Usporedba tehnika strojnog učenja u predikciji akademske uspješnosti

Martinović, Ivan

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Ivan Martinović

Usporedba tehnika strojnog učenja u predikciji akademske uspješnosti

ZAVRŠNI RAD

Varaždin, 2020.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Ivan Martinović

Matični broj: 45890/17–R

Studij: Informacijski sustavi

Usporedba tehnika strojnog učenja u predikciji akademske uspješnosti

ZAVRŠNI RAD

Mentorica:

Doc. dr. sc. Dijana Oreški

Varaždin, srpanj 2020.

Ivan Martinović

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

_____

Sažetak

U teorijskom dijelu ću analizirati znanstvene članke na temu strojnog učenja na području obrazovanja i usporediti dobivene podatke i rezultate iz različitih članaka. Pronaći najkorištenije metode u predviđanju uspješnosti i najčešće pronađene atribute najbolje za modele. U praktičnom dijelu ću upotrijebiti različite tehnike strojnog učenja za predikciju implementirane u Python-u. Analizirati rezultate dobivene izradom vlastitih programa strojnog učenja i usporediti preciznost predviđanja različitih tehnika strojnog učenja nad istim skupom podataka.

Ključne riječi: strojno učenje; neuronske mreže; stablo odlučivanja; linearna regresija; uspješnost studenata; predviđanje; Python

Sadržaj

1. Uvod

Tema završnog rada je usporedba tehnika strojnog učenja u predikciji akademske uspješnosti. U teorijskom dijelu rada ću analizirati znanstvene članke koji se bave temom strojnog učenja na području obrazovanja, a u praktičnom dijelu ću sam izraditi programe strojnog učenja za nekoliko različitih tehnika nad odabranim skupom podataka. Prikazati ću koje su prednosti toga da se koristi strojno učenje za analiziranje podataka studenata i predikcije njihove uspješnosti. Neki od znanstvenih članaka koriste strojno učenje kako bi predvidjeli studente koji bi mogli biti izbačeni. Velika korist strojnog učenja na području obrazovanja je kako bi se profesori mogli prilagoditi određenim studentima kod kojih se mogu predvidjeti neke poteškoće ili kod onih studenata za koje se misli da bi mogli biti izbačeni.

Strojno učenje nije uvijek točno, ali ima veliki postotak preciznosti u većini slučajeva i mala odstupanja u predviđanju te se sve više razvija. Najpoznatiji programski jezici za pisanje programa strojnog učenja su Python i R, no Python se koristi malo više zbog jednostavnosti. Python je također korišteniji zbog biblioteke koja se koristi za strojno učenje, a to je Tensorflow, ona se koristi u velikom broju projekata na temu strojnog učenja. Neke od vrlo poznatih tvrtki koriste Tensorflow, poput Google, Intel i DeepMind.

2. Analiza prethodnih istraživanja

U ovom poglavlju ću prikazati podatke iz znanstvenih članaka koji se bave strojnim učenjem i rudarenjem podataka na području obrazovanja te neke od najkorištenijih metoda za predikciju i koje su sve prednosti korištenja strojnog učenja u obrazovanju.

Učinkovitost studenata je jedna od bitnijih stvari obrazovnih institucija i svako sveučilište želi imati velika akademska postignuća budući da su ona mjeritelj visoke kvalitete sveučilišta. Diplomiranje studenata se najčešće koristi kao mjera učinkovitosti studenata na sveučilištima. „A Review on Predicting Student's Performance using Data Mining Techniques (Shahiri, Husain i Rashid, 2015.) su u svome znanstvenom članku analizirali tehnike strojnog učenja na području Malezije gdje se zaključna ocjena koristi za pregled učinkovitosti studenata. Zaključna ocjena ovisi o strukturi kolegija, ocjenama koje dobiju iz ostalih aktivnosti, broju bodova na završnom ispitu i izvannastavnim aktivnostima. Provjera učinkovitosti se obavlja kako bi se moglo vidjeti kako dobro funkcionira obrazovni plan ili struktura određenog kolegija. Kako Aldowah, Al-Samarraie i Fauzy (2019.) navode u „Educational Data Mining and Learning Analytics for 21stcentury higher education: A Review and Synthesis" rudarenje podatak u obrazovanju se koristi kako bismo mogli promijeniti kurikulum, izbacili određene faktore koji utječu na neuspješnost studenata ili izbacivanje studenata iz sveučilišta, povećanje učinkovitosti i kvalitete podučavanja studenata i slično. Budući da većina sveučilišta koristi online sustav za upravljanje učenjem većina materijala iz kojih studenti uče su dostupni online i kolokviji iz nekih kolegija se obavljaju online te su tako i također dostupni za analizu uz pomoć strojnog učenja.

Postoje i znanstveni članci koji se bave nečim mnogo zabrinjavajućim, a to je analizom studenata koji prijevremeno napuštaju školovanje i onih koji ne uspiju proći kolegije. Znanstveni članci poput „Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout" (Concepción, Campanario, Peña, Lara, Lizcano, Martínez, 2017.) i „Educational Data Driven Decision Making: Early Identification of Students at Risk by Means of Machine Learning" (Kovač i Oreški, 2018.) se bave analizom tih problema. Korist od ovih istraživanja je vrlo pozitivna i dokazuje kako strojno učenje može imati veliki učinak na području poput obrazovanja. Ovi članci traže razloge zbog kojih studenti napuštaju fakultet ili zbog kojih neki kolegij ima veliku stopu pada. Traže se razlozi koji dovode do pada, gdje se može raditi ili o preteškom gradivu ili loše dostupnim informacijama ili o premalom trudu

studenata na kolegiju. Zahvaljujući sve češćem korištenju online sustava za upravljanje učenjem sve više podataka je dostupno za korištenje u rudarenju podataka na području obrazovanja.

Provedeno je i istraživanje diskriminativnom analizom na Fakultetu organizacije i informatike u Varaždinu koje pronalazi koji su najutjecajniji faktori u odlučivanju uspješnosti studenata. Provedene su 2 ankete, prva anketa je provedena 2006./07. akademske godine nad studentima druge godine preddiplomskog smjera Informacijski i poslovni sustavi i druga anketa provedena 2008./09. akademske godina nad studentima treće i četvrte godine preddiplomskog smjera Informacijski sustavi. Rezultati prvog istraživanja su pokazali da faktori koji su utjecali na uspjeh na predmetu matematika su komunikacija između studenata i profesora, metoda podučavanja, uključenost studenata i važnost kolegija kroz studij. Diskriminativna analiza predmeta Odabrana poglavlja matematike je pokazala kako je 7 varijabli utjecalo na uspješnost polaganja predmeta kroz kolokvije: smatram kolokvije korisnima, smatram obaveznu zadaću korisnom, broj sati tjedno provedenih učeći, povezanost predavanja i seminara, sudjelovanje na projektu je bilo zanimljivo i korisno, seminari su bili zanimljivi i korisni te smatram da su obvezna prisustva predavanjima i seminarima korisna. Na temelju prve ankete izrađena je druga anketa gdje je pronađeno da 54 % studenata pronalazi motivaciju za trud u tome da završe fakultet, pronađu posao i krenu zarađivati što prije (posvećenost cilju). Pronađeno je prosječno trajanje završetka faksa za studente od 5.7 godina i s ocjenom 3.43. Važne varijable koje razlikuju najuspješnije i manje uspješne studente su sljedećih 8 varijabli: rezultati prijemnog ispita, osobna odgovornost, prva dobivena ocjena na fakultetu, dobro upravljanje vremenom, sposobnost raspodjele vremena za učenje, stil učenja priprema za kolegij i priprema za aktivnosti na kolegiju. (Divjak i Oreški, 2009.)

Postoje i osobni faktori koji mogu utjecati na uspješnost studenta. Kasnije je provedeno još jedno istraživanje na Fakultetu informatike i organizacije u Varaždinu gdje je pokazano da četiri faktora koja najviše utječu na uspješnost studenta su povezana uz socijalne i demografske karakteristike, također su vidjeli povezanost položaja na rang listi prilikom upisa na studij i odabir studijskog programa na uspješnost studenta. Prilikom analize ankete pronađene su razlike između muških i ženskih ispitanika. Muški ispitanici smatraju važnije faktore poput upravljanje vremenom i motivacije, dok ženski ispitanici stavljaju veću vrijednost na zdravlje, točnije na zdravstvene probleme i da studentice najčešće završe zadatke nekoliko dana prije roka predaje. Prema dostupnim podacima možemo vidjeti da na fakultetu postoji razlika u uspješnosti između muških i ženskih studenata i također da studenti koji imaju dobro predsveučilišno znanje imaju veću uspješnost na fakultetu. (Oreški, Hajdin i Kliček, 2016.)

Osim navedenih prethodnih istraživanja postoje slična istraživanja koja se bave ovom temom poput Analysis of Student Behaviour and Success Based on Logs in Moodle (Kadoić i Oreški, 2018.), Analysis of ICT students' LMS engagement and success (Oreški i Kadoić, 2018.) te Estimating profile of successful IT student: data mining approach (Oreški, Konecki i Milić, 2017.).

2.1. Metodologija

Razlog zašto se ova istraživanja izvode je kako bismo vidjeli koliko dobro funkcionira trenutni kurikulum na nekom fakultetu ili postoje li problemi s nekim kolegijem. Izvodi se sustavno preispitivanje kako bismo pronašli metode koje se mogu koristit za predviđanje s dostupnim podacima. Tako su u svome istraživanju Shahiri i sur. (2015.) postavili kriterije za istraživačka pitanja prikazana u tablici 1.

Tablica 1. Kriteriji za istraživačka pitanja

Populacija	Sveučilište (učinkovitost studenata)
Tretman	Metode/tehnike za predviđanje
Ishod	Preciznost predviđanja, uspješne tehnike za predviđanje
Kontekst	Obrazovne institucije. Sve vrste iskustvenog proučavanja poput preliminarno istraživanje, upitnici, eksperimenti i studije slučaja

(Prema: Shahiri i sur., 2015.)

Neka istraživanja koriste atribute koji se mogu prikupiti s online sustava za učenje, a neka istraživanja koriste atribute koje mogu pronaći provedbom anketa ili upitnika po fakultetu. Većina znanstvenih članaka poput A Comparative Study of Machine Learning Approaches on Learning Management System Data (Oreški i Hajdin, 2019.), Educational Data Driven Decision Making: Early Identification of Students at Risk by Means of Machine Learning (Oreški i Kovač, 2018.), Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout (Burgos i Campanario,2017.) koriste online sustave učenje kao izvor

podataka, no radovi Analyzing undergraduate students' performance using educational data mining (Asif, Merceron, Ali i Haider, 2017.) i Utilizing early engagement and machine learning to predict student outcomes (Gray i Perkins, 2019.) koriste ocjene na kolegijima te učestalost dolaska na predavanja i seminare za podatke u radu. Postoji i znanstveni članak Prediction of academic performance associated with Internet usage behaviors using machine learning algorithms (Xu, Wang, Peng i Wu, 2019.) koji koristi vrijeme provedeno na internetu, brzinu preuzimanja, brzinu prenošenja podataka i uređaj koji je korišten za spajanje na Internet.

Poznati standard za rudarenje podataka je međuindustrijski standardni postupak za rudarenje podataka (eng. *Cross-industry standard process for dana mining – CRISP DM*) koji se često koristi prilikom upotrebe tehnika strojnog učenja nad skupom podataka te su ovaj standard u svome radu primijenili Oreški i Hajdin (2019.). Ovaj standard je detaljno opisan u radu CRISP-DM process model in educational setting (Oreški, Pihir i Konecki, 2017.) gdje oni navode kako se CRISP DM sastoji od 6 koraka:

1. Razumijevanje posla (eng. *Business understanding*): sastoji se od razumijevanja koji je cilj procesa i onda pretvaranje toga znanja i problem rudarenja podataka
2. Razumijevanje podataka (eng. *Dana understanding*): pretraživanje podataka, potvrda kvalitete podataka i pronalazak odstupanja
3. Priprema podataka (eng. *Dana preparation*): vremenski najzahtjevniji korak koji pokriva sve aktivnosti vezane uz pripremu podataka. Pretvaranje u tablicu, odabir atributa i transformacija i provjera nedostajućih redova
4. Modeliranje (eng. *Modeling*): sastoji se od izgradnje i procjene modela. Potrebno je odabrati modele koje ćemo koristiti te se ti modeli primjenjuju i njihovi parametri se mijenjaju radi postizanja optimalnih vrijednosti
5. Evaluacija modela (eng. *Evaluation*): prikazuje koliko je dobar i precizan model i daje interpretaciju modela
6. Implementacija (eng. *Deployment*): odlučuje kako se dobiveni rezultati trebaju koristiti i kome su najkorisniji

2.2. Važni faktori u predviđanju učinkovitosti studenata

Prilikom istraživanja svi znanstveni članci su koristili dva faktora: atribute i metode za predviđanje. Rudarenje podataka (eng. *Data mining*) je najpopularnija tehnika za analizu

učinkovitosti studenata. Počelo se sve češće koristiti na području obrazovanja 2015. godine. Rudarenje podataka u obrazovanjue (eng. *Educational data mining*) je proces koji izdvaja korisne podatke i uzorke iz velike baze podataka obrazovanja. Dobiveni podaci se koriste za predviđanje učinkovitosti studenata te time pomažu profesorima bolji pristup predavanju. (Shahiri i sur., 2015.)

2.2.1. Važni atributi korišteni u predviđanju učinkovitosti studenata

Očekivano najkorišteniji atribut u istraživanjima je kumulativ prosjeka ocjena. Kumulativ prosjeka ocjena se koristi kako bi se odredio postignuti akademski potencijal. Znanstveni članak A Review on Predicting Student's Performance using Data Mining Techniques (Shahiri i sur., 2015.) osim kumulativa prosjeka ocjena koristi i unutarnju procjenu u svojim podacima. Unutarnjom procjenom se smatraju ocjene iz zadaća, kvizovi, labosi, kolokviji i prisustvo. U istom znanstvenom članku su pronašli kako su sljedeći najkorišteniji atributi: demografija studenata i vanjska procjena. Demografija studenata uključuje spol, dob, obitelj i invaliditet. Vanjska procjena se smatra ocjena koju su studenti dobili na završnom ispitu iz određenog predmeta. Shahiri i sur. (2015.) kao razlog zašto većina istraživača koristi demografiju studenata poput spola je zato što muški i ženski studenti imaju različite stilove učenja. Kako su naveli u svome radu gdje su koristili podatke iz znanstvenog članka autora Meit, Borges, Cubic i Seibel (2007.) da većina ženskih studenata imaju razne pozitivne stilove učenja i ponašanja, a muški studenti potpuno suprotno.

Fernandes, Holanda, Victorino, Borges, Carvalho i Erven (2019.) su u svome radu pronašli kako su atributi poput susjedstva (gdje student stanuje) i škola koju studenti pohađaju glavni faktori koji utječi na neuspjeh studenta. Ovi podaci mogu pomoći profesorima da shvate koji studenti žive na problematičnim mjestima te da im je potrebna veća pomoć da ih se usmjeri na pravi put.

2.2.2. Korištene metode predviđanja uspješnosti studenata

Kako bi se izgradili modeli predviđanja koristi se nekoliko zadataka: klasifikacija, regresija i kategorizacija. Najpopularniji zadatak za predviđanje uspješnosti studenata je klasifikacija. Postoji nekoliko algoritama pod klasifikacijom koji su korišteni u predviđanju, neki od njih su: stablo odlučivanja, neuronske mreže, Naive Bayes, metoda k-najbližih susjeda i metoda potpornih vektora. (Shahiri i sur., 2015.)

U svome radu Burgos i sur. (2017.) također navode kako su koristili samo jednu metodu nad svojim podacima i to je metodu logističke regresije. Odlučili su se za tu metodu samo zbog jednog razloga, zato što su imali zapravo binarni rezultat predviđanja, hoće li netko proći ili ne.

Oreški i Hajdin (2019.) su koristili statističku značajnost kako bi provjerili kolika je zapravo točna dobivena preciznost provedenih tehnika strojnog učenja. Ovom analizom su dokazali da postoji velika razlika statističke značajnosti između neuronske mreže i stabla odlučivanja te k-najbližih susjeda i naivni Bayesov klasifikator. Analiza je provedena nad podacima dobivenih iz online sustava za učenje.

2.2.2.1. Stablo odlučivanja

Stablo odlučivanja je jedna od popularnih tehnika za predviđanje. Većina istraživača je koristila ovu tehniku zbog jednostavnosti i jasnoće da se otkriju male ili velike strukture podataka i predvidi vrijednost. Shahiri i sur. (2015.) navode da su Romero C., Ventura S., Espejo P. G. i Hervás C. (2008.) rekli da modeli stabla odlučivanja se lako razumiju zbog njihovog postupka rasuđivanja i mogu se direktno pretvoriti u niz ako-onda pravila. Dokaz da je ova metoda popularna je i rad Oreški, Pihir, Konecki (2017.) gdje je ovo bila jedina metoda koja je korištena za predviđanje. Pokazali su da ova metoda nije kompliciрana za izraditi te da ima popriličnu dobru preciznost i malu pogrešku uz lako očitanje važnih atributa prema stablu odlučivanja.

Metoda stabla odlučivanja nije samo u većini slučajeva precizna u predviđanju, nego je i brza u izvršavanju modela što je još jedan od razloga zašto se ova metoda stalno koristi. (Oreški i Hajdin, 2019.)

2.2.2.2. Neuronske mreže

Neuronske mreže su još jedna od popularnih tehnika. Prednost neuronskih mreža je što ima mogućnost prepoznavanja svih mogućih interakcija između prediktorskih varijabli. Neuronske mreže također imaju mogućnost potpune detekcije bez ikakve sumnje čak i u kompleksnim nelinearnim vezama između ovisnih i neovisnih varijabli. Radi navedenih podataka tehnika neuronskih mreža se bira kao najbolja metoda predviđanja. (Shahiri i sur., 2015.)

2.2.2.3. Naivni Bayesov klasifikator

Naivni Bayes algoritam je također jedna od često korištenih opcija za predviđanje. Ovaj algoritam je korišten u 4 od 30 radova gdje je cilj sva četiri rada bio pronaći najučinkovitiju

tehniku predviđanja uspješnosti studenata koristeći usporedbu. Njihova istraživanja su pokazala da je Naivni Bayes klasifikator algoritam koristio sve atribute iz skupa podataka te nakon toga je analizirao važnost i neovisnost svakog podatka. (Shahiri i sur., 2015.)

Naivni Bayesov klasifikator ima veću preciznost kada radi s kategorijskim podacima, nego kada radi s brojevima. Skupovi podataka koji najčešće sadrže numeričke vrijednosti su skupovi podataka dobiveni iz online sustava za učenje, stoga ovo nije najbolja metoda za korištenje u predikciji ukoliko imamo takve podatke. (Oreški i Hajdin, 2019.)

Adekitan i Salau (2019.) su pokazali da je ova metoda poprilično precizna gdje su ju koristili u usporedbi s poznatijim metodama kao stablo odlučivanja i linearna regresija i imala je veliku preciznost gdje je bila razlika između ostalih metoda od $\pm$ 1%.

2.2.2.4. Metoda k-najbližih susjeda

Tri istraživačka rada su pokazala da metoda k-najbližih susjeda daje najbolje rezultate s dobrom preciznošću. Metoda k-najbližih susjeda je trebala manje vremena da prepozna učinkovitost studenata kao one koji sporo nauče, prosječno nauče, dobro nauče i izvrsno nauče. Ova metoda daje dobru preciznost u procjeni detaljnih uzoraka za napredovanje studenata u visokoškolskom obrazovanju. (Shahiri i sur., 2015.)

2.2.2.5. Metoda potpornih vektora

Metoda potpornih vektora je nadgledana metoda učenja koja se koristi za klasifikaciju. Ova metoda je dobra za manji skup podataka i ima dobru i bržu od ostalih mogućnost generalizacije, također ova metoda ima najveću točnost predviđanja u prepoznavanju studenata koji će biti izbačeni. (Shahiri i sur., 2015.)

Pretpostavku koju su postavili Shahiri i sur. za to čemu ova metoda najbolje služi se može vidjeti i u radu Kovač i Oreški (2018.) gdje je ova metoda kod njih imala preciznost od 95.7%.

2.3. Zaključak znanstvenih članaka

Neuronske mreže imaju najveći postotak preciznosti od 98 %, nakon njih je metoda stabla odlučivanja s 91 %. Sljedeća je metoda potpornih vektora i metoda k-najbližih susjeda koje imaju istu preciznost od 83 % te na kraju metoda s najmanjim postotkom preciznosti je Naivni Bayes klasifikator s 76 %.

Metoda neuronskih mreža je koristila atribute koji su bili kombinacija unutarnje i vanjske procjene. Korištenjem samo jedne varijable koja je vanjska procjena uzrokuje pad preciznosti od 1%. Drugi najveći postotak preciznosti je imala metoda stabla odlučivanja s 91 %. U metodu stabla odlučivanja čimbenik koji je imao najveći učinak na preciznost je kumulativni prosjek ocjena. Sljedeća je metoda potpornih vektora s postotkom preciznosti od 83 %. Prema analizi najbolji skup podataka za metodu potpornih podataka su podaci koji imaju psihometrijske faktore. Preciznost ove metode je pala na 73% ukoliko bismo dodali izvannastavne aktivnosti. Posljednja metoda koja ima najmanji postotak preciznosti od 76 % je Naivni Bayes klasifikator. Varijable koje ova metoda koristi su kumulativni prosjek ocjena, demografija studenata, pohađana srednja škola, stipendija i korištenje društvenih mreža. Nabrojani atributi su također korišteni u nekim radovima kod neuronskih mreža i stabla odlučivanja, ali Naivni Bayes klasifikator metoda se pokazala najpreciznija nad ovim skupom podataka. Razlog tome je povezanost između ovih podataka kada se koristi Naivna Bayes metoda. (Shahiri i sur., 2015.)

Kovač i Oreški (2018.) su dokazali u svome radu kako se osim tradicionalnim načinom prikupljanja podataka za korištenje prilikom predviđanja, podaci mogu pronaći i online, ako fakultet koristi online sustav za učenje poput Moodle-a, također su prikazali da se s velikom preciznošću mogu pronaći studenti koji bi mogli imati poteškoće u prolasku nekog kolegija.

Miguéis, Freitas, Garcia i Silva (2018.) također prepoznaju velike prednosti korištenja strojnog učenja u pronalasku studenata koji imaju slične karakteristike kako bi predvidjeli njihovo ponašanje i moguće probleme.

Uspješnost studenata se može povećati pravilnim korištenjem dobivenih rezultata predviđanja. Možemo prepoznati koji atributi su česti kod studenata s jako dobrim rezultatima u školovanju te koji su česti kod studenata koji odustaju od nekih predmeta. (Oreški i Hajdin, 2019.)

3. Opis podataka

Skup podataka na temelju kojeg sam radio praktični dio završnog rada preuzet je sa stranice Kaggle kao .csv datoteka dostupna na linku https://www.kaggle.com/dipam7/student-grade-prediction. Nakon odabira skupa podataka uveo sam podatke u program JMP 15 kako bih proveo analizu podataka i provjerio postoje li neki nedostajući ili krivi podaci. Podaci sadrže informacije o istraživanju koje je provedeno 2008. godine u dvije portugalske škole, Gabriel Pereira i Mousinho da Silveira. Provedeno je istraživanje za dva predmeta: matematiku i portugalski jezik. U ovom završnom radu sam odabrano istraživanje o matematici. Broj muških i ženskih ispitanika je približno jednak s 53% ženskih i 47% muških.

Ovaj skup podataka se sastoji od 395 instanci i 33 varijable, varijable koje sadrži su opisane u tablici 2.

Tablica 2. Opis skupa podataka (vlastita izrada)

Redni broj	**Engleski naziv**	**Hrvatski naziv**	**Opis varijable**	**Vrsta varijable**
1.	school	naziv škole	Naziv učenikove škole, moguće vrijednosti su 'GP' - Gabriel Pereira ili 'MS' - Mousinho da Silveira	binarna
2.	sex	spol	Spol, moguće vrijednosti su 'F' - žensko ili 'M' - muško	binarna
3.	age	dob	Dob, moguće vrijednosti su od 15 do 22	brojčana
4.	address	adresa	Učenikova adresa, moguće vrijednosti su 'U' - grad ili 'R' - selo	binarna
5.	famsize	broj članova obitelji	Broj članova obitelji, moguće vrijednosti su 'LE3' - manje ili jednako 3 ili 'GT3' - više od 3	binarna
6.	Pstatus	odvojenost roditelja	Žive li roditelji zajedno, moguće vrijednosti su 'T' - žive zajedno ili 'A' - žive odvojeno	binarna
7.	Medu	obrazovanje majke	Moguće vrijednosti: 0 – bez obrazovanja, 1 - osnovna škola (prva 4 razreda), 2 –	brojčana

			od 5. do 9. razreda, 3 - srednja škola ili 4 - viša stručna sprema	
8.	Fedu	obrazovanje oca	Moguće vrijednosti: 0 – bez obrazovanja, 1 - osnovna škola (prva 4 razreda), 2 – od 5. do 9. razreda, 3 - srednja škola ili 4 - viša stručna sprema	brojčana
9.	Mjob	status zaposlenosti majke	Moguće vrijednosti: 'teacher' (u obrazovanju), 'health' (u zdravstvu), 'services' (administrativni poslovi), 'at_home' (kod kuće) ili 'other' (ostalo)	tekstualna
10.	Fjob	status zaposlenosti oca	Moguće vrijednosti: 'teacher' (u obrazovanju), 'health' (u zdravstvu), 'services' (administrativni poslovi), 'at_home' (kod kuće) ili 'other' (ostalo)	tekstualna
11.	reason	razlog upisa	Moguće vrijednosti su close to 'home' (blizina kuće), school 'reputation' (ugled škole), 'course' preference or 'other'	tekstualna
12.	guardian	skrbnik	Moguće vrijednosti su 'mother' (majka), 'father' (otac) or 'other' (drugo)	tekstualna
13.	traveltime	trajanje putovanje	Trajanje putovanja od kuće do škole, moguće vrijednosti su 1 - <15 min., 2 - 15 do 30 min., 3 - 30 min. do 1 sat, ili 4 - >1 sat	brojčana
14.	studytime	trajanje učenja	Tjedno trajanje učenja, moguće vrijednosti su 1 - <2 sata, 2 - 2 do 5 sati, 3 - 5 do 10 sati, ili 4 - >10 sati	brojčana
15.	failures	padovi	Broj dosadašnjih padova predmeta, moguće vrijednosti su n ako 1<=n<3, inače 4	brojčana

16.	schoolsup	stipendija	Potpora u obrazovanju, moguće vrijednosti su ‘yes’ (da) ili ‘no’ (ne)	binarna
17.	famsup	potpora obitelji	Potpora učeniku u obrazovanju od strane obitelji, moguće vrijednosti su ‘yes’ (da) ili ‘no’ (ne)	binarna
18.	paid	plaćene instrukcije	Je li učenik pohađao dodatne instrukcije iz matematike, moguće vrijednosti su ‘yes’ (da) ili ‘no’ (ne)	binarna
19.	activities	aktivnosti	Izvanastavne aktivnosti. Moguće vrijednosti su 'yes' (da) ili 'no' (ne)	binarna
20.	nursery	vrtić	Je li učenik pohađao vrtić. Moguće vrijednosti su 'yes' (da) ili 'no' (ne)	binarna
21.	higher	više obrazovanje	Želi li učenik nastaviti školovanje. Moguće vrijednosti su 'yes' (da) ili 'no' (ne)	binarna
22.	internet	pristup internetu	Pristup internetu kod kuće. Moguće vrijednosti su 'yes' (da) ili 'no' (ne)	binarna
23.	romantic	status veze	Moguće vrijednosti su 'yes' (da) ili 'no' (ne)	binarna
24.	famrel	kvaliteta obiteljskih odnosa	Moguće vrijednosti su od 1 - jako loša do 5 - odlična	brojčana
25.	freetime	slobodno vrijeme	Slobodno vrijeme nakon škole, moguće vrijednosti su od 1 - vrlo nisko do 5 - vrlo visoko	brojčana
26.	goout	izlasci	Izlasci s prijateljima, moguće vrijednosti su od 1 - vrlo niski do 5 - vrlo visoki	brojčana
27.	Dalc	dnevna konzumacija	Dnevna konzumacija alkohola, moguće vrijednosti su od 1 - vrlo niska do 5 - vrlo visoka	brojčana
28.	Walc	tjedna konzumacija	Tjedna konzumacija alkohola, moguće	brojčana

			vrijednosti su od 1 - vrlo niska do 5 - vrlo visoka	
29.	health	zdravlje	Trenutno zdravstveno stanje, moguće vrijednosti su od 1- jako loše do 5- jako dobro	brojčana
30.	absences	izostanci	Broj izostanaka u školi. Moguće vrijednosti su od 0 do 93	brojčana
31.	G1	prvo polugodište	Moguće vrijednosti su od 0 do 20	brojčana
32.	G2	drugo polugodište	Moguće vrijednosti su od 0 do 20	brojčana
33.	G3	zaključna ocjena	Moguće vrijednosti su od 0 do 20	brojčana

3.1. Korelacija između podataka

U programu JMP 15 je napravljena analiza korelacije između varijable G3 koja se predviđa i sur.h brojčanih varijabli. Napravljena je Multivariate analiza korelacije između svih varijabli brojčanog tipa gdje možemo vidjeti da na varijablu G3 najviše utječu varijable G1 i G2. Varijabla G1 ima vrijednost korelacije 0.8015, a varijabla G2 ima vrijednost korelacije 0.9049. Prema analizi možemo vidjeti da varijable absences (izostanci), freetime (slobodno vrijeme), famrel (kvaliteta obiteljskih odnosa) imaju malu korelaciju s varijablom G3 te tako ne utječu previše na nju.

G2	G1	0.8521	395	0.8226	0.8770	<.0001*
G3	age	-0.1616	395	-0.2562	-0.0639	0.0013*
G3	Medu	0.2171	395	0.1211	0.3092	<.0001*
G3	Fedu	0.1525	395	0.0546	0.2474	0.0024*
G3	traveltime	-0.1171	395	-0.2133	-0.0187	0.0199*
G3	studytime	0.0978	395	-0.0009	0.1946	0.0521
G3	failures	-0.3604	395	-0.4433	-0.2714	<.0001*
G3	famrel	0.0514	395	-0.0475	0.1493	0.3086
G3	freetime	0.0113	395	-0.0875	0.1099	0.8227
G3	goout	-0.1328	395	-0.2285	-0.0346	0.0082*
G3	Dalc	-0.0547	395	-0.1525	0.0442	0.2785
G3	Walc	-0.0519	395	-0.1498	0.0470	0.3032
G3	health	-0.0613	395	-0.1590	0.0376	0.2239
G3	absences	0.0342	395	-0.0646	0.1325	0.4973
G3	G1	0.8015	395	0.7631	0.8342	<.0001*
G3	G2	0.9049	395	0.8852	0.9213	<.0001*

Slika 1 Prikaz korelacije između varijabli (vlastita izrada)

4. Opis tehnika strojnog učenja

Za praktični dio rada sam odabrao izraditi četiri različite tehnike za predviđanje podataka. Jednu od metoda regresije, linearnu regresiju, dvije metodu klasificiranja, stablo odlučivanja i k-najbliži susjedi i za kraj jednu metodu dubokog učenja, neuronsku mrežu. U nastavku ću opisati svaku od ove četiri metode.

4.1. Linearna regresija

Jedna od najjednostavnijih tehnika regresije je linearna regresija. Linearna regresija je analiza veze između ovisnih i neovisnih varijabli. Ona pretpostavlja linearnu vezu između dvije varijable. Formula za linearnu regresiju je sljedeća:

$$Y = a + bX$$

Varijabla Y predstavlja ovisnu varijablu, varijabla X neovisnu varijablu te varijabla a predstavlja odsječak regresijske linije i varijabla b nagib regresijske linije.

Na temelju promjene varijabli a i b se može dobiti više regresijskih linija koje odgovaraju, a cilj je pronaći najbolju koja odgovara podacima. Najbolja regresijska linija je ona koja ima najmanju pogrešku između stvarnih podataka i predviđenih podataka i to računamo uz pomoć iduće formule:

$$q = \sum (y_{promatran} - y_{predviđen})^2$$

gdje q predstavlja ukupnu kvadratnu pogrešku. Pokušavamo dobiti što manju ukupnu grešku prema kojoj dobijemo vrijednosti za odsječak i nagib regresijske linije. (Sarkar, Bali, Sharma, 2018., str. 319)

4.2. K-najbliži susjedi

Najčešći zadatak rudarenja podataka je klasifikacija. Primjer klasifikacije u obrazovanju je prepoznavanje kako je potrebno pomoći nekome studentu. U klasifikaciji uvijek postoji ciljana kategorijska varijabla kao na primjer zaključna ocjena. K-najbliži susjedi je najkorištenija metoda za klasifikaciju.

Ova metoda funkcionira tako da joj damo postojeće podatke s varijablom koju predviđamo i onda kada dobijemo novi podatak metoda uspoređuje najsličniji postojeći podatak te prema tome predviđa vrijednost. Metoda k-najbliži susjedi iako najkorištenija kod klasifikacije ima i

svoje nedostatke, poput koliko susjeda bi trebala imati svaka točka, odnosno koja je vrijednost varijable k te da li sve točke imaju jednaku važnost i slični problemi.

Određivanje koji su podaci najsličniji se izvodi koristeći metrike udaljenosti. Metrika udaljenosti ili funkcija udaljenosti je realna funkcija d, koja za bilo koje koordinate x, y, z ima sljedeća pravila:

1. $d(x,y) \geq 0 \;\&\; d(x,y) = 0, ako\ i\ samo\ ako\ x = y$
2. $d(x,y) = d(y,x)$
3. $d(x,z) \leq d(x,y) + d(y,z)$

Prvo svojstvo osigurava da je udaljenost uvijek pozitivan broj i nula jedino ako su koordinate x i y jednake. Drugo svojstvo je svojstvo komutativnosti, tako da na primjer udaljenost između točaka A i B, bude jednaka kao i udaljenost između točaka B i A. Treće svojstvo predstavlja nejednakost trokuta, što znači da uvođenje treće točke neće nikada smanjiti udaljenost između dvije točke.

Najkorištenija funkcija udaljenosti je Euklidska udaljenost koja predstavlja stvarni prikaz udaljenosti u svijetu. Formula Euklidske udaljenosti glasi:

$$d_{Euklid}(x,y) = \sqrt{\sum_i (x_i - y_i)^2}$$

(Larose i Larose, 2015., str. 301)

4.3. Stablo odlučivanja

Druga korištena metoda klasifikacije u ovom radu je metoda stabla odlučivanja. Stablo odlučivanja je niz čvorova spojene granama (eng. *branches*) koje se pružaju prema dole iz korijenskog čvora (eng. *root node*) sve dok ne dođu do završetka u lisnom čvoru (eng. *leaf node*). Korijenski čvor se nalazi na samome vrhu grafa stabla odlučivanja te se nakon njega testiraju atributi prema kojima se pronalaskom novih rezultata stvaraju grananja. Svako grananje vodi do novog čvora odluke ili završava lisnim čvorom.

Postoje određeni zahtjevi kako bismo mogli koristiti stablo odlučivanja:

1. Algoritam stabla odlučivanja spada pod nadzirano učenje i kao takvo zahtjeve klase varijable koja se predviđa. Mora se primijeniti skup podataka za učenje koji pruža algoritmu vrijednosti varijable koja se predviđa.
2. Skup podataka za učenje mora imati dosta varijacija u rezultatu i imati veliki broj instanci što pruža algoritmu stabla odlučivanja veliki niz grananja za klasifikaciju podataka koji se kasnije dodaju. Stablo odlučivanja uči na temelju primjera, ako

nedostaju primjeri za neki skup podataka, klasifikacija i predviđanje će biti loši ili nemogući za taj skup podataka.

3. Atribut koji se predviđa mora biti točno određen, što znači da vrijednosti tog atributa moraju biti točno definirane tako da se lako prepozna pripada li neka instanca određenom skupu ili ne.

Za izradu stabla odlučivanja dva najkorištenija algoritma su:

- Algoritam stabala klasifikacije i regresije (eng. *Classification and regression trees algorithm – CART*)
- C4.5 algoritam

Stabla odlučivanja kreirana uz pomoć CART algoritma su binarna, što znači da svaka odluka ima samo 2 grananja. Algoritam C4.5 ima veliku prednost nad CART algoritmom zato što nije ograničen na binarna grananja. Ovaj algoritam stvara stabla odlučivanja s različitim izgledom grananja.

(Larose i Larose, 2015., str. 317)

4.4. Neuronske mreže

Duboko učenje je postalo jedna od najpoznatijih predstavnika strojnog učenja. Aplikacije koje koriste duboko učenje su postalo vrlo precizne na polju prepoznavanja slika i zvuka. Duboko učenje se može smatrati proširenjem neuronskih mreža.

Neuronske mreže rade na principu učenja iz rasprostranjenih podataka. Najjednostavnije neuronske mreže se sastoje od: ulaznog sloja (eng. *input layer*), jedan skriveni sloj (eng. *hidden layer*) i izlazni sloj (eng. *output layer*).

Proces podučavanja neuronske mreže podrazumijeva sljedeće korake:

1. Definiranje strukture ili arhitekture neuronske mreže. Ovo je jako bitan korak, ako kreiramo vrlo opsežnu mrežu s velikim brojem neurona onda naš model neće vrlo dobro generalizirati podatke.
2. Odabrati nelinearnu transformaciju za primjenu na svaku vezu. Ova transformacija kontrolira učinkovitost svakog neurona u mreži.
3. Odlučiti se za funkciju gubitka koju ćemo koristiti za izlazni sloj. Ovo vrijedi ako imamo problem koji koristi nadgledano učenje.
4. Naučiti parametre neuronske mreže, odnosno odrediti težinske vrijednosti svake konekcije. Težinske vrijednosti se određuju optimizacijom funkcije gubitka.

(Sarkar i sur., 2018., str. 102)

5. Analiza podataka u Pythonu

U ovom poglavlju ću opisati kako funkcionira kod napisan u Pythonu koji izvršava četiri navedene tehnike objašnjene u prethodnom poglavlju nad skupom podataka. Svi programi su koristili pandas biblioteku za rad s .csv datotekom i Tensorflow skup biblioteka za rad s tehnikama strojnog učenja.

5.1. Linearna regresija

Prva na redu je metoda linearne regresije i ona je najjednostavnija za izradu. Sve što je trebalo je uvesti podatke koji moraju biti brojčanog tipa i nad njima provesti linearnu regresiju.

```
import pandas as pd
import numpy as np
import sklearn
from sklearn import linear_model
import matplotlib.pyplot as pyplot
from matplotlib import style


podaci = pd.read_csv("student-mat.csv", sep=";")
podaci = podaci[["G1", "G2", "G3", "studytime", "failures", "absences"]]


predvidanje = "G3"


X = np.array(podaci.drop([predvidanje], 1))
Y = np.array(podaci[predvidanje])


x_trening,          x_test,          y_trening,          y_test          =
sklearn.model_selection.train_test_split(X, Y, test_size=0.2)
```

```
linearnaRegresija = linear_model.LinearRegression().fit(x_trening, y_trening)
preciznost = linearnaRegresija.score(x_test, y_test)
print("Preciznost predviđanja podataka: ", preciznost)


parametar = 'G1'
graf = pyplot.gcf()
graf.canvas.set_window_title('Graf linearne regresije')
style.use("ggplot")
pyplot.scatter(podaci[parametar], podaci["G3"])
pyplot.xlabel(parametar)
pyplot.ylabel("Zaključna ocjena (G3)")
pyplot.show()
```

U prvih 6 linija koda sam uvezao sve potrebne biblioteke za izradu programa. Biblioteka pandas se koristi za čitanje csv datoteka, numpy je biblioteka koja podržava rad s nizovima i matematičkim funkcijama nad nizovima, sklearn je biblioteka u kojoj se nalazi funkcija za rad s linearnom regresijom i matplotlib je biblioteka koja služi za grafički prikaz.

Nakon toga u varijablu podaci spremam željene varijable nad kojima ću izvršiti linearnu regresiju, na ovome primjeru su odabrane varijable G1, G2, studytime, failures, absences i varijabla koja će se predviđati, G3, spremljena u posebnu varijablu predvidanje. Kada smo odredili koje podatke ćemo koristiti razdvojimo podatke u 2 niza, jedan niz X u kojem se nalaze svi odabrani podaci osim varijable koju predviđamo i drugi niz Y u kojem se nalazi samo podatak koji predviđamo.

Prije nego što primijenimo linearnu regresiju nad podacima, moramo razdvojiti podatke na podatke za trening i podatke za testiranje modela. Odvajanje podataka se radi metodom train_test_split koja određuje koliko podataka ide na koji dio ovisno o vrijednosti parametra test_size, u mome slučaju sam stavio test_size=0.2 što znači da će od svih podataka 20% podataka biti testni podaci, a 80 % podaci za trening linearne regresije.

Sama linearna regresija se sprema u varijablu linearnaRegresija te se provodi naredbom linear_model.LinearRegression().fit(x_trening, y_trening) gdje se ona uči nad podacima za trening. Koliko je dobra linearna regresija provjeravam tako da provedem linearnu regresiju

nad testnim podacima te gledam koliki je postotak preciznosti i to spremam u varijablu preciznost.

Zadnji odlomak koda, 8 linija koda na kraju programa, služe za grafički prikaz linearne regresije. Varijabla parameter služi za prikaz varijable koju ću prikazati na linearnom grafu u usporedbi s varijablom G3.

5.2. K-Najbliži susjedi

Isto kao i prethodna metoda na početku programa nalaze se potrebne biblioteke za rad programa. Biblioteka pandas za rad s csv datotekama te biblioteka numpy za rad s nizovima i za matematičke operacije. Iz biblioteke sklearn potreban nam je KNeighborsClassifier koji sadrži funkciju za izvršavanje metode k-najbliži susjedi i iz iste biblioteke kneighbors_graph kako bismo mogli grafički prikazati uz pomoć biblioteke matplotlib.

```
import pandas as pd
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.neighbors import kneighbors_graph
import matplotlib.pyplot as pyplot

podaci = pd.read_csv("student-mat.csv", sep=";")
podaci = podaci[["G1", "G2", "G3", "studytime", "failures", "absences"]]

predvidanje = "G3"

X = np.array(podaci.drop([predvidanje], 1))
Y = np.array(podaci[predvidanje])

knn = KNeighborsClassifier(n_neighbors=2)
```

```
knn.fit(X, Y)

knnPredvidanje = knn.predict(X)


stvarniPodaci = Y

pogreska = (((knnPredvidanje - stvarniPodaci) ** 2).sum()) /
len(knnPredvidanje)

print(pogreska)


graf = kneighbors_graph(X, 8, mode='connectivity', include_self=True)


pyplot.spy(graf)

pyplot.show()
```

Kao i kod linearne regresije prvo učitavamo podatke iz csv datoteke te odvajamo ostale varijable i varijablu koju predviđamo u dva niza X i Y. U varijablu knn spremamo pripremljenu metodu k-najbližih susjeda KNeighborsClassifier i postavljamo broj susjeda (n_neighbors) na 2. Nakon toga koristimo spremljene podatke u nizove X i Y kako bismo nad njima proveli metodu naredbom knn.fit() i spremimo predviđanje podataka u varijablu knnPredvidanje. Na kraju računao prosječnu kvadratnu pogrešku tako da usporedimo stvarne podatke za predviđanje i knnPredvidanje te spremimo u varijablu pogreska.

Graf k-najbližih susjeda se prikazuje naredbom kneighbors_graph() te se prikazuje pomoću matplotlib biblioteke i pyplot naredbe spy() koja služi za prikaz dvodimenzionalnog niza podataka.

5.3. Stablo odlučivanja

Za metodu stabla odlučivanja potrebne su nam slične biblioteke kao i prije, biblioteka pandas za rad s csv datotekama, biblioteka sklearn iz koje su preuzete funkcije za izvođenje stabla odlučivanja i na kraju biblioteka matplotlib za grafički prikaz stabla odlučivanja.

```
import pandas as pd
from sklearn.model_selection import train_test_split
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics
import matplotlib.pyplot as pyplot
from sklearn import tree

podaci = pd.read_csv("student-mat.csv", sep=";")
podaci = podaci[["G1", "G2", "G3", "studytime", "failures", "absences"]]

predvidanje = podaci["G3"]

x_trening, x_test, y_trening, y_test = train_test_split(podaci,
predvidanje, test_size=0.3, random_state=0)

stabloOdlucivanja = DecisionTreeClassifier(criterion="entropy",
max_depth=4)

stabloOdlucivanja.fit(x_trening, y_trening)
predvideno = stabloOdlucivanja.predict(x_test)

print("\nPreciznost stabla odlučivanja: ",
metrics.accuracy_score(y_test, predvideno))

fn=['G1', 'G2', 'G3', 'Study time', 'Failures', 'Absences']
cn=['1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13',
'14', '15', '16', '17', '18', '19', '20']
graf, osi = pyplot.subplots(nrows = 1,ncols = 1,figsize = (4,4),
dpi=1200)

tree.plot_tree(stabloOdlucivanja,
               feature_names = fn,
               class_names=cn,
               filled = True)
graf.savefig('stabloOdlucivanja.png')
```

Prvo kao i kod metode linearne regresije razdvojimo podatke u 2 niza i 2 dijela uz pomoć train_test_split funkcije gdje 70 % podataka ide pod varijable za trening, a 30 % pod varijable za testiranje metode. Nakon podjele podataka u varijablu stabloOdlucivanja spremam model za stablo odlučivanja te uz pomoć naredbe fit() učim model na temelju podataka za trening.

Varijabla koja se predviđa se sprema u varijablu predvideno te se pomoću sklearn metrics.accuracy računa preciznost modela.

Zadnja dva odlomka programa služe za kreiranje grafičkog prikaza modela stabla odlučivanja i za spremanje tog grafa. U varijablu fn se spremaju nazivi varijabli koje se koriste u stablu odlučivanja, a u varijablu cn se spremaju moguće izlazne vrijednosti stabla odlučivanja. Pomoću naredbe tree.plot_tree() se kreira graf stabla odlučivanja s nazivima varijabli i izlaznih vrijednosti te pomoću naredbe savefig() se sprema slika grafa stabla odlučivanja.

5.4. Neuronska mreža

Kao i sve metode do sada na početku su uvezene potrebne biblioteke za rad programa. Pandas biblioteka za rad s csv datotekama, sklearn biblioteka za pripremu i izvršavanje modela i random biblioteku za pomoć kod pripreme modela te matplotlib biblioteku za grafički prikaz rezultata neuronske mreže.

```
import pandas as pd
from keras import Sequential
from keras.layers import Dense, Flatten
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as pyplot

podaci = pd.read_csv("student-mat.csv", sep=";")
podaci = podaci[["G1", "G2", "G3", "studytime", "failures", "Medu"]]

predvidanje = podaci["G3"]

df_norm = podaci[['G1', 'G2', 'studytime', 'failures',
'Medu']].apply(lambda x: (x - x.min()) / (x.max() - x.min()))

x_trening, x_test, y_trening, y_test = train_test_split(df_norm,
predvidanje, test_size=0.2)

model = Sequential()
model.add(Dense(17, input_dim=5, activation="relu"))
model.compile(loss="mean_squared_error", optimizer="adam",
```

```
metrics=["accuracy"])
obradaModela = model.fit(x_trening, y_trening, epochs=100,
validation_data=(x_test, y_test))

_, preciznostTrening = model.evaluate(x_trening, y_trening, verbose=0)
_, preciznostTest = model.evaluate(x_test, y_test, verbose=0)
print("Preciznost trening podataka: ", preciznostTrening)
print("Preciznost testnih podataka: ", preciznostTest)

pyplot.subplot(211)
pyplot.title('Gubitak')
pyplot.plot(obradaModela.history['loss'], label='train')
pyplot.plot(obradaModela.history['val_loss'], label='test')
pyplot.legend()
pyplot.subplot(212)
pyplot.title('Preciznost')
pyplot.plot(obradaModela.history['accuracy'], label='train')
pyplot.plot(obradaModela.history['val_accuracy'], label='test')
pyplot.legend()
pyplot.show()
```

Prvo se uvezu željeni podaci iz csv dokumenta te se nakon toga napravi normalizacija vrijednosti koje se spreme u varijablu df_norm tako da se u toj varijabli nalaze samo vrijednosti između 0 i 1. Pomoću funkcije train_test_split razdvajam podatke na testne i podatke za trening gdje 20 % podataka ide na testne podatke, a ostalih 80 % na podatke za trening modela. Model neuronske mreže se kreira naredbom Sequential(). Čvorovi se dodaju na model funkcijom Dense() unutar koje parametar input_dim označuje koliko ima ulaznih čvorova. Naredba compile() izvršava model prema zadanim parametrima. Učenje modela se radi naredbom fit() koja izvršava model neuronske mreže nad podacima za učenje onoliko puta kolika je vrijednost parametra epochs što znači da se u ovom primjeru to provodi 100 puta.

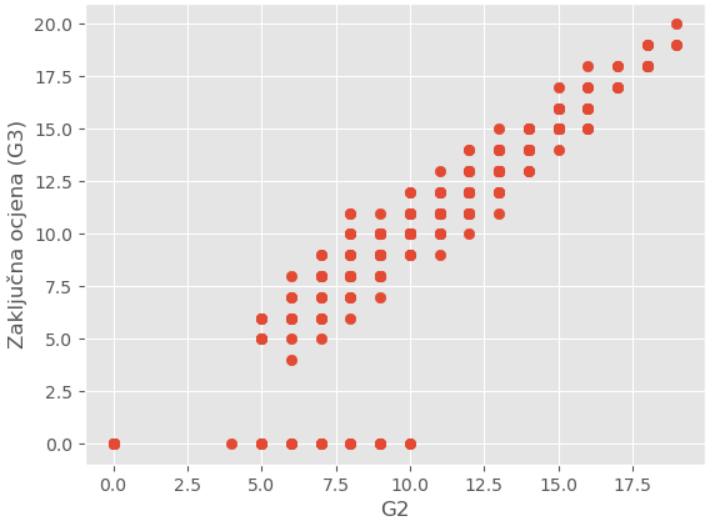
U varijablama preciznostTrening i preciznostTest se sprema preciznost predviđanja podataka za trening i testnih podataka tako da se evaluira model neuronske mreže. Zadnji odlomak programskog koda služi za grafički prikaz promjene koliki je gubitak preciznosti podataka i koliki je postotak preciznosti kroz provedbe učenja neuronske mreže.

6. Pregled rezultata metoda strojnog učenja

Programi napisani u Pythonu su provedeni nad 2 skupa atributa, prvi skup atributa je povezan uz studentove rezultate na nastavi i prisutnost na nastavi, a drugi skup podataka je vezan uz obrazovanje roditelja, obiteljske odnose i zdravlje te izlaske. U prvom skupu atributa su atributi: G1, G2, vrijeme učenja, broj dosadašnjih padova predmeta i broj izostanaka u školi, a u drugom skupu atributa su atributi: obrazovanje majke, obrazovanje oca, kvaliteta obiteljskih odnosa, zdravlje i izlasci s prijateljima te naravno svaki skup podataka ima atribut G3 koji se i predviđa.

6.1. Skup atributa studentove uspješnosti i dolazaka na nastavu

Za prvi skup podataka metoda linearne regresije ima preciznost od 80.25 % (slika 3) te graf linearne regresije ima grupirane podatke blizu regresijske linije (slika 2) što i dokazuje veliku preciznost metode.



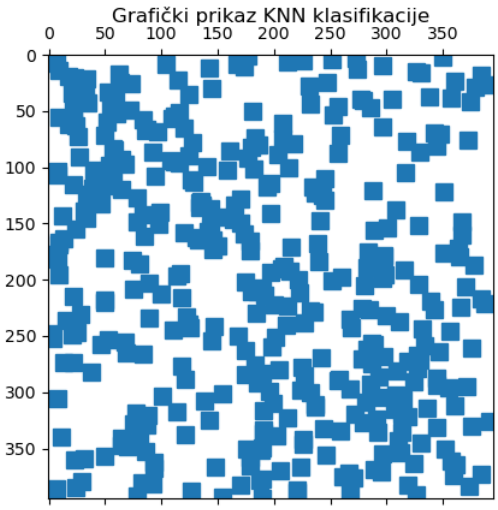
Slika 2 Graf linearne regresije prvog skupa podataka (vlastita izrada)

```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/LinearnaRegresija.py
Preciznost predviđanja podataka:  0.8024812497760324

Process finished with exit code 0
```

Slika 3 Preciznost metode linearne regresije za prvi skup podataka (vlastita izrada)

Metoda k-najbliži susjedi ima vrlo rasprostranjene točke na grafu (slika 4), no vidljivo je nekoliko grupiranih skupova podataka što možemo vidjeti i u postotku preciznosti podataka od 92.41 % i malom srednjom kvadratnom pogreškom od samo 0.75 (slika 5).



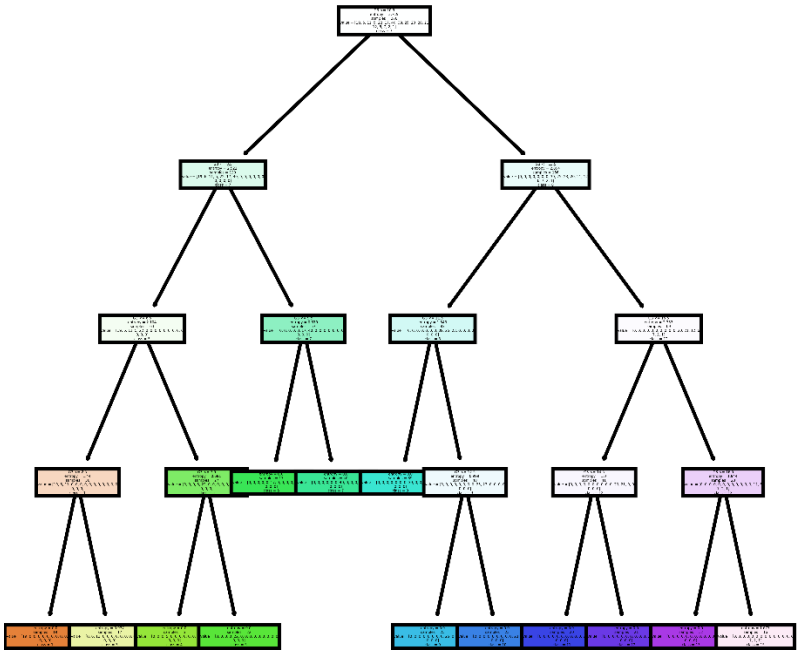
Slika 4 Grafički prikaz KNN klasifikacije prvog skupa podataka (vlastita izrada)

```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/KNajbliziSusjedi.py
Srednja kvadratna pogreška iznosi  0.7493670886075949 , a preciznost iznosi  0.9240506329113924

Process finished with exit code 0
```

Slika 5 Srednja kvadratna pogreška i preciznost KNN metode za prvi skup podataka (vlastita izrada)

Stablo odlučivanja ima najveći postotak preciznosti od svih algoritama s 95.80% (slika 7). Prema stablu odlučivanja (slika 6) najveći broj studenata pripada skupini koja ima vrijednost G3 7 i 8.



Slika 6 Grafički prikaz stabla odlučivanja prvog skupa podataka (vlastita izrada)

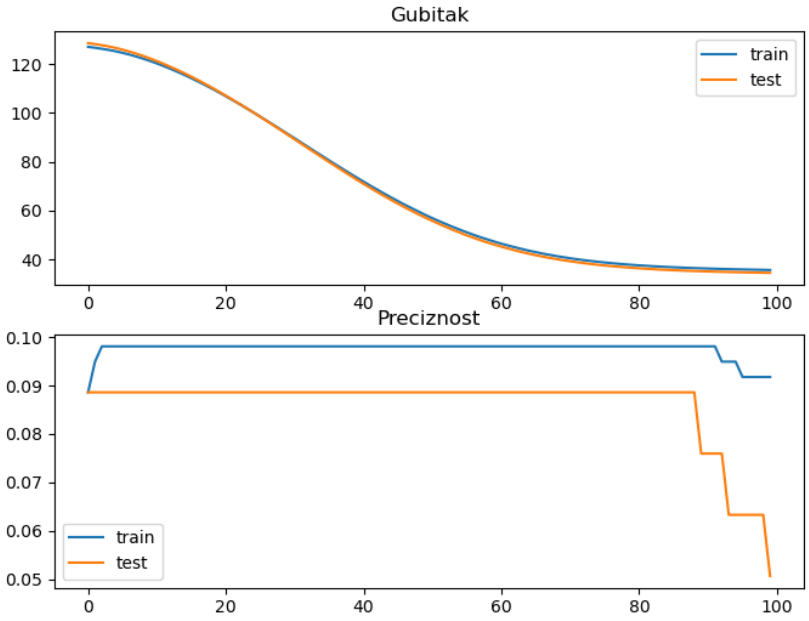
```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/StabloOdlucivanja.py

Preciznost stabla odlučivanja:  0.957983193277311

Process finished with exit code 0
```

Slika 7 Preciznost stabla odlučivanja za prvi skup podataka (vlastita izrada)

Za metodu neuronske mreže sam dobio iznenađujuće rezultate gdje je ona imala najmanju preciznost od svih metoda s preciznošću od 5.06 % (slika 9). Na slici 8 se može vidjeti graf gubitka i preciznosti kroz epohe učenja neuronske mreže, povećanjem broja epoha se smanjuje gubitak, ali se može primijetiti i pad preciznosti i kod testnih podataka i kod podataka za trening.



Slika 8 Graf gubitka i preciznosti kroz epohe neuronske mreže za prvi skup podataka (vlastita izrada)

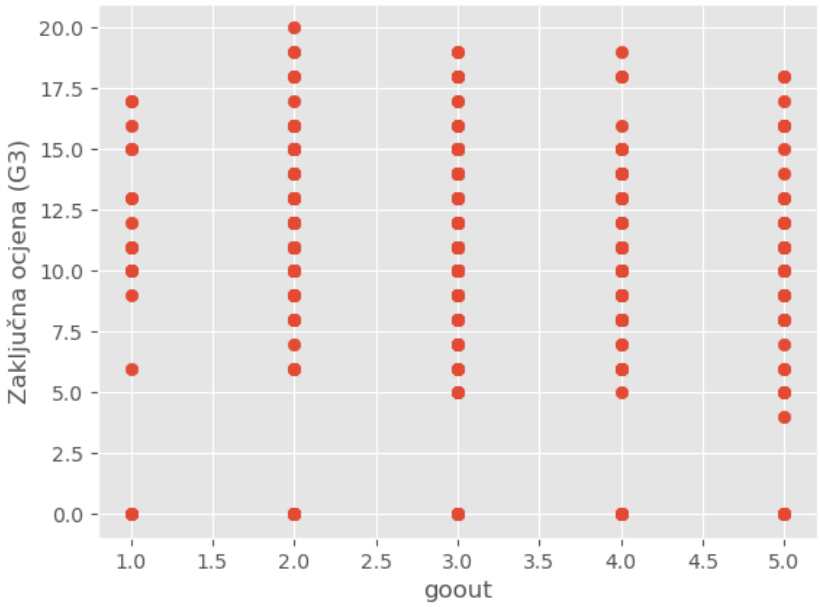
```
Preciznost trening podataka:  0.08860759437084198
Preciznost testnih podataka:  0.050632912665605545

Process finished with exit code 0
```

Slika 9 Preciznost neuronske mreže za prvi skup podataka (vlastita izrada)

6.2. Skup atributa studentovog zdravlja, roditelja i izlazaka

Linearna regresija nad skupom podataka gdje nisu uzeti podaci s velikom korelacijom ima jako lošu preciznost od 5.41 % (slika 11). Ovako niska preciznost je vidljiva i na grafu na slici 10 gdje je jako velika raspršenost instanci od regresijske linije.



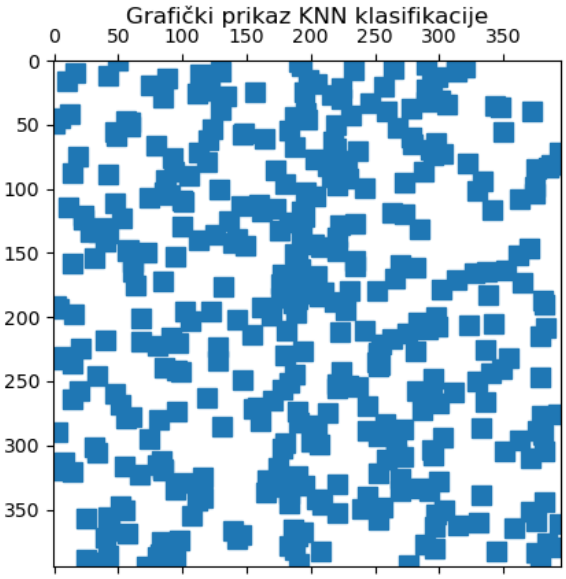
Slika 10 Graf linearne regresije drugog skupa podataka (vlastita izrada)

```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/LinearnaRegresija.py
Preciznost predviđanja podataka:  0.054102853488432245

Process finished with exit code 0
```

Slika 11 Preciznost metode linearne regresije za drugi skup podataka (vlastita izrada)

Metoda k-najbližih susjeda je za ovaj skup podataka imala drugu najveću preciznost od 74.68 % i poprilično visoku srednju kvadratnu pogrešku od 12.27 (slika 13), ali ova metoda je imala puno veću preciznost i nisku srednju kvadratnu pogrešku za prvi skup podataka. Graf k-najbližih susjeda nam i objašnjava zašto je smanjena preciznost gdje je puno veća raspršenost podataka i manji broj velikih skupova.



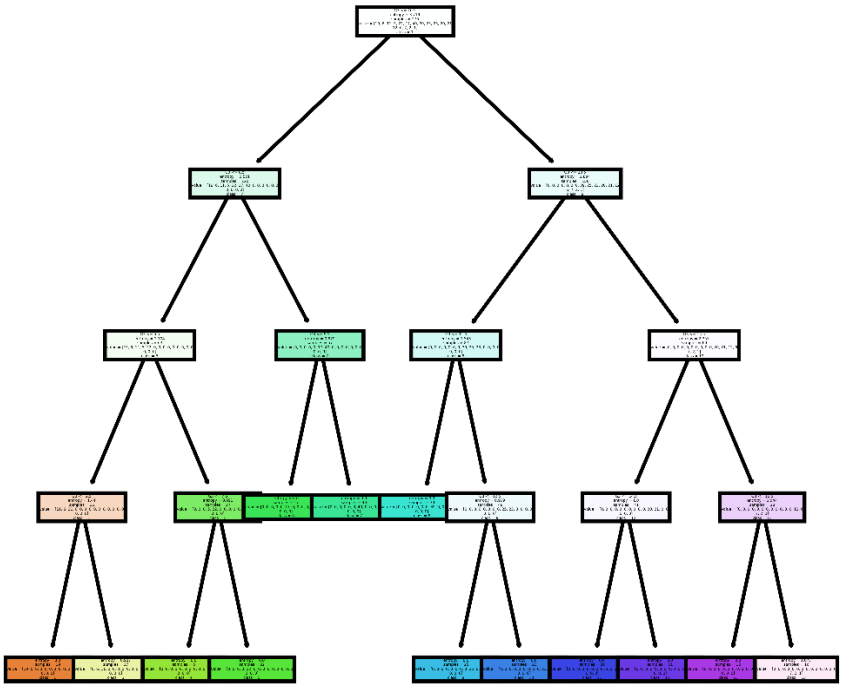
Slika 12 Grafički prikaz KNN klasifikacije drugog skupa podataka (vlastita izrada)

```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/KNajbliziSusjedi.py
Srednja kvadratna pogreška iznosi  12.268354430379746 , a preciznost iznosi  0.7468354430379747

Process finished with exit code 0
```

Slika 13 Srednja kvadratna pogreška i preciznost KNN metode za drugi skup podataka (vlastita izrada)

Metoda stabla odlučivanja kao i u slučaju prvog skupa podataka ponovno ima najveću preciznost koja iznosi 95.80 % (slika 15). Poput i stabla odlučivanja za prvi skup podataka i ovo stablo odlučivanja ima najviše instanci za vrijednost varijable G3 7 i 8.



Slika 14 Grafički prikaz stabla odlučivanja drugog skupa podataka (vlastita izrada)

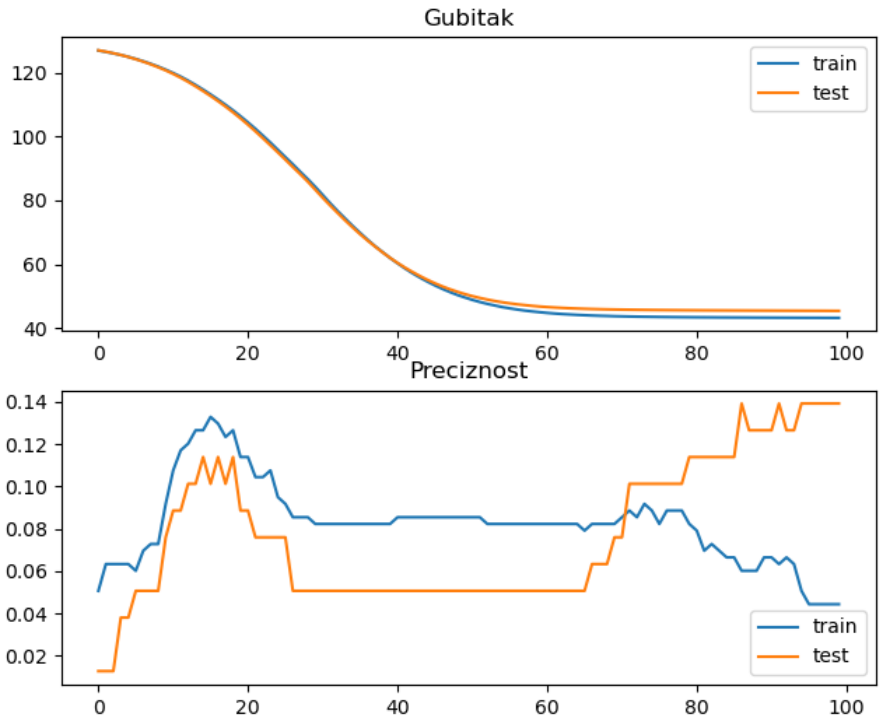
```
C:\Users\Ivan\PycharmProjects\ZavrsniRad\venv\Scripts\python.exe C:/Users/Ivan/PycharmProjects/ZavrsniRad/StabloOdlucivanja.py

Preciznost stabla odlučivanja:  0.957983193277311

Process finished with exit code 0
```

Slika 15 Preciznost stabla odlučivanja za drugi skup podataka (vlastita izrada)

Metoda neuronske mreže za drugi skup podataka ima veću preciznost za razliku od prvog skupa podataka od 13.92% (slika 17) te je vidljivo na grafu gubitka i preciznosti (slika 16) da se u ovom slučaju preciznost podatak povećava s većim brojem epoha i da gubitak pada.



Slika 16 Graf gubitka i preciznosti kroz epohe neuronske mreže za drugi skup podataka (vlastita izrada)

```
Preciznost trening podataka:  0.04113924130797386
Preciznost testnih podataka:  0.13924050331115723

Process finished with exit code 0
```

Slika 17 Preciznost neuronske mreže za drugi skup podataka (vlastita izrada)

7. Zaključak

Prikazao sam u teorijskom dijelu rada analizom znanstvenih članaka kako strojno učenje ima velike koristi na području obrazovanja. Koristi se kako bi se profesori mogli prilagoditi posebnim potrebama nekih studenata ili kako bi se moglo prepoznati studente kojima ne ide najbolje i potrebna im je dodatna pomoć kako ne bi odustali od fakulteta. Korištenje strojnog učenja daje poprilično veliku preciznost i sve više se koristi na području obrazovanja.

Praktični dio rada se sastoji od 4 programa koja sam izradio u programskom okruženju PyCharm koristeći programski jezik Python. Prikazao sam kako nije potrebno veliko znanje za izradu ovih programa i da su to poprilično kratki kodovi. Dobio sam neočekivane rezultate za metodu neuronske mreže koja je imala vrlo malu preciznost, također sam i očekivano dobio vrlo visoku preciznost za metodu stabla odlučivanja budući da je to poprilično dobra metoda klasifikacija kao što je slučaj za ovaj skup podataka gdje su se studenti grupirali prema zaključnoj ocjeni. Tako je i metoda k-najbližih susjeda imala visoki postotak preciznosti kod oba skupa podataka. Metoda linearne regresije je imala na drugom skupu podataka jako niski postotak preciznosti što se može i vidjeti prema grafu linearne regresije za taj skup podataka.

Jednostavnost izrade ovakvih programa i visoka razvijenost biblioteka za strojno učenje omogućava laku izradu vrlo preciznih programa za strojno učenje. Strojno učenje se počinje sve više koristiti i u obrazovanju i općenito, baš zbog toga što je vrlo jednostavno da se napravi takav program i mogu se vidjeti velike koristi od ovakvih programa.

Popis literature

1. Adekitan, A. I., i Salau, O. (2019.). The impact of engineering students' performance in the first three years on their graduation result using educational data mining. *Heliyon*, *5*(2), e01250.
2. Aldowah, H., Al-Samarraie, H., i Fauzy, W. M. (2019.). Educational data mining and learning analytics for 21st century higher education: A review and synthesis. *Telematics and Informatics*, *37*, 13-49.
3. Asif, R., Merceron, A., Ali, S. A., i Haider, N. G. (2017.). Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, *113*, 177-194.
4. Burgos, C., Campanario, M. L., de la Peña, D., Lara, J. A., Lizcano, D., i Martínez, M. A. (2018.). Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout. *Computers & Electrical Engineering*, *66*, 541-556.
5. Divjak, B., i Oreski, D. (2009.). Prediction of academic performance using discriminant analysis. In *Proceedings of the ITI 2009 31st International Conference on Information Technology Interfaces* (pp. 225-230). IEEE.
6. Fernandes, E., Holanda, M., Victorino, M., Borges, V., Carvalho, R., i Van Erven, G. (2019.). Educational data mining: Predictive analysis of academic performance of public school students in the capital of Brazil. *Journal of Business Research*, *94*, 335-343.
7. Gray, C. C., i Perkins, D. (2019.). Utilizing early engagement and machine learning to predict student outcomes. *Computers & Education*, *131*, 22-32.
8. Kovač, R., i Oreški, D. (2018.). Educational data driven decision making: early identification of students at risk by means of machine learning. In *Central European Conference on Information and Intelligent Systems* (pp. 231-237). Fakultet organizacije i informatike Varaždin.
9. Larose, D. T., i Larose, D. C. (2015.). *Data mining and predictive analytics*. John Wiley & Sons.
10. Miguéis, V. L., Freitas, A., Garcia, P. J., i Silva, A. (2018.). Early segmentation of students according to their academic performance: A predictive modelling approach. *Decision Support Systems*, *115*, 36-51.
11. Oreški, D., i Hajdin, G. (2019.). A Comparative Study of Machine Learning Approaches on Learning Management System Data. In *2019 International Conference on Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO)* (str. 136-141). IEEE.
12. Oreski, D., Pihir, I., i Konecki, M. (2017.). Crisp-DM process model in educational setting. *Economic and Social Development: Book of Proceedings*, 19-28.
13. Oreški, D., Hajdin, G., i Klicek, B. (2016.). Role of personal factors in academic success and dropout of IT students: Evidence from students and alumni. *TEM Journal*, *5*(3), 371.
14. Sarkar, D., Bali, R., i Sharma, T. (2018.). Practical machine learning with Python. *A problem-solvers guide to building real-world intelligent systems. Apress, Berkely*.
15. Shahiri, A. M., Husain, W., i Rashid, N. A. (2015.). 'A review on predicting student's performance using data mining techniques,''Procedia Comput. *Sci, 72*, 414.

16. Xu, X., Wang, J., Peng, H., i Wu, R. (2019.). Prediction of academic performance associated with internet usage behaviors using machine learning algorithms. *Computers in Human Behavior*, *98*, 166-173.

Popis slika

Popis tablica