\Phi(v_R)\right) \tag{3.5}$$

pri čemu je n_L broj jedinki koji su se na temelju realizacije indikator-funkcije $\mathbb{1}_{\{X<\theta\}}$ s čvora v *spustile* na čvor v_L te $n_R = n - n_L$. Maksimizacijom funkcije $Gain$ smanjujemo težinski prosjek nečistoća novokreiranih čvorova-nasljednika. Proces grananja provodimo sve dok neki od zaustavnih kriterija nije zadovoljen, odnosno dok jedan od regularizacijskih kriterija ne onemogući daljnju izgradnju stabla.

Predikcija binarnim klasifikacijskim stablom. Pretpostavimo kako binarno klasifikacijsko stablo zajmoprimcu $\mathbf{a}^{(j)}$ s pripadnom realizacijom vektora prediktora $\mathbf{x}^{(j)}$ sustavnim spustom niz svoju strukturu dodjeljuje list l te kako su n_{l1} i n_l redom broj loših zajmoprimaca i zajmoprimaca općenito koji su se pri izgradnji modela našli na listu l. Uvjetnu vjerojatnost $P(Y = 1 \,|\, \mathbf{x}^{(j)})$ tada procjenjujemo s n_{l1}/n_l, odnosno udjelom loših zajmoprimaca u svim zajmoprimaca koji su se pri izgradnji modela našli na listu l.

Regularizacija binarnog klasifikacijskog stabla. Ako nisu postavljene restrikcije na strukturu stabla niti minimalnu promjenu u funkciji $Gain$ koju je potrebno ostvariti kako bi došlo do biparticije čvora, stablo može rasti sve dok svaki od njegovih listova ne ostvari minimalnu moguću nečistoću, premda na nekima od njih imali tek jednu jedinku iz trening skupa. Iako će takva struktura stabla ići na ruku njegove performanse nad trening-skupom, ista će biti na uštrb njegove vanjske valjanosti.

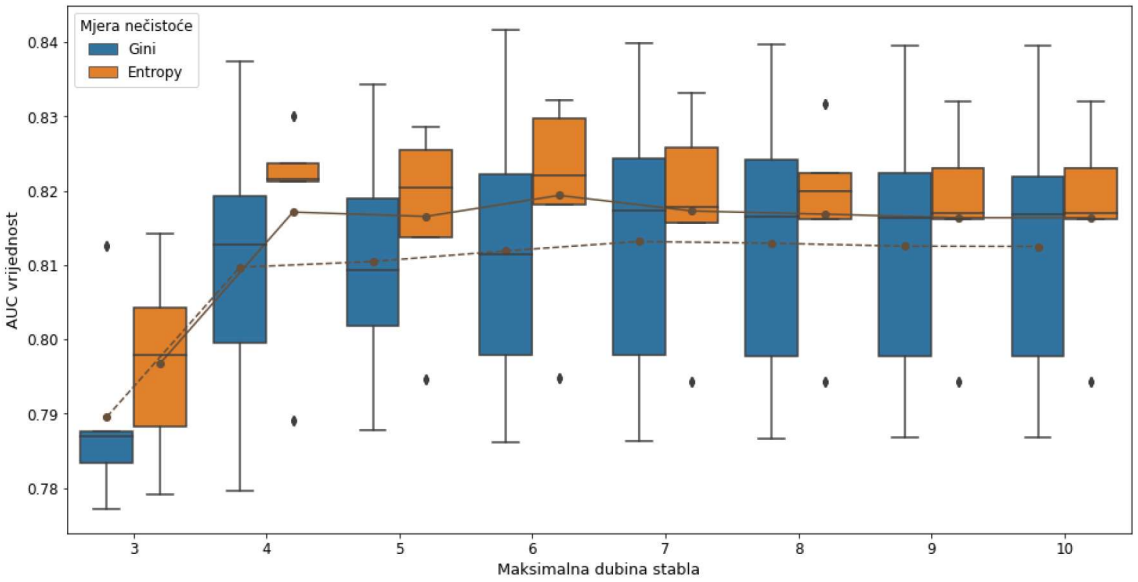
- Kako se predikcije formiraju na temelju udjela loših zajmoprimaca na pojedinom listu stabla, koja uz naše pretpostavke odgovara procjeni srednje vrijednosti Bernoullijeve distribucije, vođeni zakonom velikih brojeva pouzdanost u kvalitetu formiranih predikcija možemo povećati postavljanjem zahtjeva na minimalni broj jedinki koji se pri izgradnju stabla mora naći na listu kako bi isti uopće bio kreiran. Dodatno, ograničenje možemo postaviti i na najmanji broj jedinki koji se pri izgradnji stabla mora naći na pojedinome čvoru prije no razmotrimo njegovu možebitnu biparticiju na čvorove-sljedbenike.

- Kako bi se umanjila (zajamčena) prenaučenost stabla te pojednostavila njegova struktura, izgradnju istoga obogaćujemo algoritmom rezidbe (obrezivanja, podrezivanja). Neka je T_0 brojem čvorova najveće moguće stablo izgrađeno nad trening-podatcima, uvažavajući pri njegovoj izgradnji limitacije postavljene na maksimalnu dubinu i minimalni broj jedinki na listovima/čvorovima prije biparticije. Pretpostavimo kako T_0 ima $t \in \mathbb{N}$ podstabala $T_1, \ldots, T_t$ koji s istim dijele korijen, neka je L_i broj listova u i-tom stablu T_i te neka je $R_i^{(j)}$ realizacija mjere nečistoće na j-tom listu i-tog stabla. Rezidbom odabiremo k-to stablo T_k koje zadovoljava

$$k \in \underset{i=0,1,\ldots,t}{\operatorname{argmin}} \left(\alpha L_i + \sum_{j=1,\ldots,L_i} R_i^{(j)} \right) \tag{3.6}$$

  pri čemu je α proizvoljno odabran pozitivni hiperparametar (u radu nazvan parametar rezidbe). Drugim riječima, rezidbom izabiremo podstablo koje pronalazi balans između broja listova i ukupne nečistoće nad istima, dok odabirom parametra α prilagođavamo koliki naglasak stavljamo na broj listova pa time i dubinu stabla.

Pretraga mreže hiperparametara. Kako je navedeno u *Metodologiji*, potpoglavlju *Binarni klasifikatori i selekcija kandidata*, prije no što smo proveli pretragu mreže hiperparametara sa ciljem izgradnje što je boljeg binarnog klasifikacijskog stabla, analizirali smo utjecaj promjene dubine stabla i parametra rezidbe na njegovu performansu, mjerenu AUC vrijednošću.



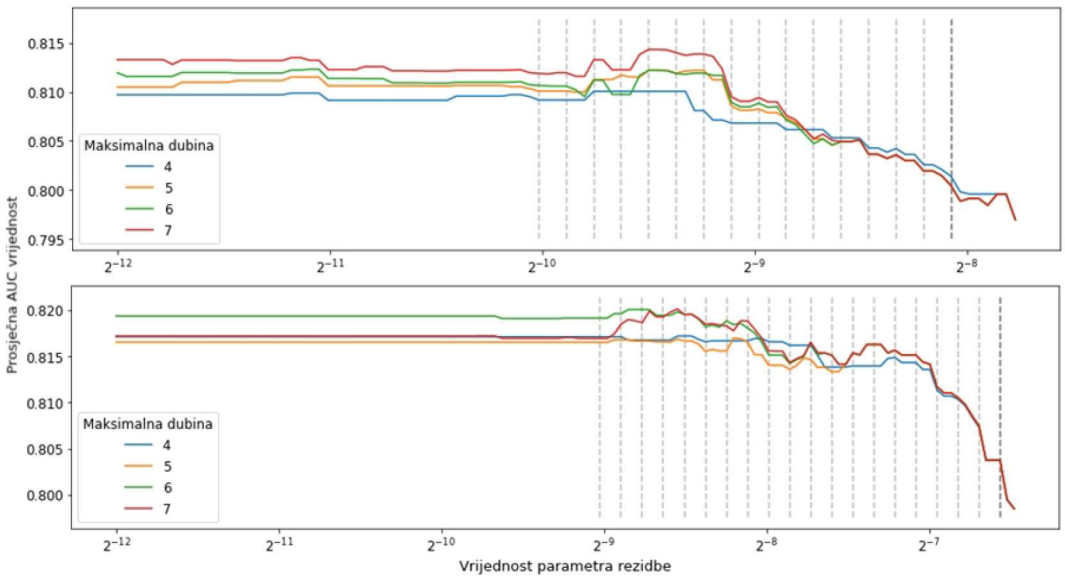
Slika 3.1: Kutijasti dijagrami realizacija AUC vrijednosti CART klasifikatora u ovisnosti o pripadnoj maksimalnoj dubini i mjeri nečistoće upotpunjeni trajektorijama promjena srednjih vrijednosti.
minimalni broj jedinki na listu iznosi 5%; minimalni broj jedinki za razdiobu iznosi 1,875% trening-podskupa

Kreirali smo trideset skupova za validaciju, svaki od njih dobivši slučajnim odabirom 20% jedinki iz trening-skupa. Za svaku od osam različitih vrijednosti hiperparametra maksimalne dubine stabla smo izgradili trideset modela i svaki evaluirali nad korespondentnim validacijskim skupom, prikupivši trideset realizacija AUC vrijednosti. Na *Slici 3.1* su prikazani odgovarajući kutijasti dijagrami, vizualizirani odvojeno za različite mjere nečistoće. Vođeni kretanjem srednjih vrijednosti mjere performanse za različite mjere nečistoće, odlučili smo se pri pretrazi mreže hiperparametara usredotočiti na stabla čija je maksimalna dubina barem četiri, ali ne više od sedam razina.

Analogno analizi utjecaja veličine šume na performansu, trening-skup smo trideset puta podijelili na skup za izgradnju modela i skup za validaciju modela u omjeru 4:1 te smo na temelju istih za svaku od razmotrenih realizacija hiperparametra maksimalne dubine stabla odredili trideset AUC vrijednosti. Inicijalno smo za obje mjere nečistoće analizirali 256 različitih vrijednosti parametra rezidbe

$$\alpha = 2^{-\left(1+\frac{11}{255}i\right)} \quad \text{za} \quad i = 0, 1, \ldots, 255.$$

Na *Slici 3.2* su pak prikazani tek dijelovi trajektorija prosjeka AUC vrijednosti, zaustavljajući se neposredno nakon one vrijednosti parametra rezidbe za koju prosječne AUC vrijednosti za sve četiri maksimalne dubine stabla padaju ispod 80% (označena tamnijom, posljednjom isprekidanom crtom u nizu). Isprekidanim crtama označene su realizacije parametra rezidbe uključene u konačnu pretragu mreže hiperparametara te se razlikuju među mjerama nečistoće. U nastavku su navedeni skupovi realizacija parametra rezidbe označeni s $\mathbf{S}_{Gini}$ i $\mathbf{S}_{Entropy}$.



Slika 3.2: Trajektorije prosjeka AUC vrijednosti CART klasifikatora u ovisnosti o pripadnoj maksimalnoj dubini i realizaciji parametra rezidbe za Gini nečistoću (iznad) i entropy nečistoću (ispod).
minimalni broj jedinki na listu iznosi 5%; minimalni broj jedinki za razdiobu iznosi 1,875% trening-podskupa

U *Tablici 3.1* je navedena mreža hiperparametara na temelju koje smo izabrali najbolje reprezentante CART klasifikatora u okviru ovog istraživanja. Kako konkretne vrijednosti parametra rezidbe ne smatramo od prevelike važnosti, navest ćemo ih u notaciji skupa. Razmotreno je 4608 modela, među kojima je u 2048 modela implementirana Gini nečistoća te u 2560 modela implementirana entropija.

Hiperparametar	**Vrijednosti razmotrene pri pretrazi mreže hiperparametara**	
Mjera nečistoće	Gini	Entropy
Parametar rezidbe	$\mathbf{S}_{Gini}$	$\mathbf{S}_{Entropy}$
Maksimalna dubina stabla	4; 5; 6; 7	
Minimalni broj jedinki za razdiobu	10%n; 8,75%n; 7,5%n; 6,25%n; 5%n; 3,75%n; 2,5%n; 1,25%n	
Minimalni broj jedinki na listu	5%n; 2,5%n; 1,25%n; 0,625%n	

Tablica 3.1: Pretražena mreža hiperparametara za familiju CART klasifikatora.
$n = 2.340$ *je broj jedinki u trening-podskupu*

3.6. Slučajna šuma

Slučajna šuma [18] [6] je neparametarska ansambl metoda strojnoga učenja koja jednostavnost i interpretabilnost CART klasifikatora oplemenjuje bagging (engl. *Bootstrap Aggregation*) algoritmom i ograničenjima postavljenim na proces grananja njome obuhvaćenih klasifikacijskih stabala. Bagging algoritmom izgrađujemo ansambl sačinjen od određenog broja instanci istoga modela treniranog nad različitim skupovima jedinki, obično dobivenih jednostavnim slučajnim uzorkovanjem s ponavljanjem iz inicijalnog trening-skupa. Ključne beneficije koje slijede iz njegove implementacije, neovisno o prirodi metode koju agregiramo, demokratizacija je, odnosno disperzija procesa formiranja predikcija te drastično smanjenje varijance metode koju agregiramo[6]. U nastavku ćemo predstaviti proces izgradnje, formiranja predikcija i regularizacije isključivo za slučajne šume binarnih klasifikacijskih stabala kakve su uspoređene u okvirima istraživanja.

Izgradnja slučajne šume. Za unaprijed odabran broj procjenitelja u šumi, iz trening-podataka jednostavnim slučajnim uzorkovanjem s ponavljanjem izdvajamo trening-podskupove željene veličine te nad istima neovisno jedno o drugome izgrađujemo binarna klasifikacijska stabla. Bitno je istaknuti kako su sva stabla odlučivanja obilježena jednakim vrijednostima strukturnih i regularizacijskih hiperparametara. Kako bi se postigla što veća općenitost uzoraka koje model prepoznaje u podatcima, pri biparticiji svakog od čvorova u pojedinom stablu se maksimalna promjena u *Gain* funkciji utvrđuje nad slučajno odabranom podskupu prediktora

[6]Visoka varijanca modela, svojstvo koje obilježava CART klasifikatore, implicira njegovu nestabilnost s obzirom na odabir trening-skupa – ako iz iste populacije uzorkujemo dva različita trening-skupa jednakih generalnih karakteristika (veličine, omjera dobrih i loših zajmoprimaca, udjela nedostajućih vrijednosti za pojedine varijable i sl.) te nad njima izgradimo CART klasifikator s jednakim hiperparametrima, njihova će performansa, ali i struktura, biti znatno različita. Visoka varijanca povlači nisku vanjsku valjanost.

unaprijed fiksirane veličine, za razliku od cijelog skupa prediktora u slučaju kada smo CART klasifikator proučavali kao zaseban model.

Predikcija slučajnom šumom. Neka slučajnu šumu čini $T \in \mathbb{N}$ stabala. Procjena uvjetne vjerojatnosti $P(Y = 1 \,|\, \mathbf{x}^{(j)})$ algoritmom slučajne šume određuje se kao prosjek procjena iste uvjetne vjerojatnosti pojedinačnim stablima, odnosno kao

$$\hat{p}^{(j)} = \frac{1}{T} \sum_{t=1,\ldots,T} \hat{p}_t^{(j)} \tag{3.7}$$

pri čemu je $\hat{p}_t^{(j)}$ procjena navedene uvjetne vjerojatnosti t-tim stablom šume po u prošlom potpoglavlju opisanome principu.

Regularizacija slučajne šume. Za razliku od nekih drugih ansambl metoda, među kojima je i kasnije predstavljeni XGBoost, slučajna šuma klasifikacijskih stabala nije pod velikim rizikom prenaučenosti – međutim, promjene u strukturnim i bagging parametrima uz dovoljno velik broj stabala u šumi može popraviti performansu modela nad pojedinim (prije možda i neprimijećenim) uzorcima u podatcima te popraviti njezinu performansu nad test-podatcima.

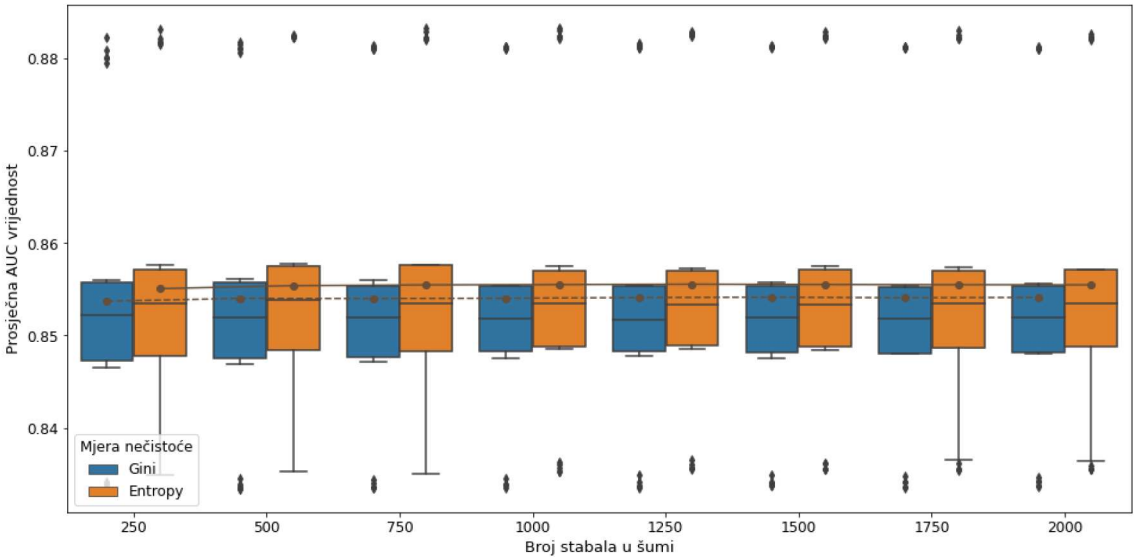
- Zbog vremenske složenosti kojom bi rezultiralo uključivanje parametra rezidbe u algoritam pretrage mreže hiperparametra, strukturu stabala smo ograničili isključivo minimalni brojem jedinki po listu, minimalni brojem jedinki potrebnim za biparticiju čvora te maksimalnom dubinom stabala.

- Kako je navedeno prije, bagging algoritam zahtjeva generiranje jednostavnim slučajnim uzorkovanjem generiranih podskupova trening skupa. Proučit ćemo kako različite veličine trening skupa interferiraju s različitim veličinama šume.

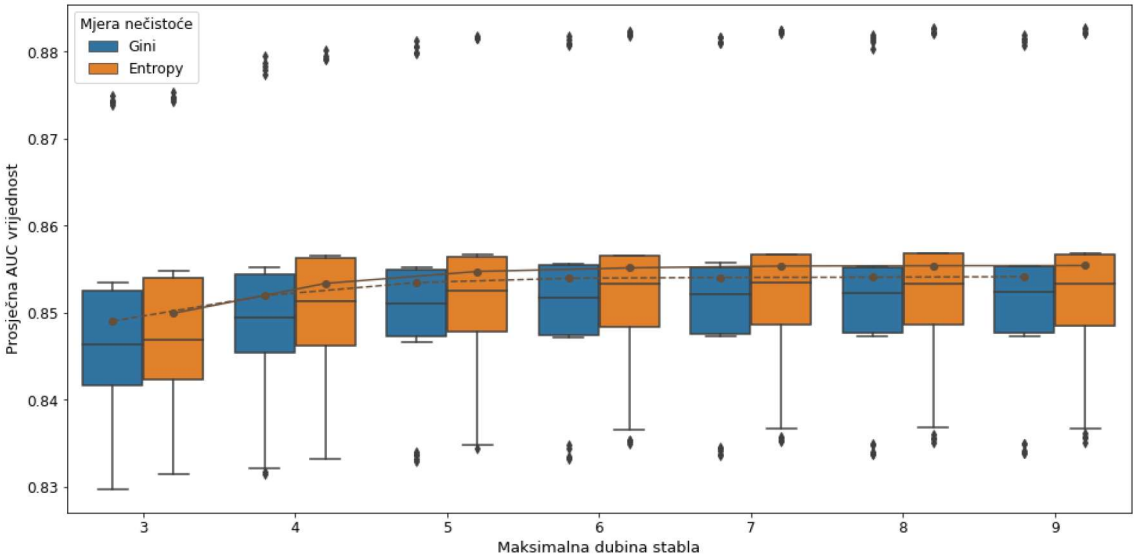
- Za broj prediktora nad kojima ćemo razmatrati pojedinu biparticiju čvorova, kako je standard u praksi, uzet ćemo D, $\sqrt{D}$ ili $\log_2 D$ pri čemu je D broj prediktora u uzorku.

Pretraga mreže hiperparametara. Čak i ako šumu izgradimo širu no što je to potrebno, s obzirom na to da predikcije formiramo kao prosjek, njezina performansa nad test-podatcima neće biti narušena. Međutim, isti fenomen možemo sagledati i iz drugoga kuta – dodavanjem stabala u ansambl nakon određene širine neće popraviti performansu modela nad test-podatcima, ali će povećati vremensku složenost izgradnje modela, čak i nad manjim skupovima podataka. Stoga smo, kako je navedeno u *Metodologiji*, potpoglavlju *Binarni klasifikatori i selekcija kandidata*, prije no što smo proveli pretragu mreže hiperparametara ne bismo li pronašli što je bolji RF klasifikator, analizirali utjecaj promjene širine šume i dubine njome obuhvaćenih stabala na njegovu performansu, mjerenu AUC vrijednošću. Kreirali smo trideset skupova za validaciju, svaki od njih dobivši slučajnim odabirom 20% jedinki iz trening-skupa. Za svaku od osam razmotrenih veličina šume smo izgradili trideset modela i svaki evaluirali nad korespondentnim

validacijskim skupom, prikupivši trideset realizacija AUC vrijednosti. Na *Slici 3.3* su prikazani odgovarajući kutijasti dijagrami, vizualizirani odvojeno za različite mjere nečistoće. Kako rast šume ne povlači značajnu promjenu u prosjeku AUC vrijednosti nakon 750 stabala, odlučili smo razmatranja u nastavku provesti nad ansamblima sačinjenim od 500, 750 i 1000 stabala.



Slika 3.3: Kutijasti dijagrami realizacija AUC vrijednosti RF klasifikatora u ovisnosti o veličini šume i mjeri nečistoće upotpunjeni trajektorijama srednjih vrijednosti.
minimalni broj jedinki na listu iznosi 5% trening-podskupa; minimalni broj jedinki za razdiobu iznosi 1,875% trening-podskupa; maksimalna dubina stabala iznosi 7; bagging parametar iznosi 70%



Slika 3.4: Kutijasti dijagrami realizacija AUC vrijednosti RF klasifikatora u ovisnosti o maksimalnim dubinama stabala i mjeri nečistoće upotpunjeni trajektorijama srednjih vrijednosti.
minimalni broj jedinki na listu iznosi 5% trening-podskupa; minimalni broj jedinki za razdiobu iznosi 1,875% trening-podskupa; veličina šume iznosi 750 stabala; bagging parametar iznosi 0,7

Analogno analizi utjecaja veličine šume na performansu, trening-skup trideset puta podijelili na skup za izgradnju modela i skup za validaciju modela u omjeru 4:1 te smo na temelju istih za svaku od razmotrenih realizacija hiperparametra maksimalne dubine stabla odredili trideset AUC vrijednosti. Na *Slici 3.4* su prikazani odgovarajući kutijasti dijagrami, vizualizirani odvojeno za različite mjere nečistoće. Kako poboljšanje prosječne AUC vrijednosti za obje mjere nečistoće stagnira nakon restrikcije maksimalne dubine pojedinačnih stabala na šest razina te kako bismo olakšali usporedbu performanse RF i CART metode, pri pretrazi mreže hiperparametara smo se usredotočili na maksimalnu dubinu od barem četiri, ali ne više od sedam razina.

U *Tablici 3.2* je navedena mreža hiperparametara na temelju koje smo izabrali najbolje reprezentante RF klasifikatora u okviru ovog istraživanja. Razmotreno je 11.520 modela.

Hiperparametar	**Vrijednosti razmotrene pri pretrazi mreže hiperparametara**
Mjera nečistoće	Gini; Entropy
Broj stabala u šumi	500; 750; 1000
Maksimalna dubina stabala	4; 5; 6; 7
Minimalni broj jedinki za razdiobu	$10\%m$; $8{,}75\%m$; $7{,}5\%m$; $6{,}25\%m$; $5\%m$; $3{,}75\%m$; $2{,}5\%m$; $1{,}25\%m$
Minimalni broj jedinki na listu	$5\%m$; $2{,}5\%m$; $1{,}25\%m$; $0{,}625\%m$
Bagging parametar	$0{,}5n$; $0{,}6n$; $0{,}7n$; $0{,}8n$; $0{,}9n$
Broj prediktora po razdiobi	D; $\sqrt{D}$; $\log_2 D$

Tablica 3.2: Pretražena mreža hiperparametara za familiju RF klasifikatora.
$n = 2.340$ je broj jedinki u trening-podskupu, m je broj jedinki korišten pri izgradnji pojedinačnih stabala u sklopu bagging algoritma, $D = 55$ je broj prediktora u uzorku

3.7. Ekstremno podizanje gradijenta

XGBoost [13][27], slično metodi slučajne šume, neparametarska je ansambl metoda strojnog učenja čije bazne modele čine stabla s jednakim vrijednostima strukturnih i regularizacijskih hiperparametara. XGBoost algoritam idejno slijedi temeljne principe algoritma podizanja gradijenata (engl. *gradient boosting*) stabala. Konkretno, za razliku od modela slučajne šume u kojima je i-to stablo izgrađeno na temelju realizacija $(\mathbb{X}, Y)$, algoritmom podizanja gradijenata stabala ansambl gradimo sekvencijski, i-to stablo trenirajući na temelju realizacija $(\mathbb{X}, R_{i-1})$, pri čemu je R_{i-1} varijabla kojom kvantificiramo greške predikcija $(i-1)$-og stabla [18]. Kako je pri provođenju istraživanja korišteno isključivo ekstremno podizanje gradijenata klasifikacijskih stabala, nećemo se posvetiti implementaciji ostalih istoga za ostale pojedinačne metoda.

Izgradnja XGBoost modela. Pretpostavimo kako gradimo ansambl od $T \in \mathbb{N}$ stabala. Konvencionalnosti radi, klasifikacijsko binarno stablo ćemo predstavljati funkcijom $f : \mathbb{R}^D \to \mathbb{R}$ s pravilom pridruživanja $f(\mathbf{x}^{(j)}) = \omega_{q(\mathbf{x}^{(j)})}$ pri čemu su $\omega_1, \ldots, \omega_{L_f}$ skor-vrijednosti (nadalje težine) na njegovim listovima (kojih je L_f) te pri čemu je $q : \mathbb{R}^D \to \{1, \ldots, L_f\}$ funkcija koja pojedinoj realizaciji vektora prediktora spustom niz strukturu stabla dodjeljuje odgovarajući list. Drugim riječima, stabla predstavljamo kao preslikavanja koja jedinkama dodjeljuju vrijednosti

spuštanjem na pripadajući list. Težine su ovdje definirane ad hoc, njihovo će značenje postati jasnije nešto kasnije. Predikciju[7] ansambla želimo određivati kao sumu predikcija pojedinog njime obuhvaćenog stabla, odnosno kao

$$\hat{y}^{(j)} = \sum_{t=1,\ldots,T} f_t(\mathbf{x}^{(j)}). \tag{3.8}$$

Ne bismo li postigli što bolju kvalitetu prediktora, ansambl određujemo kao onu kombinaciju stabala koja minimizira *funkciju troška*[8]. Intuitivno, funkcija troška ansambla agregira realizacije funkcije kojom kvantificiramo grešku predikcija ansambla nad trening-skupom (koju nazivamo funkcijom gubitka) s funkciju kojom kvantificiramo složenost ansambla (koju nazivamo regularizacijskom funkcijom). Konkretno, funkcija troška XGBoost klasifikatora dana je sa

$$obj = \sum_{i=1,\ldots,N} L\left(y^{(i)}, \hat{y}^{(i)}\right) + \sum_{k=1,\ldots,k} \Omega(f_k). \tag{3.9}$$

Funkcija gubitka korištena u istraživanju je unakrsna entropija, odnosno log-gubitak (engl. *log-loss*) čiji je jednovarijabilni analogon korišten kao mjera nečistoće pri izgradnji klasifikacijskoga stabla. Funkcija (3.9) time, u okvirima ovoga istraživanja, poprima oblik

$$obj = \sum_{k=1,\ldots,k} \Omega(f_k) - \frac{1}{N} \sum_{i=1,\ldots,N} \left[y^{(i)} \ln(p_i) + (1 - y^{(i)}) \ln(1 - p_i)\right], \; p_i = \left(1 + e^{-\hat{y}^{(i)}}\right)^{-1}. \tag{3.10}$$

Iako postavljamo uvjet na diferencijabilnost funkcije troška, minimum izraza ovisi kako o predikcijama pojedinih stabala, tako i o njihovim strukturama te isti ne možemo odrediti standardnim minimizacijskim metodama u Euklidskome prostoru. Ansambl zato gradimo aditivno. Neka je $\hat{y}_t^{(j)}$ predikcija varijable cilja za j-tu jedinku nakon dodavanja t-tog stabla u ansambl. Uz $\hat{y}_0^{(j)} = 0$, iz jednakosti (3.8) slijedi kako za $t = 1, \ldots, T$ vrijedi $\hat{y}_t^{(j)} = \hat{y}_{t-1}^{(j)} + f_t(\mathbf{x}^{(j)})$. Stablo f_t, koje dodajemo u t-tom koraku, određujemo kao stablo koje minimizira funkciju troška (3.9) koja sada poprima oblik

$$obj^{(t)} = \sum_{i=1,\ldots,N} L\left(y^{(i)}, \hat{y}_{t-1}^{(i)} + f_t\left(\mathbf{x}^{(i)}\right)\right) + \sum_{k=1,\ldots,k} \Omega(f_k). \tag{3.11}$$

[7]Iako je u izvornome članku vrijednost $\hat{y}^{(j)}$ nazvana predikcijom te iako je naziv statistički opravdan kada stablom procjenjujemo neprekidnu vrijednost uz MSE funkciju greške, u slučaju klasifikacije s log-loss funkcijom greške vrijednost $\hat{y}^{(j)}$ valja transformirati sigmoidalnom funkcijom ne bismo li dobili procjenu vjerojatnosti insolventnosti.

[8]Funkcija troška je termin posuđen iz sfere dubokoga učenja. Opisana funkcija u izvornome članku nazvana je funkcijom cilja (engl. *objective function*), ali kako bismo olakšali njihovu usporedbu, odlučili smo pri raščlambi XGBoost i ANN algoritma koristiti istu nomenklaturu.

S obzirom na to kako su strukture prije dodanih stabala poznate, vrijednost $\sum_{k=1,\dots,t-1}\Omega(f_k)$ možemo smatrati konstantom koja ne utječe na problem optimizacije funkcije (3.11). Ako s $\mathcal{F}$ označimo klasu svih klasifikacijskih stabala čija struktura zadovoljava određene, unaprijed definirane restrikcije, stablo f_t možemo odrediti kao

$$f_t \in \underset{f\in\mathcal{F}}{\operatorname{argmin}} \left(\Omega(f) + \sum_{i=1,\dots,N} L\left(y^{(i)}, \hat{y}_{t-1}^{(i)} + f\left(\mathbf{x}^{(i)}\right)\right) \right). \tag{3.12}$$

Zbog aditivne prirode algoritma, funkcija gubitka u t-tom koraku funkcija je jedne realne varijable $f\left(\mathbf{x}^{(j)}\right)$ te ju možemo aproksimirati Taylorovim polinomom drugog stupnja oko točke $\hat{y}_{t-1}^{(i)}$. Time minimizacijski problem (3.12) svodimo na

$$\underset{f\in\mathcal{F}}{\operatorname{argmin}} \left(\Omega(f) + \sum_{i=1,\dots,N} \left[g_i f\left(\mathbf{x}^{(i)}\right) + \frac{1}{2} h_i f^2\left(\mathbf{x}^{(i)}\right) \right] \right) \tag{3.13}$$

pri čemu g_i i h_i predstavljaju odgovarajuće derivacije funkcije gubitka s obzirom na varijablu $\hat{y}_{t-1}^{(i)}$. Stabla smo regularizirali L2 regularizacijom

$$\Omega(f) = \frac{1}{2}\lambda \sum_{i=1,\dots,L_f} \omega_i^2, \tag{3.14}$$

odnosno složenost stabla kvantificirali smo veličinom l_2 norme vektora težina na listovima stabala, intuitivno preferirajući stabla s manjim težinama na listovima (i)li manjim brojem listova. Ako u minimizacijski problem (3.13) uvedemo pravilo pridruživanja regularizacijske funkcije (3.14) te za $k = 1, \dots, L_f$ definiramo indeksne skupove $I_k = \left\{ j : q(\mathbf{x}^{(j)}) = k \right\}$ zajmoprimaca kojima je dodijeljen k-ti list stabla f, stablo f_t možemo odrediti kao

$$f_t \in \underset{f\in\mathcal{F}}{\operatorname{argmin}} \sum_{k=1,\dots,L_f} \left[G_{I_k}\omega_k + \frac{1}{2}(\lambda + H_{I_k})\omega_k^2 \right] \tag{3.15}$$

pri čemu definiramo

$$G_{I_k} \coloneqq \sum_{i\in I_k} g_i, \quad H_{I_k} \coloneqq \sum_{i\in I_k} h_i.$$

Uz fiksiranu strukturu stabla f predstavljenu funkcijom q, lako se može utvrditi kako je optimalna vrijednost funkcije troška dana s

$$obj^{(t)\star} = -\frac{1}{2} \sum_{k=1,\dots,L_f} \frac{G_{I_k}^2}{\lambda + H_{I_k}} \tag{3.16}$$

Kako je gotovo pa nemoguće međusobno usporediti sve strukture stabla f_t te na temelju realizacije vrijednosti (3.16) izabrati najbolju među njima, pohlepnim algoritmom gradimo binarno stablo koristeći (3.16) kao analogon mjere nečistoće u izgradnji CART klasifikatora. Konkretno, neka je I indeksni skup jedinki koje su se izgradnjom t-tog stabla našle na čvoru v za kojeg proučavamo možebitnu biparticiju s obzirom na prediktor X na čvorove-nasljednike s korespondentnim indeksnim skupovima jedinki I_L te I_R. Funkciju dobiti biparticije čvora v na čvorove-nasljednike na temelju indikator-funkcije $\mathbb{1}_{\{X<\theta\}}$ tada možemo definirati s

$$Gain(v; X, \theta) = \frac{1}{2}\left[\frac{G_{I_L}^2}{\lambda + H_{I_L}} + \frac{G_{I_R}^2}{\lambda + H_{I_R}} - \frac{G_I^2}{\lambda + H_I}\right] \tag{3.17}$$

Biparticiju čvora temeljimo na prediktoru i graničnoj vrijednosti koji maksimiziraju funkciju (3.17). Nakon što je maksimalan broj biparticija čvorova izvršen, algoritam vrši njegovu rezidbu na temelju pozitivnosti funkcije $Gain$, počevši od listova te završivši na korijenu. Za tako dobiveno stablo f_t, težine na listovima skaliraju se stopom učenja $\eta < 1$ kako bi se *usporilo učenje* te izbjegla prenaučenost modela.

XGBoost algoritam ne zahtijeva imputaciju nedostajućih vrijednosti u varijablama. Recimo kako prediktor X bilježi nedostajuće vrijednosti te neka je r_M stopa insolventnosti jedinki za koje realizacija prediktora nije poznata. Pri grananju čvora na čvorove-nasljednike razmatramo samo poznate realizacije prediktora X te, nakon što je biparticija izvršena, za skup jedinki na svakom čvoru-nasljedniku utvrđujemo stopu insolventnosti, označene s r_L i r_R. Jedinke s nedostajućim vrijednostima spustit ćemo na list-nasljednik sa sličnijom vrijednošću stope insolventnosti, odnosno svrstavamo ih na list-nasljednik ovisno o tome koja od vrijednosti $|r_L - r_M|$ i $|r_R - r_M|$ bilježi manju vrijednost.

Predikcija XGBoost modelom. Kako model uči vrijednosti $\hat{y}^{(j)}$ minimiziranjem (3.10), vjerojatnost $P(Y = 1 \,|\, \mathbf{x}^{(j)})$ procjenjujemo s

$$\hat{p}^{(j)} = \left(1 + e^{-\hat{y}^{(j)}}\right)^{-1}$$

pri čemu se $\hat{y}^{(j)}$ određuje po formuli (3.8).

Regularizacija XGBoost klasifikatora. Ako zamjeni stohastičke prirode izgradnje slučajne šume sekvencijskom izgradnjom ansambla koja modelima dopušta da, više doslovno nego figurativno, uče na vlastitim greškama možemo pripisati izvrsnu performansu i sposobnost prepoznavanja uzoraka u podatcima, istoj moramo pripisati i sklonost XGBoost ansambla izrazitoj prenaučenosti. Regularizacija je stoga neophodna ukoliko želimo postići prihvatljivu vanjsku valjanost.

- Broj stabala u ansamblu jedan je od hiperparametara s najvećim utjecajem na prenaučenost modela. Problem u njegovom zadavanju proizlazi iz činjenice kako je optimalan broj stabala aposteriorna spoznaja, ovisna o konfiguraciji svih ostalih regularizacijskih hi-

perparametara. Stoga broj stabala u XGBoost ansamblu ne određujemo unaprijed, već konzervativno koristimo metodu ranog zaustavljanja. Nakon dodavanja svakog stabla u ansambl, evaluiramo performansu klasifikatora nad validacijskim skupom podataka, mjerenu AUC vrijednošću. Ako se performansa modela nad validacijskim skupom ne poboljša kroz sljedećih 50 iteracija algoritma, dodavanje stabala u ansambl se prekida te se pri predikciji koristi restringirana verzija prediktora koja je postigla najbolju performansu. Bitno je napomenuti kako bi odabir drugih mjera performanse rezultirao drugačijim ansamblima za istu konfiguraciju hiperparametara. Jedini očiti manjak primjene metode ranog zaustavljanja je kako ćemo ionako malen trening-skup morati dodatno smanjiti za 20% kako bismo kreirali validacijski skup.

- Analogno slučajnim šumama, pri dodavanju novog stabla u ansambl smanjit ćemo dimenzionalnost trening-skupa nad kojim je isto izgrađeno. Konkretno, prije izgradnje stabla slučajnim uzorkovanjem kreiramo podskup trening-skupa unaprijed određene veličine nad kojim se stablo trenira. Osim jedinki, pri grananju svakoga čvora razmatramo tek dio inicijalnog skupa prediktora. Time stablima omogućujemo uočavanje uzoraka koji bi inače ostali nezamijećeni.

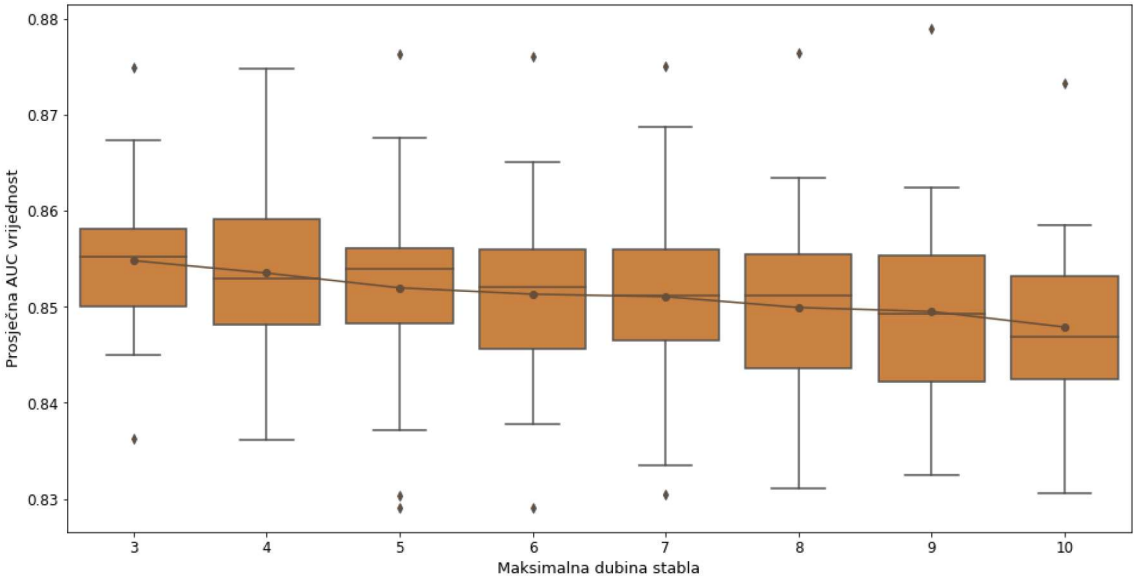
- Svako pojedino stablo može grananjem postići unaprijed odabranu maksimalnu dubinu dok se velike vrijednosti na listovima obeshrabruju korištenjem L2 regularizacije.

- Kako bismo otežali *prebrzo pronalaženje* optimalnog ansambla za klasifikaciju trening-podataka, regulirat ćemo vrijednosti parametra stope učenja.

Pretraga mreže hiperparametara. Kako je navedeno u *Metodologiji*, potpoglavlju *Binarni klasifikatori i selekcija kandidata*, prije no što smo proveli pretragu mreže hiperparametara sa ciljem izgradnje što je boljeg XGBoost klasifikatora, analizirali smo utjecaj promjene maksimalne dubine stabla i na njihovu performansu, mjerenu AUC vrijednošću.

Kreirali smo trideset skupova za validaciju, svaki od njih dobivši slučajnim odabirom 20% jedinki iz trening-skupa. Za svaku od osam različitih vrijednosti hiperparametra maksimalne dubine stabla smo izgradili trideset modela i svaki evaluirali nad korespondentnim validacijskim skupom, prikupivši trideset realizacija AUC vrijednosti. Na *Slici 3.5* su prikazani odgovarajući kutijasti dijagrami, vizualizirani odvojeno za različite mjere nečistoće. Može se uočiti jasan trend opadanja prosjeka AUC vrijednosti s povećanjem maksimalne dubine stabala u ansamblu. Pri pretrazi mreže hiperparametara smo se stoga usredotočili na maksimalnu dubinu od barem tri, ali ne više od šest razina.

U *Tablici 3.3* je navedena mreža hiperparametara na temelju koje smo izabrali najbolje reprezentante XGBoost klasifikatora u okviru ovog istraživanja. Razmotreno je 4.500 modela.



Slika 3.5: Kutijasti dijagrami realizacija AUC vrijednosti XGBoost klasifikatora u ovisnosti o maksimalnim dubinama stabala upotpunjeni trajektorijama srednje vrijednosti.
stopa učenja iznosi 0,05; stopa L2 regularizacije iznosi 0,001; bagging parametar iznosi 0,7; 70% prediktora razmotreno je pri razdiobi

Hiperparametar	**Vrijednosti razmotrene pri pretrazi mreže hiperparametara**
Maksimalna dubina stabala	3; 4; 5; 6
Stopa učenja	0,005; 0,01; 0,05; 0,1; 0,3
L2 regularizacija	0,1‰; 0,25‰; 0,5‰; 0,75‰; 1‰; 2,5‰; 5‰; 7,5‰; 10‰
Bagging parametar	0,5n; 0,6n; 0,7n; 0,8n; 0,9n
Broj prediktora po razdiobi	0,5D; 0,6D; 0,7D; 0,8D; 0,9D

Tablica 3.3: Pretražena mreža hiperparametara za familiju XGBoost klasifikatora.
$n = 2.340$ je broj jedinki u trening-podskupu, $D = 55$ je broj prediktora u uzorku

3.8. Višeslojni perceptron

Višeslojni perceptron [11] [7] [8], poznat i kao unaprijedna neuronska mreža (engl. *feedforward neural network*), osnovica je familije neuronskih mreža iz koje su potekli gotovo svi složeniji ANN algoritmi. Iako u usporedbi s najsuvremenijim ANN metodama primitivna, njezina su snaga i široka primjenjivost neosporive, o čemu svjedoči i činjenica kako su sve u potpoglavlju *Sustavni pregledi literature* razmotrene usporedbe metoda korištenih u upravljanju potrošačkim kreditnim rizikom ukazale na njezinu izvrsnu performansu i robusnost. Osmišljen kako bi simulirao neurone u životinja, perceptron se suštinski svodi na sustavnu jednosmjernu razmjenu signala među čvorovima (neuronima), počevši od ulaza (podražaja), na posljetku formirajući izlaznu vrijednost (reakciju). Za razliku od prethodne tri ML metode koje su detaljno opisane, pregled načina na koji se perceptron izgrađuje i na koji formira predikcije kao i pojedinih regularizacijskih metoda izuzetno je pojednostavljen – njihova matematička složenost nadilazi

volumen ovoga rada.

Istaknimo kako smo pri treningu perceptrona i formiranju predikcija radili s dvodimenzionalnim ekvivalentom ciljane varijable Y definiranim s $\mathbb{Y} = (1 - Y, Y) = \left(\mathbb{1}_{\{Y=0\}}, \mathbb{1}_{\{Y=1\}}\right)$ pri čemu ćemo realizaciju navedene transformacije varijable cilja za j-tu jedinku označavati s $\mathbf{y}^{(j)} = \left(y_0^{(j)}, y_1^{(j)}\right)$.

Struktura višeslojnog perceptrona. Struktura perceptrona, kao i ANN modela općenito, je slojevita – razlika je u tome što perceptron dopušta definiciju preko svoje slojevitosti, konkretno kao slijed od barem dva gusta sloja[9] (engl. *dense layers*). Pojednostavljeno, gusti slojevi su slojevi čiji čvorovi komuniciraju s čvorovima iz prethodnog sloja, odnosno primaju vrijednosti iz prethodnoga sloja, po principu svaki sa svakim – svaki čvor gustoga sloja prima vrijednosti iz svakog čvora prethodnoga sloja. Slojevi jednodimenzionalnog perceptrona, kakav je razmotren pri ovome istraživanju, predstavljaju vektore realnih vrijednosti dok čvorovi predstavljaju pojedine elemente istih vektora. Veze među slojevima, odnosno čvorovima razmatranog sloja i sloja-sljedbenika predstavljaju (skalirani) doprinos pojedinog čvora iz razmatranog sloja linearnoj kombinaciji vrijednosti svih čvorova razmatranog sloja izračunatoj zasebno nad svakim čvorom u sloju-sljedbeniku. Prvi sloj u slijedu slojeva naziva se ulazni sloj te isti prima vektor numeričkih podataka, u našem slučaju realizaciju vektora prediktora $\mathbb{X}$, ne bi li ga proslijedio sljedećem sloju. Broj čvorova u ulaznome sloju stoga ovisi o broju prediktora na kojima izgrađujemo model. Posljednji sloj u slijedu gusti je sloj koji nazivamo izlaznim slojem te isti u našem slučaju kreira izlazne vrijednosti koje odgovaraju procjenama vjerojatnosti da je konkretni zajmoprimac dobar i loš. Kako procjenjujemo dvije vrijednosti, izlazni sloj čine dva čvora. Ostali slojevi u perceptronu nazivaju se skriveni slojevi (engl. *hidden layers*) te je njihova primarna svrha primjena nelinearnih transformacija nad vrijednostima koje su došle iz prijašnjih slojeva kako bi se sa što većom kvalitetom prepoznavali uzorci u podatcima. Pri pretrazi mreže hiperparametara smo razmotrili isključivo perceptrone s jednim ili s dva skrivena sloja, pri čemu su u kasnije navedenom slučaju oba skrivena sloja imala isti broj čvorova. Prije no što djelujemo nelinearnom transformacijom na svaki čvor skrivenog sloja, nad istim se izračuna linearna kombinacija svih vrijednosti iz prošloga sloja s koeficijentima koje zovemo težinama (matrice $\mathbf{w}$ u konkretizaciji navedenoj ispod) te se tako dobivenoj vrijednosti dodaje parametar pristranosti (vektori $\mathbf{b}$ u konkretizaciji navedenoj ispod). Nelinearne transformacije koje se primjenjuju nad gustim slojevima nazivaju se aktivacijskim funkcijama. Aktivacijska funkcija na izlaznome sloju je funkcija *softmax* σ_{max}. Kako izlazni sloj interpretiramo kao dvodimenzionalni vektor, funkcija $\sigma_{max} : \mathbb{R}^2 \to (0,1)^2$ dana je pravilom pridruživanja

$$\sigma_{max}(x_1, x_2) = \left(\frac{e^{x_1}}{e^{x_1} + e^{x_2}}, \frac{e^{x_2}}{e^{x_1} + e^{x_2}}\right).$$

[9]Formalno, višeslojni perceptron je neuronska mreža s barem jednim skrivenim, gusto povezanim slojem. Međutim, kako bismo izbjegli uvođenje pojma skrivenog sloja prije no što je isti definiran, dali smo ekvivalentnu definiciju (ekvivalentna je pod pretpostavkom kako ulazni sloj nije gust).

Pri istraživanju razmotrene aktivacijske funkcije na skrivenim slojevima su funkcija $ReLU : \mathbb{R}^n \to \mathbb{R}^n$(engl. *Rectified Linear Unit function*) te funkcija tangens hiperbolni $tanh : \mathbb{R}^n \to \mathbb{R}^n$ s pravilima pridruživanja

$$ReLU\,(x_1, x_2, \ldots, x_n) = (\max\{0, x_1\}, \max\{0, x_2\}, \ldots, \max\{0, x_n\})$$

te

$$tanh\,(x_1, x_2, \ldots, x_n) = \left(\frac{e^{x_1} - e^{-x_1}}{e^{x_1} + e^{-x_1}}, \frac{e^{x_2} - e^{-x_2}}{e^{x_2} + e^{-x_2}}, \ldots, \frac{e^{x_n} - e^{-x_n}}{e^{x_n} + e^{-x_n}}\right).$$

Konkretno, pretpostavimo kako imamo višeslojni perceptron sa H skrivenih slojeva (dakle, $H+2$ slojeva općenito[10]), kako skriveni slojevi redom imaju $h_1, \ldots, h_H$ čvorova te kako su odgovarajuće aktivacijske funkcije redom dane s $f_1, \ldots, f_H$. Jednostavnosti radi, realizaciju vektora prediktora za j-tog zajmoprimca označimo sa $\mathbf{z}_0$ te njegovu dimenziju označimo s h_0. Tada slučajni vektor $\mathbf{y}^{(j)} = \left(P(Y = 0 \,|\, \mathbf{x}^{(j)}), P(Y = 1 \,|\, \mathbf{x}^{(j)})\right)$ procjenjujemo sljedećom iteracijom:

- za $i = 1, \ldots, H$ definiramo $\mathbf{z}_i = f_i\left(\mathbf{w}_i \mathbf{z}_{i-1} + \mathbf{b}_i\right)$ pri čemu je $\mathbf{w}_i \in \mathbb{R}^{h_i \times h_{i-1}}$ matrica težina (u literaturi poznata i kao jezgra, engl. *kernel*) i $\mathbf{b}_i \in \mathbb{R}^{h_i}$ vektor pristranosti,
- određujemo $\hat{\mathbf{y}}^{(j)} = \sigma_{max}\left(\mathbf{w}_{fin} \mathbf{z}_H + \mathbf{b}_{fin}\right)$ uz $\mathbf{w}_{fin} \in \mathbb{R}^{2 \times h_H}$ i $\mathbf{b}_{fin} \in \mathbb{R}^2$

Jednakost ćemo sažeto zapisati kao $\hat{\mathbf{y}}^{(j)} = f\left(\mathbf{x}^{(j)}; \boldsymbol{\Theta}\right)$ pri čemu je $\boldsymbol{\Theta}$ vektor parametara obuhvaćenih svim matricama težina i svim vektorima pristranosti.

Izgradnja perceptrona. Izgradnja perceptrona može se svesti na uštimavanje vektora parametara Θ iz gornje formule, odnosno iterativno prilagođavanje njegovih vrijednosti primjenom (jedne od unaprijeđenih inačica) algoritma minibatch stohastičkog gradijentnog spusta za zadanu (gotovo svuda) diferencijabilnu funkciju troška, sve dok unaprijed definiran uvjet nije zadovoljen. Optimalnu konfiguraciju vektora parametara $\boldsymbol{\Theta}$ tada teoretski možemo odrediti kao

$$\underset{\theta}{\operatorname{argmin}}\, \widetilde{L}(\theta) = \underset{\theta}{\operatorname{argmin}} \left(\Omega(\theta) + \sum_{i=1,\ldots,N} L\left(\mathbf{y}^{(i)}, f\left(\mathbf{x}^{(i)}; \theta\right)\right)\right). \tag{3.18}$$

U okviru rada, parametri težina su regularizirani L1 i L2 regularizacijom dok je log-loss (korištena i u XGBoost klasifikatoru) jedina razmotrena funkcija gubitka, pa stoga funkcija troška $\widetilde{L}(\theta)$ u izrazu (3.18) poprima oblik

$$\widetilde{L}(\theta) = \lambda_1 ||\mathbf{w}||_1 + \lambda_2 ||\mathbf{w}||_2^2 - \frac{1}{N} \sum_{k=1,\ldots,N} \left[y_0^{(i)} \ln \hat{y}_0^{(i)} + y_1^{(i)} \ln \hat{y}_1^{(i)}\right]. \tag{3.19}$$

[10]Ovisno o literaturi na koju se referiramo, ulazni sloj se često ne broji kao sloj modela, budući da isti samo prosljeđuje informacije u prvi skriveni sloj, bez transformiranja istih. Kako ne bismo uzrokovali konfuznost, odlučili smo brojati sve slojeve, bez obzira na njihovu funkciju.

Kako smo naveli pri objašnjenju strukture perceptrona prezentiranog pravilom pridruživanja $f(\mathbf{x}; \Theta)$, ovisnost funkcije troška o parametrima težina i parametrima pristranosti nelinearne je prirode što pak implicira njezinu nekonveksnost. Iako i dalje možemo, a zbog složenosti problema i moramo pri određivanju njezinog minimuma koristiti numeričke metode, iste su posljedično pod rizikom konvergencije ka lokalnome minimumu. Nekoliko desetljeća napredaka u području numeričke matematike i računalnih znanosti rezultiralo je šarenilom optimizacijskih algoritama korištenjem kojih bismo nad istim uzorkom postigli drugačije, možebitno bolje performanse perceptrona. Međutim, modeli su u okviru ovoga istraživanja izgrađeni isključivo na temelju Adam algoritma [4] uz pripadne vrijednosti hiperparametra jednake onima navedenima u izvornome članku – odluka je to koju opravdavamo njegovom općenitošću (suštinski predstavlja objedinjenje kvaliteta drugih algoritama u jednu metodu), njegovom adaptivnošću (stopa učenja prilagođava za pojedini parametar) te izvrsnim performansama koje bilježi nad skupovima podataka različitih veličina.

Pretpostavimo kako smo vrijednosti parametara težina inicijalizirali na slučajno odabrane vrijednosti iz dovoljno velikog susjedstva nul-vektora te kako smo parametre pristranosti inicijalizirali na nulu dobivši time inicijalnu vrijednost vektora parametara $\mathbf{\Theta}_0$. Pretpostavimo dodatno kako ćemo vrijednost parametra ažurirati n puta. Za $t \in \{1, \ldots, n\}$ algoritam nastavljamo na sljedeći način:

i) **Propagacija unaprijed.** Koristeći postupak naveden pri opisu strukture perceptrona, ali uz trenutnu vrijednost parametara težina i pristranosti, određujemo vrijednosti nad čvorovima u svim slojevima, uključujući i predikcije $\hat{\mathbf{y}}_{t-1}^{(i)} = f(\mathbf{x}^{(i)}; \mathbf{\Theta}_{t-1})$.

ii) **Propagacija unatrag.** Adam algoritam zahtjeva izračun gradijenta funkcije troška s obzirom na vektor parametara $\mathbf{\Theta}$ u odgovarajućim točkama. Motivirani općenito velikom dimenzionalnošću problema, postupak određivanja parcijalnih derivacija pojednostavljujemo metodom propagacije unazad. Počevši od izlaznoga sloja, funkciju troška deriviramo s obzirom na pojedini parametar težine ili pristranosti koji se pojavljuje na pojedinom čvoru razmatranog sloja, izračunavamo njezinu vrijednost koristeći vrijednosti dobivene pri propagiranju unaprijed te se pomičemo na sloj-prethodnik. Postupak nastavljamo dok ne dođemo do ulaznoga sloja. Zbog svojstva ulančanosti derivacije kompozicije funkcija, pri izračunu vrijednosti derivacije na svakome čvoru koristit ćemo vrijednosti koje smo izračunali na prethodnome, izbjegavajući time redundantne kalkulacije.

iii) **Adam algoritam.** Koristeći u prethodnome koraku izračunate vrijednosti gradijenta računamo sljedeću aproksimaciju vektora parametara $\mathbf{\Theta}_t$ te se vraćamo na korak i).

Predikcije višeslojnim perceptronom. Korištenjem softmax aktivacijske funkcije nad izlaznim slojem uvjetovali smo izgradnju perceptrona koji je pri svakoj iteraciji prilagođavao parametre ne bi li u slučaju da je j-ti zajmoprimac loš $\hat{y}_1^{(j)}$ bila što bliža jediničnoj vrijednosti

te u slučaju da je zajmoprimac dobar bila što bliža nuli. Stoga vjerojatnost $P(Y = 1 \,|\, \mathbf{x}^{(j)})$ procjenjujemo s $\hat{y}_1^{(j)}$ pri čemu je $\left(\hat{y}_0^{(j)}, \hat{y}_1^{(j)}\right) = f(\mathbf{x}^{(j)}; \boldsymbol{\Theta}_n)$.

Regularizacija višeslojnog perceptrona. Jedna od karakteristika svih najsuvremenijih ANN modela znatno je veći broj parametara u odnosu na broj jedinki u skupovima podataka nad kojima su isti izgrađeni. Iako time teoretski otvaramo vrata prenaučenosti modela, u praksi se čini kako kvalitetna regularizacija kompenzira velik broj parametara te rezultira istančanom robusnošću u kontekstu vanjske valjanosti. Dodatan problem proizlazi i iz načina na koji se perceptroni grade te kako se treniraju – složenost modela treba rasti sa složenošću uzoraka i šumom u podatcima. Stoga ćemo se posebnu pozornost posvetiti izboru regularizacijskih hiperparametara dok ćemo izbor što je bolje strukture modela prepustiti empiriji umjesto se oslanjati na teoriju.

- Činjenicu da imamo velik broj parametara možemo kompenzirati smanjenjem njihova doprinosa u konačnoj predikciji. Vrijednost parametara težine ćemo održavati relativno malenima primjenom L1 i L2 regularizacije. Praksa je ne regularizirati parametre pristranosti zbog moguće podnaučenosti.

- Iako činjenica kako vrijedi $\partial_{x_i}||(x_1, \ldots, x_n)||_1 = \text{sign}(x_i)$ implicira kako će L1 regularizacija rezultirati svođenjem nekih od realizacija parametara težina na nulu, dodatno ćemo proučiti utjecaj ispuštanja pojedinih konekcija među gustim slojevima implementacijom regularizacije ispuštanjem (engl. *dropout*). Dropout, iako češće interpretiran kao metoda potpunog ispuštanja slučajno uzorkovanih čvorova u slojevima koji nisu izlazni, možemo ekvivalentno protumačiti kao postavljanje izlaznih vrijednosti slučajno odabranog čvora na nulu, funkcionalno ga zanemarujući pri formiranju predikcija u sklopu jedne iteracije treninga. Upuštanje u raščlambu detalja potrebnih za apsolutno razumijevanje regularizacije ispuštanjem nadilaze volumen ovoga rada. Ukazat ćemo samo kako možemo povući paralelu između restrikcije postavljene na grananje čvorova u XGBoost algoritmu pri čemu smo razmatrali tek dio prediktora, slučajno uzorkovan iz skupa svih prediktora, i restrikcije postavljene u regularizaciji ispuštanjem pri čemu smo razmatrali tek dio čvorova, slučajno uzorkovan iz skupa svih čvorova u sloju.

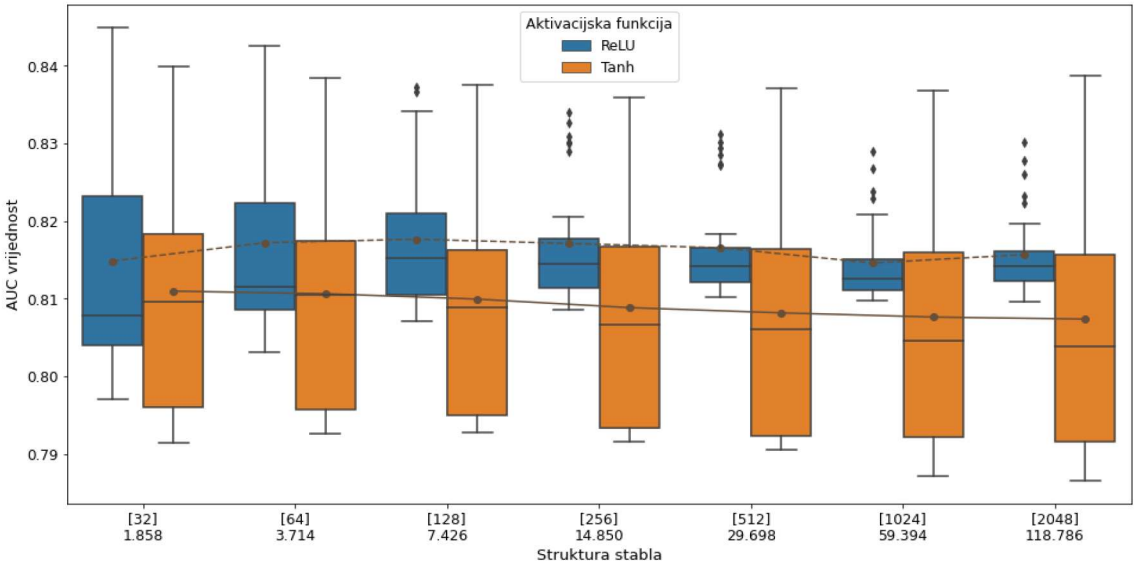
- S većim brojem epoha raste i rizik od prenaučenosti, posebice kada broj parametara nadilazi broj jedinki. Šum u podatcima uparen s relativno malenim brojem jedinki u trening-skupu povukao je značajno fluktuiranje u trajektorijama AUC vrijednosti pri treniranju, posebice kada u razmatranje uključimo regularizaciju ispuštanjem. Pokazalo se kako pristranost ka validacijskom skupu podataka kojom rezultira rano zaustavljanje sa sobom povlači nisku stopu vanjske valjanosti nad trening-skupom. Stoga smo fiksirali broj epoha u treningu na stotinu, ali rizik od prenaučenost smo umanjili korištenjem dovoljno strmog propadanja stope učenja. Početna stopa učenja fiksirana je na 0,001 dok su

razmotrene završne stope učenja s vrijednošću 0,0001 i 0,00001 te su razmotrene metoda linearnog i metoda eksponencijalnog propadanja stope učenja.

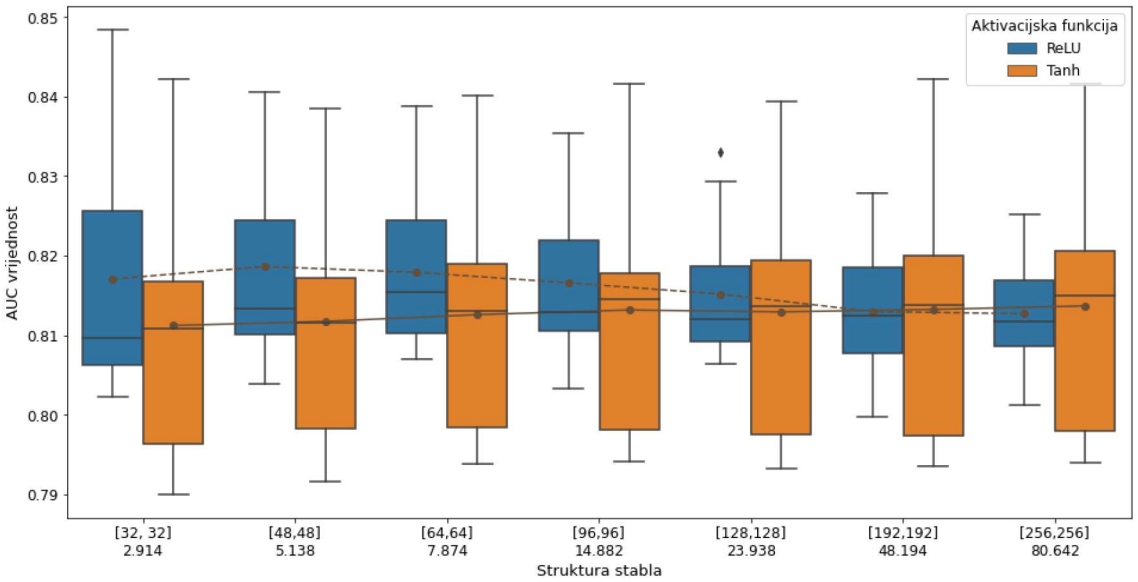
- Dvodimenzionalnost predikcija dozvoljava implementaciju zaglađivanja oznaka (engl. *label smoothing*), metode kojom uvodimo šum u stvarne realizacije ciljane varijable ne bismo li izbjegli prenapuhanost težina i uljuljkivanje modela u prenaučenost. Umjesto da modeliramo slučajan vektor $\mathbb{Y} \in \{(0,1),(1,0)\}$, modeliramo njegovu transformaciju $\mathbb{Y}' \in \{(1-\epsilon,\epsilon),(\epsilon,1-\epsilon)\}$ za *malenu* pozitivnu vrijednost ϵ (obično ne veću od 20%).

Pretraga mreže hiperparametara. Činjenica kako je struktura perceptrona u razmatranju u potpunosti određena brojem skrivenih slojeva i njihovom širinom dopušta konvencionalnost označavanja modela s $[n]$ kada perceptron ima jedan skriveni sloj s n čvorova te s $[n,m]$ kada perceptron ima dva skrivena sloja redom s n i m čvorova.

Kako je navedeno u *Metodologiji*, potpoglavlju *Binarni klasifikatori i selekcija kandidata*, prije no što smo proveli pretragu mreže hiperparametara sa ciljem izgradnje što je boljeg binarnog ANN klasifikatora, analizirali smo utjecaj promjene strukture modela na njegovu performansu, mjerenu AUC vrijednošću. Kreirali smo trideset skupova za validaciju, svaki od njih dobivši slučajnim odabirom 20% jedinki iz trening-skupa. Za svaku od četrnaest različitih struktura perceptrona smo izgradili trideset modela i svaki evaluirali nad korespondentnim validacijskim skupom, prikupivši trideset realizacija AUC vrijednosti. Na *Slici 3.6* su prikazani odgovarajući kutijasti dijagrami za perceptrone s jednim, a na *Slici 3.7* za perceptrone s dva skrivena sloja, vizualizirani odvojeno za različite aktivacijske funkcije.



Slika 3.6: Kutijasti dijagrami realizacija AUC vrijednosti ANN klasifikatora s jednim skrivenim slojem ovisno o broju čvorova i pripadnoj aktivacijskoj funkciji upotpunjeni trajektorijama srednje vrijednosti. Ispod strukture modela naveden je ukupan broj parametara u perceptronu.
stopa ispuštanja iznosi 0,25; propadanje stope učenja je linearno; završna stopa učenja iznosi 0,00005; stopa L1 regularizacije iznosi 0,0001; stopa L2 regularizacije iznosi 0,0001; parametar zaglađivanja iznosi 0,05



Slika 3.7: Kutijasti dijagrami realizacija AUC vrijednosti ANN klasifikatora s dva skrivena sloja ovisno o broju čvorova i pripadnoj aktivacijskoj funkciji upotpunjeni trajektorijama srednje vrijednosti. Ispod strukture modela naveden je ukupan broj parametara u perceptronu.
stopa ispuštanja iznosi 0,25; propadanje stope učenja je linearno; završna stopa učenja iznosi 0,00005; stopa L1 regularizacije iznosi 0,0001; stopa L2 regularizacije iznosi 0,0001; parametar zaglađivanja iznosi 0,05

Neovisno o broju skrivenih slojeva u perceptronu, korištenje *ReLU* aktivacijske funkcije rezultiralo je istim trendom inicijalnog povećanja prosjeka AUC vrijednosti popraćenim njegovim smanjenjem za složenije strukture. Dodatno, u oba slučaja uočavamo i trend smanjenja raspršenosti realizacija AUC vrijednosti s povećanjem broja parametara. Kada je nad skrivenim slojevima korištena aktivacijska funkcija bila tangens hiperbolni, prosjek AUC vrijednosti opadao je s povećanjem broja čvorova u slučaju jednoga skrivenog sloja te se isti povećavao zajedno s brojem čvorova u slučaju dva skrivena sloja. Pri pretrazi mreže hiperparametara smo se usredotočili na modele čija je struktura oblika [64], [128] i [256] te [48, 48], [64, 64] i [96, 96].

U *Tablici 3.4* je navedena mreža hiperparametara na temelju koje smo izabrali najbolje reprezentante perceptron klasifikatora u okviru ovog istraživanja. Razmotreno je 3888 modela.

Hiperparametar	**Vrijednosti razmotrene pri pretrazi mreže hiperparametara**
Struktura perceptrona	[64], [128], [256], [48,48], [64,64], [96,96].
Stopa zaglađivanja ϵ	0; 0,05; 0,10
Stopa ispuštanja čvorova	0; 0,25; 0,50;
Tip propadanja stope učenja	linearni, eksponencijalni
Završna stopa učenja	0,00001; 0,0001
Stopa L1 regularizacije	0; 0,0001; 0,001
Stopa L2 regularizacije	0; 0,0001; 0,001

Tablica 3.4: Pretražena mreža hiperparametara za familiju ANN klasifikatora.
n = 2.340 je broj jedinki u trening-podskupu

4. Rezultati

Za svaku familiju ML modela izabrano je najboljih 1% u terminima prosječne AUC vrijednosti određene peterostrukom unakrsnom validacijom u okviru pretrage mreže hiperparametara, odnosno 47 najboljih CART klasifikatora, 116 najboljih RF klasifikatora, 45 najboljih XGBoost klasifikatora te 39 najboljih ANN klasifikatora. Za svaki od četiri skupa najboljih 1% klasifikatora odredili smo prosječne vrijednosti mjera diskriminacijske snage i klasifikacijske točnosti, navedenih u *Tablici D3* u *Dodatku*, bloku *Najboljih 1% predstavnika*. Ne bismo li uspostavili što precizniju hijerarhiju u njihovoj kvaliteti, za svaku od šest mjera performansi proveli smo osam instanci t-testa, po potrebi jednostranog ili dvostranog, uspoređujući međusobno sve parove skupova najboljih predstavnika. Odnosi među modelima u terminima različitih mjera performansi prezentirani su u *Tablici 4.1* kao slijed (ne)jednakosti među odgovarajućim srednjim vrijednostima.

AUC				**ACC**			
XGBoost	$>$	RF	$(p = 3,10 \times 10^{-47})$	XBBoost	$>$	RF	$(p = 7,89 \times 10^{-43})$
RF	$>$	ANN	$(p = 3,85 \times 10^{-66})$	RF	$>$	ANN	$(p = 7,47 \times 10^{-06})$
ANN	$>$	CART	$(p = 4,33 \times 10^{-25})$	ANN	$>$	CART	$(p = 3,47 \times 10^{-25})$
O				**S**			
XGBoost	$>$	ANN	$(p = 1,14 \times 10^{-37})$	XGBoost	$>$	RF	$(p = 5,90 \times 10^{-19})$
ANN	$=$	CART	$(p = 1,34 \times 10^{-01})$	RF	$>$	ANN	$(p = 6,38 \times 10^{-14})$
CART	$>$	RF	$(p = 4,55 \times 10^{-03})$	ANN	$>$	CART	$(p = 8,42 \times 10^{-20})$
PPV				**NPV**			
XGBoost	$>$	RF	$(p = 4,72 \times 10^{-31})$	XGBoost	$>$	RF	$(p = 1,83 \times 10^{-42})$
RF	$>$	ANN	$(p = 6,61 \times 10^{-11})$	RF	$=$	ANN	$(p = 8,44 \times 10^{-01})$
ANN	$>$	CART	$(p = 1,05 \times 10^{-29})$	ANN	$>$	CART	$(p = 1,48 \times 10^{-03})$

Tablica 4.1: Odnosi među modelima u kontekstu pojedine mjere performanse skupa s p-vrijednostima jednostranih ili dvostranih t-testova kojima je isti odnos utvrđen.

Uočavamo kako, neovisno o tome kojom od razmotrenih mjera ju kvantificirali, XGBoost klasifikatori bilježe najbolju performansu, generalno praćeni RF klasifikatorima, ANN klasifikatorima te, na posljetku, CART klasifikatorima. Izuzetak navedenog pravila su slučaj mjere osjetljivosti, kada CART postiže neočekivano visoke vrijednosti, doduše na uštrb mjere specifičnosti, te u slučaju negativne prediktivne vrijednosti kada razlika među srednjim vrijednostima za RF i ANN klasifikatore nije statistički značajna.

Iz skupa modela izgrađenih pri pretrazi mreže hiperparametara za svaku smo familiju modela izdvojili njezinog najboljeg predstavnika čije smo vrijednosti hiperparametara sistematizirali u *Tablici 4.2*. Pripadne realizacije mjera performanse nad trening i test podatcima navedene su u *Tablici D3* u *Dodatku*, bloku *Najbolji predstavnik prije selekcije prediktora*.

Model	Hiperparametar	Vrijednost hiperparametra
CART	Mjera nečistoće	Entropija
	Parametar rezidbe	0,002299115
	Maksimalna dubina stabla	6
	Minimalni broj jedinki za razdiobu	$5\%n$
	Minimalni broj jedinki na listu	$1,25\%n$
RF	Mjera nečistoće	Entropija
	Broj stabala u šumi	500
	Maksimalna dubina stabla	7
	Minimalni broj jedinki za razdiobu	$0,625\%n$
	Minimalni broj jedinki na listu	$0,3125\%n$
	Bagging parametar	$0,5n$
	Broj prediktora po biparticiji	D
XGBoost	Maksimalna dubina stabla	3
	Stopa učenja	0,05
	L2 regularizacija	10‰
	Bagging parametar	$0,5n$
	Broj prediktora po biparticiji	$0,5D$
ANN	Struktura perceptrona	$[96, 96]$
	Zaglađivanje oznaka epsilon	0,05
	Stopa ispuštanja čvorova	0,25
	Tip propadanja gradijenata	linearni
	Završna stopa učenja	0,0001
	Stopa L1 regularizacije	0,001
	Stopa L2 regularizacije	0

Tablica 4.2: Vrijednosti hiperparametara najboljih predstavnika razmotrenih familija modela utvrđene pretragom mreže hiperparametara.
n = 2.340 je broj jedinki u trening-podskupu, D = 55 je broj prediktora u uzorku

Uspoređujući vrijednosti zabilježene u blokovima *Najboljih 1%* te *Najbolji predstavnik prije selekcije prediktora Tablice D3*, uočavamo kako hijerarhiju među najboljim predstavnicima familija modela prati hijerarhiju zabilježenu među skupovima najboljih 1% klasifikatora – XGBoost klasifikator postiže najbolje rezultate, neovisno o izboru mjere, generalno praćen RF i ANN klasifikatorom, dok se CART klasifikator nalazi na začelju. Izuzetci za navedeno pravilo i u slučaju najboljih predstavnika zabilježeni su za osjetljivost i negativnu prediktivnu vrijednost.

Provođenjem SFFS algoritma, za svaki od najboljih predstavnika familija modela izabrali smo deset najrelevantnijih prediktora. Modele smo potom izgradili ograničivši se isključivo na probrane prediktore te utvrdili vrijednosti mjera performansi novih inačica modela nad test-podatcima, navedenih u *Tablici D3* u *Dodatku*, bloku *Najbolji predstavnik nakon selekcije prediktora*. Usporedbom performansi modela prije i poslije provođenja SFFS algoritma, uočavamo kako se utjecaj selekcije prediktora manifestira drugačije za različite klasifikatore. U jednu ruku, performansa XGBoost i ANN klasifikatora generalno je narušena – dok je ANN klasifika-

tor zabilježio pogoršanje u svim mjerama performanse, XGBoost klasifikator zabilježio je lošiju performansu u svim mjerama izuzev osjetljivosti i negativne prediktivne vrijednosti. U drugu ruku, performansa CART i RF klasifikatora generalno je poboljšana – dok je RF klasifikator zabilježio pogoršanje isključivo u AUC vrijednosti, CART klasifikator je pogoršanje zabilježio za specifičnost i pozitivnu prediktivnu vrijednost.

Ishodi selekcije prediktora razlikuju se među modelima i u prirodi prediktora koji su algoritmom utvrđeni kao relevantni. Sistematizacija deset najrelevantnijih prediktora za svaki model navedene su u *Tablici D4* u *Dodatku*. Uočavamo kako je stupanj pokrića jedini financijski pokazatelj relevantan za četiri razmotrena modela. Pokazatelje koji su relevantni za tri od četiri razmotrena modela možemo podijeliti na one zajedničke modelima temeljenim na stablima, što su omjer neto dobiti i prihoda od prodaje te trajanja kreditiranja dobavljača, te one zajedničke modelima niže varijance, što su branša djelatnosti te koeficijent rasta prihoda od prodaje (doduše u neprekidnom i diskretnom obliku).

Polovinu najrelevantnijih prediktora za CART klasifikator čine pokazatelji aktivnosti dok među preostalim pokazateljima bilježimo dva pokazatelja profitabilnosti te po jedan pokazatelj likvidnosti, zaduženosti i ekonomičnosti. Četiri od deset najrelevantnijih prediktora za RF klasifikator čine pokazatelji aktivnosti dok među preostalim pokazateljima bilježimo dva pokazatelja zaduženosti te po jedan pokazatelj profitabilnosti, likvidnosti, rasta te pokazatelj iz grupe ostalih pokazatelja. Uočavamo marginalnu diversifikaciju pokazatelja u odnosu na CART klasifikator. Tri od deset najrelevantnijih prediktora za XGBoost klasifikator čine pokazatelji zaduženosti dok među preostalim pokazateljima bilježimo dva pokazatelja profitabilnosti te po jedan pokazatelj likvidnosti, aktivnosti, rasta, uvozne-izvozne aktivnosti te pokazatelj iz grupe ostalih pokazatelja. Uočavamo znatnu diversifikaciju pokazatelja u odnosu na RF i CART klasifikator. Tri od deset najrelevantnijih prediktora za ANN klasifikator čine ostali pokazatelji dok među preostalim pokazateljima bilježimo dva pokazatelja likvidnosti, zaduženosti i aktivnosti te jedan pokazatelj rasta. Iako uočavamo znatnu diversifikaciju pokazatelja u odnosu na CART klasifikator, isto ne možemo tvrditi kada isti usporedimo s RF i XGBoost klasifikatorima.

5. Rasprava

Analizom vrijednosti obuhvaćenih *Tablicom D3* uočavamo kako točnost hipoteza prezentiranih u zadnjim ulomcima poglavlja *Pregled literature*, potpoglavlja *Srodna istraživanja - poduzetnički kreditni rizici* ovisi o metodi kvantifikacije kvalitete modela koju razmatramo te o inačici modela koju razmatramo. Ako se usredotočimo na diskriminacijsku snagu modela, neovisno govorili o najboljih 1% klasifikatora za svaku od familija modela ili o najboljim predstavnicima prije selekcije prediktora, empirijski rezultati istraživanja podupiru pretpostavke o hijerarhiji modela kao i odnose između performansi ansambala modela i pojedinačnih modela koje su zabilježili *Celik et al.* (2020) te *D'Addona et al.* (2022) – XGBoost klasifikator bilježi najbolju performansu,

praćen RF klasifikatorom, zatim ANN klasifikatorom te, na posljetku, CART klasifikatorom. Navedena je hijerarhija narušena selekcijom prediktora, kada CART klasifikator prelazi na poziciji ANN klasifikatora.

Međutim, pitanje odnosa među familijama klasifikatora u kontekstu mjera klasifikacijske točnosti nema jedinstveni odgovor, posebice nakon selekcije najrelevantnijih prediktora, što gotovo u cijelosti možemo pripisati problemu (ne)arbitrarnosti selekcije granične vrijednosti pri kategorizaciji predikcija vjerojatnosti. Odabir granične vrijednosti minimizacijom udaljenosti između osjetljivosti i specifičnost nad trening-podatcima u slučaju razmatranih podataka generalno rezultira značajnim jazom među spomenutim vrijednostima nad test-podatcima zbog čega, usprkos njezinoj analitičkoj smislenosti nad trening-podatcima, ista poprima svojstvo skore arbitrarnosti nad test-podatcima. Dodatno, zbog prorijeđenosti vrijednosnog skupa predikcija vjerojatnosti CART klasifikatorom (procjena $\hat{P}(Y = 1 \mid \mathbb{X})$ stablom maksimalne dubine n može poprimiti najviše 2^n različitih vrijednosti), korištenu metodu odabira granične vrijednosti ne možemo smatrati prihvatljivom u slučaju stabla odlučivanja. Čak i uz robusnije, generalno kvalitetnije metode odabira granične vrijednosti (koje su ukratko predstavljene u referiranoj literaturi), pretpostavka kako će jedna metoda biti adekvatna za sve familije klasifikatora i dalje može rezultirati suboptimalnom klasifikacijom. Istraživanje bi stoga valjalo proširiti podrobnijom usporedbom različitih metoda odabira granične vrijednosti ne bismo li osigurali lakšu usporedbu performanse kvantificirane klasifikacijskom točnošću. Na prije navedene poteškoće uzrokovane poluproizvoljnim izborom granične vrijednosti nadovezuje se i problem hijerarhije važnosti među razmotrenim mjerama klasifikacijske točnosti, odnosno koju ćemo od uparenih mjera smatrati prioritetom te koliko smo spremni komprimirati među realizacijama uparenih mjera. Primjerice, iako intuitivno ima smisla preferirati nešto veće vrijednosti osjetljivosti u usporedbi sa specifičnošću budući da netočno označavanje lošeg poduzeća kao dobrog predstavlja veći financijski rizik od netočnog označavanja dobrog poduzeća kao lošeg, koliki kompromis između osjetljivosti i specifičnosti smatramo prihvatljivim valja utvrditi podrobnijom analizom te bi ga također valjalo uzeti u obzir pri odabiru granične vrijednosti. S obzirom na opisanu složenost problema uspostave hijerarhije među ML metodama, preostaje tek uspostaviti koji je klasifikator kandidat za najbolji u kontekstu generalne klasifikacijske točnosti, što je titula koju smo odlučili dodijeliti XGBoost klasifikatoru – prije selekcije najrelevantnijih prediktora, tvrdnja vrijedi trivijalno, dok nakon selekcije prediktora tvrdnju argumentiramo činjenicom kako u slučaju mjera za koje isti ne bilježi najbolje rezultate (osjetljivost i specifičnost), XBGoost klasifikator za najboljim klasifikatorom zaostaje vrijednošću manjom od 1,5% te u oba slučaja postiže vrijednost koja je druga u slijedu po veličini.

Promjene u hijerarhiji performansi nakon selekcije najrelevantnijih prediktora dijelom je rezultat načina na koji su najrelevantniji prediktori odabrani, točnije arbitrarnosti odluke kako ćemo za svaki model razmotriti skup od točno deset prediktora – neopravdano je očekivati kako će CART klasifikator razmjerno malene maksimalne dozvoljene dubine svoju najbolju per-

formansu zabilježiti nad jednakim brojem relevantnih prediktora kao i ANN klasifikator koji generalno beneficira od veće dimenzije ulaznoga sloja. Stoga ističemo kako bi popratna istraživanja trebala uključiti inačice SFFS algoritma drugačijeg zaustavnog kriterija, kao što su minimalno relativno povećanje vrijednosti mjere performanse dodavanjem pojedinog prediktora, razni kriteriji multikolinearnosti i slično.

Opservacija kako su optimalne vrijednosti pojedinih hiperparametra, odnosno njihove vrijednosti pri izgradnji najboljih predstavnika familija modela, nerijetko ekstremi pripadnog skupa vrijednosti analiziranog algoritmom pretrage mreže hiperparametra povlači slutnju kako skupovi vrijednosti hiperparametara nisu kvalitetno određeni te da pojedine familije modela mogu zabilježiti (moguće znatno) bolje vrijednosti mjera performansi. Naveden trend posebno je istančan u slučaju RF klasifikatora čiji je najbolji predstavnik nastojao izgraditi što je moguće dublja stabla sa što je manjim restrikcijama glede broja jedinki nad listovima i minimalnog broja jedinki potrebnih za biparticiju čvora, povećavajući heterogenost šume sa što je moguće manjom vrijednošću bagging parametra. Iz navedenoga možemo zaključiti kako RF algoritam postavljanjem što blažih restrikcija na strukturu njime obuhvaćenih stabala nastoji izgraditi što je kompleksnije pojedinačne klasifikatore te rezultirajuću varijabilnost umanjiti bagging principom. Dodatno, gušći bi skupovi vrijednosti hiperparametara analizirani algoritmom pretrage mreže hiperparametara ili pak implementacija nasumične pretrage mreže hiperparametara (engl. *random (grid) search*) u konačnici mogle rezultirati sveobuhvatno boljim modelima.

Istaknimo još kako imamo teorijskog i empirijskog razloga vjerovati da drugi predstavnici metoda dubokog učenja mogu biti znatno kvalitetniji od perceptrona razmotrenih u okvirima ovoga istraživanja. Izvrsne performanse metoda dubokog učenja u analiziranim sustavnim pregledima literature u pravilu su zabilježene za konvolucijske i rekurzivne neuronske mreže, strukturom višestruko složenije od analiziranih perceptrona. Dodatno, napreci u području analize tabularnih podataka metodama dubokog učenja kulminirali su 2019. godine kreacijom *TabNet* modela, transformera specijaliziranog za formiranje predikcija na temelju tabularnih podataka općenito veće diskriminacijske snage od XGBoost klasifikatora, koji dozvoljava veću interpretabilnost no prije spomenute složenije metode dubokoga učenja [3]. Uz činjenicu kako je najbolji predstavnik među perceptronima i strukturno najsloženiji te kako bi isti moguće beneficirao od povećanja broja skrivenih slojeva (i)li broja njima obuhvaćenih čvorova, uočavamo kako je pretrage mreže hiperparametara preferirao što sporije propadanje stope učenja, kako odabirom linearnog propadanja umjesto eksponencijalnog, tako i odabirom veće od dvije razmotrene finalne vrijednosti stope učenja. Navedenim motivirani ističemo kako bi u daljnjem istraživanju metodu propadanja stope učenja valjalo zamijeniti regularizacijom ranim zaustavljanjem uz postavljanje zahtjeva na minimalni broj epoha učenja ne bismo li izbjegli u potpoglavlju *Višeslojni perceptron* navedeni fenomen podnaučensoti, kako bi razmotreni perceptroni trebali biti strukturno složeniji te kako bi iste valjalo upotpuniti i drugim metodama nadziranoga dubokoga učenja.

Literatura

[1] P. M. Addo, D. Guegan, B. Hassani, *Credit Risk Analysis Using Machine and Deep Learning Models*, Risks 6 (M. Steffensen, Ed.), MDPI, Basel, 2018.

[2] A. Ara, G. B. Fernandes, F. Louzada, *Classification methods applied to credit scoring: Systematic review and overall comparison*, Surveys in Operations Research and Management 21 (F. S. da Gama, Ed.), Elsevier, Amsterdam, 2016. Str. 117-134.

[3] S. Ö. Arik, T. Pfister, *TabNet: Attentive Interpretable Tabular Learning*, Proceedings of the AAAI Conference on Artificial Intelligence 35(2021), AAAI Press, Palo Alto. Str. 6679-6687.

[4] J. L. Ba, D. P. Kingma, *Adam: A Method for Stochastic Optimization*, 3rd International Conference on Learning Representations (2015), San Diego.

[5] B. Baesens, S. Lessmann, H.-V. Seow, L. Thomas, *Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research*, Surveys in European Journal of Operational Research 247 (R. Słowiński, Ed.), Elsevier, Amsterdam, 2015. Str. 124-136.

[6] S. Ben-David, S. Shalev-Scwartz, *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, Cambridge, 2014a. Str. 250-255.

[7] S. Ben-David, S. Shalev-Scwartz, *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, Cambridge, 2014b. Str. 268-281.

[8] Y. Bengio, A. Courville, I. Goodfellow, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016. Str. 165-270.

[9] M. Benšić, N. Šuvak, *Primijenjena statistika*, Sveučilište J.J. Strossmayera, Odjel za matematiku, Osijek, 2013, str. 134.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, 2006a. Str. 663-666.

[11] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, 2006b. Str. 225-269.

[12] T. Celik, X. Dastile, M. Potsane, *Statistical and machine learning models in credit scoring: A systematic literature survey*, Applied Soft Computing Journal 91 (Mario Köppen, Ed.), Elsevier, Amsterdam, 2020.

[13] T. Chen, C. Guestrin, *XGBoost: A Scalable Tree Boosting System*, KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016), San Francisco. Str. 785–794.

[14] M. M. Cornett, A Saunders, *Financial Institutions Management: A Risk Management Approach*, McGraw-Hill Education, New York, 2014. Str. 176;291-298.

[15] S. D'Addona, W. Luo, G. Pau, s. Shi, R. Tse, *Machine learning-driven credit risk: a systemic review*, Neural Computing and Applications 34 (J. MacIntyre, Ed.), Springer, New York, 2022.

[16] F. J. Ferri, M. Hatef, J. Kittler, P. Pudil, *Comparative Study of Techniques for Large-Scale Feature Selection*, Machine Intelligence and Pattern Recognition 16 (E. S. Gelsema, L. S. Kanal, Eds.), Elsevier, Amsterdam, 1994. Str. 403-413.

[17] E. W. Forgy, J. H. Myers, *The Development of Numerical Credit Evaluation Systems*, Journal of the American Statistical Association 58(1963), str. 799-806.

[18] T. Hastie, G. James, R. Tibshirani, D. Witten, *An Introduction to Statistical Learning with Applications in R*, Springer, New York, 2013. Str. 303-324.

[19] M. Jamshidian, M. Mata, *2 - Advances in Analysis of Mean and Covariance Structure when Data are Incomplete*, Handbook of Computing and Statistics with Applications, Handbook of Latent Variable and Related Models (S.-Y. Lee, Ed.), Elsevier, Amsterdam, 2007. Str. 21-44.

[20] A. Klamargias, V. Siakoulis, E. Stavroulakis, A. Petropoulos, *A robust machine learning approach for credit risk analysis of large loan level datasets using deep learning and extreme gradient boosting*, IDEAS (2018)
URL:`https://ideas.repec.org/h/bis/bisifc/50-22.html` [5.09.2022.]

[21] T. C. Koopmans, P. A. Samuelson, J. R. N. Stone, *Report of the Evaluative Committee for Econometrica*, Econometrica 22(1954), str. 141-146.

[22] A. J. Larner, *The 2x2 Matrix: Contingency, Confusion and the Metrics of Binary Classification*, Springer Nature Switzerland AG, Cham, 2021. Str. 15-24.

[23] D. Micci-Barreca, *A Preprocessing Scheme for High-Cardinality Categorical Attributes in Classification and Prediction Problems*, ACM SIGKDD Explorations Newsletter 3 (U. Fayyad, Ed.), Association for Computing Machinery, New York, 2001. Str. 27-32.

[24] V. K. Rohatgi, A. K. Md. E. Saleh, *An Introduction to Probability and Statistics*, John Wiley & Sons, New Jersey, 2015. Str. 519.

[25] L. C. Thomas, *A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers*, International Journal of Forecasting 16 (J. G. de Gooijer, Ed.), Elsevier, Amsterdam, 2000. Str 149–172.

[26] L. Vidovic, L. Yue, *Machine Learning and Credit Risk Modelling*, S&P Global Market Intelligence (2020)
URL:`https://www.spglobal.com/marketintelligence/en/news-insights/blog/machine-learning-and-credit-risk-modelling` [5.09.2022.]

[27] dmlc XGBoost (2021), *XGBoost Documentation*
URL: `https://xgboost.readthedocs.io` [27.08.2022.]

Dodatak

Grupa	Financijski pokazatelj	Formula
Pokazatelji ekonomičnosti	Ekonomičnost ukupnog poslovanja	UP / ukupni rashodi
	Ekonomičnost poslovnih aktivnosti	PP / poslovni rashodi
	Koeficijent financiranja dobiti	EBITDA / financijski rashodi
Pokazatelji profitabilnosti	Neto rentabilnost imovine	100 × dobit razdoblja / UI
	Neto rentabilnost vlastitog kapitala	100 × dobit razdoblja / kapital
	Neto marža profita	100 × dobit razdoblja / UP
	Omjer neto dobiti i PP	
	Omjer zadržane dobiti i UI	
Pokazatelji likvidnosti	Koeficijent tekuće likvidnosti	KI / KO
	Koeficijent ubrzane likvidnosti	(KI - zalihe) / KO
	Koeficijent trenutne likvidnosti	novac / KO
	Omjer KI i UI	
	Omjer gotovine i PP	
	Omjer gotovine i UI	
	Omjer gotovine i UO	
Pokazatelji zaduženosti	Koeficijent zaduženosti	(DO + KO) / UI
	Odnos duga i kapitala	(DO + KO) / kapital
	Stupanj pokrića I	kapital / DI
	Stupanj pokrića II	(kapital + DO) / DI
	Omjer DO i KI	
	Omjer KO i kapitala *(nep.)*	
	Omjer KO i kapitala *(kat.)*	
	Omjer KO i UI	
	Koeficijent vlastitog financiranja	kapital / UI
	Omjer kredita i UI	
	Omjer kredita i prihoda	
	Omjer UO i UP	
	Omjer UO i EBITDA	
	Omjer KO i DO	

Tablica D1: Financijski pokazatelji, odnosno prediktori kojima su opisana poduzeća u uzorku. Pokazatelji koji su dio stručnog nazivlja imenovani su po standardima struke te je naknadno navedena pripadna formula. Za pokazatelje iz čijega se naziva može ustanoviti metoda izračuna formula nije navedena.
UP = ukupni prihodi, PP = prihodi od prodaje; UI = ukupna imovina, KI = kratkotrajna imovina, DI = dugotrajna imovina; UO = ukupne obveze, KO = kratkoročne obveze, DO = dugoročne obveze; nep. = neprekidna inačica, kat. = kategorizirana inačica

Grupa	Financijski pokazatelj	Formula
Pokazatelji aktivnosti	Koeficijent obrtaja UI	UP / UI
	Koeficijent obrtaja DI	UP / DI
	Koeficijent obrtaja KI	UP / KI
	Koeficijent obrtaja zaliha	PP / zalihe
	Koeficijent obrtaja potraživanja	PP / potraživanja
	Dani vezivanja zaliha	365 × zalihe / PP
	Trajanje naplate potraživanja	365 × potraživanja / PP
	Trajanje kreditiranja dobavljača	365 × MT / KO_d
	Omjer OK i UI	
	Omjer PP i OK	
	Omjer potraživanja i UI	
Pokazatelji rasta	Koeficijent rasta prodaje *(nep.)*	
	Koeficijent rasta prodaje *(n-kat.)*	$(PP_{2019} - PP_{2018})/PP_{2018}$
	Koeficijent rasta prodaje *(i-kat.)*	
Pokazatelji R&D	Omjer NI i UI	
	Omjer izdataka za razvoj i UI	
Pokazatelji produktivnosti	Poslovni prihodi po zaposleniku	
Pokazatelji UIA	Omjer PP u inozemstvu i PP	
	Omjer uvoza i PP	
	Omjer uvoza i UI	
	Omjer PP u inozemstvu i uvoza	
Pokazatelji invest. aktivnosti	Omjer investicija u DMNI i UI	
	Omjer investicija u DMNI i UP	
Ostali pokazatelji	Branša djelatnosti	
	Broj zaposlenih	
	Županija u kojoj poduzeće posluje	

Tablica D2: Financijski pokazatelji, odnosno prediktori kojima su opisana poduzeća u uzorku. Pokazatelji koji su dio stručnog nazivlja imenovani su po standardima struke te je naknadno navedena pripadna formula. Za pokazatelje iz čijega se naziva može ustanoviti metoda izračuna formula nije navedena.
UP = ukupni prihodi, PP = prihodi od prodaje; UI = ukupna imovina, NI = nematerijalna imovina, KI = kratkotrajna imovina, DI = dugotrajna imovina, DMNI = dugotrajna materijalna i nematerijalna imovina; UO = ukupne obveze, KO = kratkoročne obveze, DO = dugoročne obveze; nep. = neprekidna inačica, n-kat. = kategorizacija po negativnosti, i-kat. = kategorizacija po stopi insolventnosti; UIA = uvozna - izvozna aktivnost, KO_d = kratkoročne obveze prema dobavljačima, OK = obrtni kapital (OK = KI - DO)

Najboljih 1% predstavnika							
Model	**Skup**	**AUC**	**ACC**	**O**	**S**	**PPV**	**NPV**
ANN	Trening	0.852153±0.001066	0.774532±0.001568	0.774518±0.001564	0.774546±0.001573	0.755866±0.001660	0.792163±0.001476
	Test	0.798448±0.000556	0.735743±0.001905	0.724599±0.002501	0.745788±0.002502	0.719843±0.002214	0.750302±0.001907
CART	Trening	0.863499±0.001499	0.780355±0.001901	0.793017±0.006332	0.768944±0.006342	0.756155±0.003693	0.805231±0.003690
	Test	0.790846 ±0.000834	0.713987±0.002255	0.731069±0.008086	0.698591±0.006562	0.686477±0.003042	0.743015±0.004289
XGBoost	Trening	0.953119±0.005745	0.878196±0.009105	0.878269±0.009124	0.878130±0.009089	0.866635±0.009833	0.888896±0.008414
	Test	0.857930±0.001139	0.774196±0.001686	0.767851±0.002813	0.779913±0.002945	0.758800±0.002327	0.799522±0.001919
RF	Trening	0.902227±0.001542	0.814661±0.001536	0.814646±0.001537	0.814675±0.001536	0.798461±0.001638	0.829836±0.001436
	Test	0.825000±0.000301	0.740626±0.003719	0.719915±0.001154	0.759292±0.001011	0.729430±0.000826	0.750506±0.000763
Najbolji predstavnik prije selekcije prediktora							
Model	**Skup**	**AUC**	**ACC**	**O**	**S**	**PPV**	**NPV**
ANN	Trening	0.850795	0.774094	0.774094	0.773879	0.755274	0.791889
	Test	0.798679	0.737705	0.729107	0.745455	0.720798	0.753281
CART	Trening	0.870870	0.788107	0.795242	0.781676	0.766504	0.809011
	Test	0.790564	0.725410	0.743516	0.709091	0.697297	0.754144
XGBoost	Trening	0.900499	0.816667	0.816952	0.816409	0.800353	0.831954
	Test	0.863176	0.781421	0.780980	0.781818	0.763380	0.798408
RF	Trening	0.909511	0.823308	0.823360	0.823262	0.807638	0.837963
	Test	0.828482	0.744536	0.743516	0.745455	0.724719	0.763298
Najbolji predstavnik nakon selekcije prediktora							
Model	**Skup**	**AUC**	**ACC**	**O**	**S**	**PPV**	**NPV**
ANN	Trening	0.840253	0.770335	0.770007	0.770630	0.751583	0.788040
	Test	0.790965	0.726776	0.714697	0.737662	0.710602	0.741514
CART	Trening	0.856560	0.769994	0.839942	0.706953	0.720916	0.830534
	Test	0.805199	0.729508	0.812680	0.654545	0.689518	0.794953
XGBoost	Trening	0.900474	0.810256	0.810640	0.809911	0.793469	0.826015
	Test	0.860579	0.769126	0.804035	0.737662	0.734211	0.806818
RF	Trening	0.898950	0.812030	0.811824	0,812216	0.795760	0.827267
	Test	0.824028	0.750000	0.749280	0.750649	0.730337	0.768617

Tablica D3: Realizacije (intervalnih procjena) mjera performanse klasifikatora. Narančastom bojom su naglašene najveće realizacije svakog pojedinog pokazatelja nad test-podatcima, posebno za najboljih 1% te najbolje pojedinačne predstavnike prije i poslije provođenja SFFS algoritma.

CART	**RF**	**XGBoost**	**ANN**
EK_Ekonomičnost poslovnih aktivnosti	OST_Branša djelatnost	OST_Branša djelatnost	OST_Branša djelatnost
PROF_Omjer neto dobiti i PP	PROF_Omjer neto dobiti i PP	PROF_Omjer neto dobiti i PP	OST_Broj zaposlenih
PROF_Omjer zadržane dobiti i UI	LIKV_Omjer gotovine i UI	PROF_Omjer zadržane dobiti i UI	OST_Županija
LIKV_Koeficijent trenutne likvidnosti	ZAD_Omjer UO i UP	LIKV_Koeficijent trenutne likvidnosti	LIKV_Koeficijent ubrzane likvidnosti
ZAD_Stupanj pokrića I	ZAD_Stupanj pokrića I	ZAD_Omjer UO i UP	LIKV_Omjer potraživanja i UI
AKT_Koeficijent obrtaja KI	AKT_Koeficijent obrtaja UI	ZAD_Omjer kredita i prihoda	ZAD_Koeficijent vlastitog financiranja
AKT_Koeficijent obrtaja potraživanja	AKT_Omjer potraživanja i UI	ZAD_Stupanj pokrića II	ZAD_Stupanj pokrića I
AKT_Koeficijent obrtaja UI	AKT_Trajanje kreditiranja dobavljača	AKT_Trajanje kreditiranja dobavljača	AKT_Omjer potraživanja i UI
AKT_Trajanje kreditiranja dobavljača	AKT_Trajanje naplate potraživanja	RAST_Koeficijent rasta prodaje *(i-kat.)*	AKT_Dan vezivanja zaliha
AKT_Omjer PP i OK	RAST_Koeficijent rasta prodaje *(nep.)*	IZV_Omjer uvoza i PP	RAST_Koeficijent rasta prodaje *(i-kat.)*

Tablica D4: Deset najrelevantnijih prediktora za najbolje predstavnike odgovarajućih familija modela određenih SFFS algoritmom. Redoslijed prediktora u pojedinom stupcu je arbitraran, zadovoljavajući uvjet da pokazatelji iz iste grube budu u slijedu.
UP = ukupni prihodi, PP = prihodi od prodaje; UI = ukupna imovina, DI = dugotrajna imovina, KI = kratkotrajna imovina; UO = ukupne obveze; nep. = neprekidna inačica, i-kat. = kategorizacija po stopi insolventnosti; OK = obrtni kapital (OK = KI - DO)

Sažetak: Cilj istraživanja bilo je usporediti performanse CART metode, metode slučajne šume, metode ekstremnog podizanja gradijenta te metode višeslojnog perceptrona u procjeni kreditnoga rizika za iz javnih baza podataka uzorkovana mala i srednja poduzeća s područja Republike Hrvatske. Istraživanje je provedena u tri faze: između skupova najboljih 1% predstavnika svake metode te između pojedinačnih najboljih predstavnika metoda prije i poslije selekcije relevantnih pokazatelja SFFS algoritmom. Hijerarhija kvalitete predstavnika unutar četiriju familija metoda utvrđena je algoritmom pretrage mreže hiperparametara. Zabilježena je dominacija ansambla modela nad pojedinačnim modelima pri čemu je, generalno govoreći, XGBoost postigao najbolje rezultate, praćen slučajnom šumom, perceptronom te, na posljetku, klasifikacijskim stablom. Istraživanje valja proširiti dodatnim metodama (LR, SVM), složenijim metodama dubokog učenja (CNN, RNN, TabNet) te ga obogatiti većim skupom jedinki opisanih većim brojem pokazatelja.

Ključne riječi: Procjena kreditnog rizika, mala i srednja poduzeća, strojno učenje, duboko učenje, klasifikacijsko i regresijsko stablo, CART, slučajna šuma, RF, ekstremno podizanje gradijenta, XGBoost, višeslojni perceptron, pretraga mreže hiperparametara, sekvencijska selekcija prediktora unaprijed, SFFS

Credit risk assessment for SMEs using machine and deep learning methods

Abstract: The objective of the research was to compare the performances of the CART, random forest, extreme gradient boosting and multilayer perceptron methods in credit risk assessment for small and medium-sized enterprises sampled from public databases in the Republic of Croatia. We conducted the research in three phases: between the 1% of the best representatives of each method and between the individual best representatives of the methods before and after selecting correspondent relevant features using the SFFS algorithm. The hierarchy of the quality of the representatives within the four families of methods was established by the hyperparameter grid search algorithm. We noted a trend of ensemble methods dominating individual models, with XGBoost achieving the best results in general, followed by random forest, perceptron and, finally, the classification tree. The research should be expanded with additional methods (LR, SVM), more complex deep learning methods (CNN, RNN, TabNet) and enriched with a greater set of individuals described by a greater number of indicators.

Keywords: credit risk assessment, SMEs, machine learning, deep learning, classification and regression tree, CART, random forest, RF, extreme gradient boosting, XGBoost, multilayer perceptron, hyperparameter grid search, sequential forward feature selection, SFFS

Životopis

Zovem se Antonio Krizmanić i rođen sam 29. rujna 1997. u Osijeku. Po završetku Osnovne škole „Lovas" upisujem Gimnaziju „Vukovar" gdje se prvi put počinjem ozbiljnije posvećivati znanosti ostvarujući izvrsne rezultate na natjecanjima iz matematike, biologije i geografije. Završivši srednjoškolsko obrazovanje, 2016. godine upisujem Preddiplomski studij matematike na Odjelu za matematiku Sveučilišta u Osijeku. Isti sam studij završio 2019. godine s najvećom pohvalom, a za završni rad na temu „Redni i kardinalni brojevi. Hipoteza kontinuuma." mi je dodijeljena Rektorova nagrada godinu nakon. Iste sam godine nastavio obrazovanje na Odjelu upisujući Diplomski studij financijske matematike i statistike. Dio svojeg visokog obrazovanja proveo sam na Karlovu sveučilištu u Pragu. U trenutku obrane diplomskoga rada sam nezaposlen.