Uporaba metoda dubinske analize podataka za potrebe upravljanja rizikom u sportskom klađenju

Šutalo, Ana

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 319

UPORABA METODA DUBINSKE ANALIZE PODATAKA ZA POTREBE UPRAVLJANJA RIZIKOM U SPORTSKOM KLAĐENJU

Ana Šutalo

Zagreb, veljača 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 319

UPORABA METODA DUBINSKE ANALIZE PODATAKA ZA POTREBE UPRAVLJANJA RIZIKOM U SPORTSKOM KLAĐENJU

Ana Šutalo

Zagreb, veljača 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 2. listopada 2023.

DIPLOMSKI ZADATAK br. 319

Pristupnica:	**Ana Šutalo (0036507787)**
Studij:	Računarstvo
Profil:	Znanost o podacima
Mentor:	prof. dr. sc. Damir Pintar

Zadatak: **Uporaba metoda dubinske analize podataka za potrebe upravljanja rizikom u sportskom klađenju**

Opis zadatka:

Upravljanje rizikom je praksa identificiranja, procjene i kategoriziranja potencijalnih rizika povezanih s kladioničarskom aktivnošću ili portfeljem. Upravljanje rizikom uključuje analizu podataka i trendova povezanih s određenim događajem ili igrom koji bi mogli utjecati na ishode, kao i razvoj strategija koje su usmjerene na smanjenje potencijalnih gubitaka. Jedan od izazova s kojima se kladioničarske kuće susreću jest pokušaj ostvarivanja profita kroz zlouporabu sustava; metode dubinske analize podataka potencijalno mogu otkriti određene uzorke povezane s takvim aktivnostima te automatski odbiti provedbu klađenja za oklade koje se procijene rizičnima. Zadatak rada je proučiti i detektirati potencijalno rizične sportske listiće. U tu svrhu potrebno je na temelju dostupnih podataka razviti prediktivni model koji će uspješno razlikovati rizične listiće od ostalih i provesti evaluaciju efikasnosti korištenih modela kako bi se utvrdila njihova pouzdanost u praktičnoj primjeni. Konačno rješenje bilo bi realizirano u obliku programske skripte koja bi demonstrirala učinkovitost odabrane metode upravljanja rizikom kroz implementaciju i validaciju nad stvarnim podacima iz domene sportskog klađenja.

Rok za predaju rada: 9. veljače 2024.

Sadržaj

1. Uvod

Upravljanje rizikom ključno je za uspjeh u mnogim industrijama, pa tako i u sportskom klađenju. Rizik se promatra iz perspektive poduzeća, odnosno kladionice. Upravljanje njime potrebno je za održavanje stabilnosti poslovanja, a uzmu li se u obzir nepredvidivost sportskih događaja i složenost samog klađenja, takve je probleme jako teško rješavati ručno.

Ovaj rad istražuje primjenu metoda dubinske analize podataka kako bi se uspješno klasificirali rizični sportski listići i igračima onemogućila uplata istih kako bi se minimizirali gubitci u poslovanju. Takvi su događaji rijetki, što dodatno otežava problem njihova prepoznavanja. Cilj je rada pokazati kako se korištenjem raznih metoda dubinske analize podataka može automatizirati proces odbijanja rizičnih listića te kako pojedine metode utječu na njihovu klasifikaciju.

U drugom je poglavlju dan uvod u sportsko klađenje, definirani su ključni pojmovi i opisani mogući rizici. Treće poglavlje donosi pregled koraka u procesu dubinske analize podataka, a četvrto poglavlje opisuje problem nebalansiranosti ciljne klase i pristupe njegovu rješavanju. Konkretan problem klasifikacije sportskih listića kojim se rad bavi i korištene metode opisani su u petom poglavlju, a dobiveni rezultati u šestom.

2. Analiza podataka u domeni sportskog klađenja

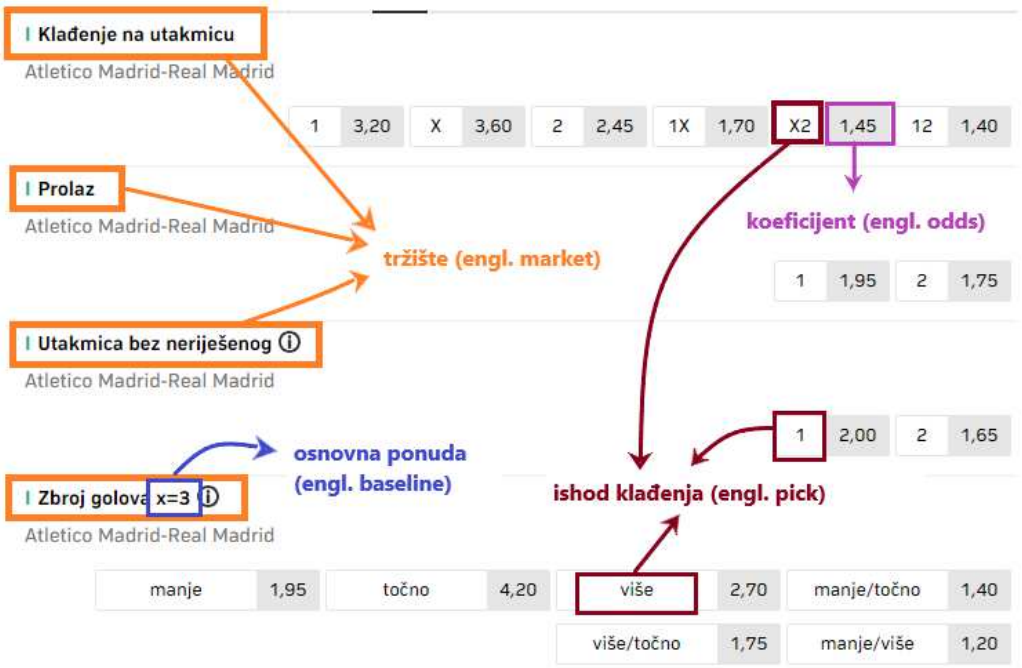
2.1. Sportsko klađenje

Klađenje kao takvo postoji odavno, stoga ne čudi da razvojem online platformi postaje sve popularnije zbog velike dostupnosti. Online kladionice omogućuju pojedincima klađenje iz udobnosti vlastitog doma, u bilo koje doba dana. Proces klađenja uključuje nekoliko sudionika:

- kladioničari (kladioničarske kuće, engl. *bookmakers*) — tvrtke koje omogućuju klađenje: sastavljaju ponude ishoda, određuju koeficijente, pružaju platforme za klađenje, isplaćuju dobitak kladiteljima, …
- kladitelj (dalje u tekstu: igrač) — osoba koja se kladi, ulaže novac na neki mogući ishod kako bi ostvario profit u slučaju ostvarenja njegove oklade
- regulatorna tijela — nadziru procese klađenja, sprečavaju manipulaciju rezultata, postavljaju ograničenja i provode zakone vezane uz klađenje

Ovaj rad vezan je isključivo uz sportsko klađenje u kojemu igrači ulažu novac i pokušavaju predvidjeti razne ishode sportskih događaja kako bi ostvarili profit. Mnoge platforme za klađenje imaju ponudu oklada na velikom broju sportova, a za svaki sport postoje različita tržišta (engl. *market*) koja sadrže slične, međusobno isključive oklade, npr. broj golova u utakmici. Slika 2.1. prikazuje samo nekoliko od mnogih primjera mogućih oklada na nogometnoj utakmici.

Oklada je definirana odabirom jednog od ishoda klađenja (engl. *pick*) na tržištu. Neka tržišta dodatno su definirana osnovnom ponudom (engl. *baseline*), kao što je na slici 2.1. slučaj s tržištem Zbroj golova. Osnovna ponuda postavlja se ovisno o utakmici



Slika 2.1. Dijelovi oklade na sportski događaj

— razlikuje se za različite utakmice: ovisno o međusobnom omjeru snaga protivnika, razini natjecanja i sl. Za svaki mogući ishod klađenja u ponudi, kladioničari definiraju pripadni koeficijent (engl. *odds*), uzimajući u obzir vjerojatnost svakog ishoda. Budući da je jako teško dobro procijeniti vjerojatnosti ishoda, kladioničari ne dijele svoje računice i nerijetko različite kladioničarske kuće nude različite koeficijente za iste ishode. Oni ovise o brojnim faktorima, kao što su statistike natjecatelja u prethodnim utakmicama, prednost domaćeg terena, stalnost rezultata pojedinca/ekipe (npr. je li ekipa u pobjedničkom nizu), ozljeda natjecatelja, širina sastava, faza natjecanja, itd. Koeficijent, zajedno s količinom uplaćenog novca, određuje isplatu osobi koja se kladila u slučaju ostvarenja ishoda oklade: $isplata = uplata * koeficijent$. Pri određivanju koeficijenata kladioničarima je bitno održati ravnotežu između svoje zarade i primamljivosti koeficijenata igračima. Ukoliko kladionice postave previše nepoštene koeficijente, riskiraju gubitak igrača na platformi.

Oklade se međusobno razlikuju po broju mogućih ishoda koje ju definiraju [1], a neke od najčešćih su:

- dva moguća ishoda događaja — klađenje je isključivo na jedan ili drugi ishod (npr.

pobjednik teniskog meča)

- tri moguća ishoda — moguće se kladiti na jedan od tri ishoda ili u nekim slučajevima na dva simultana ishoda (npr. pobjeda ili neriješeno za jednu od ekipa u nogometnoj utakmici)

- više mogućih ishoda — npr. točan rezultat, ukupan broj žutih kartona, rezultat 1. gema, itd.

- pobjednik natjecanja — igrač može odabrati jednog od natjecatelja za kojega predviđa pobjedu na određenom natjecanju (npr. Hrvatska će osvojiti Europsko prvenstvo u vaterpolu)

Kako bi svaki igrač pronašao opciju koja mu najviše odgovara, kladionički često nude različite vrste klađenja. Tipovi klađenja mogu se podijeliti na dva načina: prema vremenu uplate i prema broju događaja na koje se igrač kladi.

Prema vremenu uplate klađenje se dijeli na:

- **Klađenje prije početka meča** (engl. *prematch*) "klasično" je klađenje, u kojemu oklada može biti zaključena isključivo prije početka događaja. Koeficijenti se u takvim ponudama relativno rijetko mijenjaju jer ne uzimaju u obzir tijek utakmice ili natjecanja, već se vjerojatnosti ishoda računaju najvećim dijelom na temelju povijesnih podataka. Tu je najčešće ponuda oklada najveća upravo zbog količine dostupnih podataka za određivanje koeficijenata i manje nepredvidivosti u odnosu na klađenje uživo.

- **Klađenje uživo** (engl. *live*) ne ograničava igrača na postavljanje oklada prije početka utakmice. Postoje tržišta na kojima se mogu stavljati oklade tijekom meča i najčešće su vezana za tijek ogleda: npr. tko će osvojiti sljedeći set, točan rezultat na poluvremenu, koji sastav će prvi dobiti isključenje, ...

- **Kombinirano klađenje** (engl. *mix*) je kombinacija oklada na listiću koja sadrži barem po jednu okladu prije početka meča i uživo.

S obzirom na broj događaja na koje se igrač kladi, razlikuje se nekoliko vrsta klađenja:

- **Pojedinačno**: najjednostavniji oblik klađenja — igrač ulaže novac na jednu okladu, tj. odabire samo jedan ishod na jednom događaju. Primjer takve oklade je uplata na pobjedu Real Madrida u utakmici protiv Liverpoola u Ligi prvaka.

- **Zbirno**: igrač želi predvidjeti više ishoda na različitim događajima (utakmicama). Nije moguće kladiti se na više ishoda unutar istog događaja, a pojedini događaji moraju biti međusobno nezavisni kako bi se mogla staviti oklada. Ovisno o kladioničarskoj kući i/ili tržištu na kojem posluje, određuje se maksimalan broj oklada koje mogu biti stavljene na jednu uplatu, odnosno listić (engl. *ticket*). Listić je dobitan ako su svi ishodi na njemu pogođeni. Tada kladionica računa isplatu igraču na sljedeći način:

$$isplata = uplata \cdot \prod_{b=1}^{B} coef_b \tag{2.1}$$

  pri čemu je $B > 1$ ukupan broj oklada koje se nalaze na listiću, a $coef_b$ koeficijenti pojedinačnih oklada. Zbog nezavisnosti događaja, ukupan koeficijent odgovara umnošku svih pojedinih. Vjerojatnost isplate listića obrnuto je proporcionalna broju ishoda na listiću:

$$P_{isplata} = \frac{1}{2^B} \tag{2.2}$$

  Ovakva je vrsta klađenja za igrača više rizična - što više ishoda odabere, veća je vjerojatnost da se barem jedan od njih neće dogoditi, a samim time da cijeli listić ne ostvaruje dobitak. Međutim, kao i kod svakog ulaganja, rizik je proporcionalan potencijalnom dobitku, što ovakvo klađenje čini atraktivnijim dijelu igrača.

- **Sistemsko**: tip klađenja najsličniji zbirnom klađenju — igrač može odabrati više nezavisnih događaja čije ishode želi predvidjeti. Sistemski listić razlikuje se od zbirnog po tome što se ne moraju svi ishodi ostvariti da bi isti bio dobitan i ostvario profit igraču. Ako je sistemski listić definiran kao a/b, pri čemu vrijedi $a < b$, $a, b \in \mathbb{N}$, igrač je odabrao ukupno b ishoda, a potrebno ih je minimalno a ostvarenih da bi igraču bio isplaćen neki iznos. Dakle, za isplatu listića igraču mogu se prebrojati moguće kombinacije, označene s P:

$$P = \binom{b}{b} + \binom{b}{b-1} + \cdots + \binom{b}{a} \tag{2.3}$$

Broj svih kombinacija ishoda označen je s T, a iznosi:

$$T = 2^b \tag{2.4}$$

Vjerojatnost isplate sistemskog listića računa se kao omjer mogućih i svih kombinacija ishoda:

$$P_{isplata} = \frac{P}{T} \tag{2.5}$$

Ona to veća, što ima više mogućih kombinacija koje mogu biti dobitne. To se ostvaruje kada je a što manji u odnosu na b. Neki primjeri sistemskih listića, zajedno s pripadnim vjerojatnostima isplate prikazani su ispod:

oznaka listića	vjerojatnost isplate
2/3	$4/8 = 0.5$
2/4	$11/16 = 0.3125$
3/4	$5/16 = 0.6875$
3/5	$16/32 = 0.5$

Tablica 2.1. Primjeri sistemskih listića

Ovakva vrsta klađenja manje je rizična za igrače jer im omogućava eventualne pogreške (neostvarivanje ishoda) pri klađenju, uz ostvarivanje dobiti. Iznos isplate ovisi o koeficijentima ostvarenih ishoda i u njih je "ugrađena" vjerojatnost ostvarivanja svakog ishoda. U cilju smanjenja svog rizika, kladionice nude niže koeficijente za sistemsko klađenje nego za zbirno jer je veća vjerojatnost da će morati isplatiti neki iznos igraču u odnosu na zbirno klađenje.

- **BetBuilder**: omogućava postavljanje oklade na više ishoda unutar istog događaja (utakmice/meča) [2]. Ishodi su međusobno zavisni, stoga koeficijent oklade nije poznat unaprijed, nego se određuje na temelju zajedničke vjerojatnosti ostvarivanja odabranih ishoda. Nije moguće kladiti se na međusobno potpuno isključive događaje — npr. pobjeda ekipe A i ekipa A neće zabiti gol. Budući da je računanje koeficijenata za ovakvo klađenje dosta zahtjevno (izazovno), ne nude ga sve kladionice, a one koje ga nude to rade samo za najpopularnije sportove.

2.2. Rizici u sportskom klađenju

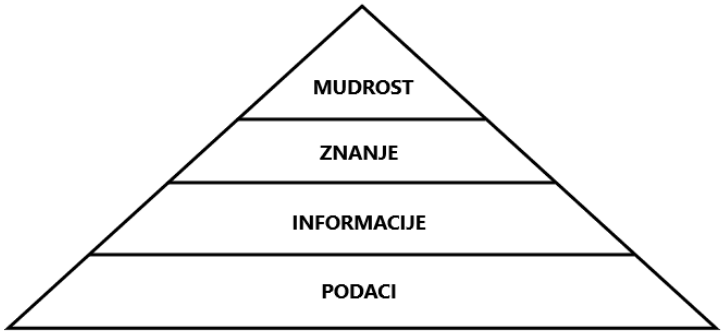
Rizik je definiran kao mogućnost da se pri ostvarenju određenoga cilja, on ne ostvari djelomično ili potpuno[3]. U procesu klađenja, igrači ulažu svoj novac kako bi ostvarili dobitak. Njihov je cilj ostvarivanje događaja na koji su uložili novac i posljedično ostvarivanje profita, a pri tom riskiraju izgubiti uloženi novac. Profit je proporcionalan koeficijentu određenom za odabrani ishod, odnosno obrnuto proporcionalan vjerojatnosti ostvarivanja ishoda. Kladionice se mogu promatrati kao poduzeća pa je iz njihove perspektive cilj ostvarivanje profita. Kako bi to ostvarile, koeficijenti koje nude ne odgovaraju stvarnim vjerojatnostima, nego je u njih uključen i profit.

Za kladionicu je ključno pronaći ravnotežu između količine marže (profita kladionice) i povoljnih koeficijenata za igrače kako ne bi svoj novac uložili u konkurentnim kladionicama. Visoki koeficijenti pridjeljuju se ishodima s malim vjerojatnostima. Na prvi pogled čini se da je bolje da igrači što više uplaćuju na malo vjerojatne ishode, ali oni su zapravo više rizični za kladionice. Takvi ishodi se rijetko ostvaruju, ali kada se ostvare predstavljaju velik trošak za kladioničarsku kuću.

3. Metode dubinske analize podataka

3.1. Što je dubinska analiza podataka?

Podaci u današnjem svijetu mogu biti jako vrijedni. Prikupljaju se, dijele, čak i prodaju — gotovo kao da su neka sirovina. Kao i svaku sirovinu, tako i podatke treba pripremiti i obraditi kako bi bili korisni. U područjima informacijske znanosti i upravljanja znanjem definiran je koncept piramide znanja, odnosno hijerarhije između podataka, informacije, znanja i mudrosti (engl. *Data-Information-Knowledge-Wisdom (DIKW) hierarchy*). Iako postoje različite definicije tih pojmova[4], hijerarhija među njima uvijek je ista, kao na slici 3.1. Svaki pojam uključuje preostale pojmove koji se nalaze ispod njega — da se dođe do znanja i mudrosti, potrebno je prvo skupiti podatke. Zatim od njih pokušati dobiti korisne informacije koje će dati neka nova saznanja, a sve to kako bi se pospješio proces odlučivanja (mudrost). U tu svrhu koriste se metode dubinske analize podataka.



Slika 3.1. Hijerarhija DIKW

Dubinska analiza podataka uključuje niz tehnika i algoritama koji se koriste za dobivanje korisnih informacija[5] i uočavanje uzoraka iz velikih skupova podataka. Ukoliko postoje, uzorci u podacima korisni su za predviđanje ponašanja novih primjera poda-

taka. Uzorci se mogu iskazati na dva oprečna načina: kao crna (uzorci nisu eksplicitni i razumljivi, ali mogu se modelirati) ili bijela kutija (uzorci se mogu prikazati nekom strukturom koju čovjek može lako razumjeti)[6].

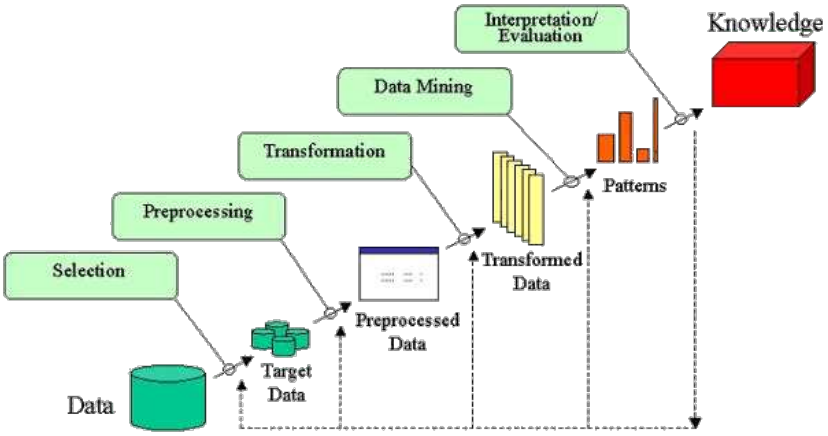
Osnovni su ciljevi dubinske analize podataka opisivanje i učenje o zadanom skupu podataka ili predviđanje ponašanja novih podataka primjenom algoritama strojnog učenja[7]. Naglasak nije na samim algoritmima strojnog učenja, nego na njihovoj učinkovitoj primjeni u raznim domenama. U usporedbi s ljudima, takav automatiziran pristup izrazito je brz, konzistentan i u jednom smislu nepogrešiv: ako se utvrdi da model radi dobro, bit će dosljedan i raditi dobro za bilo koji broj novih primjera[8]. Funkcija i svrha alata za modeliranje zapravo je transformirati znanje sadržano u određenom skupu podataka u oblik koji je koristan ili razumljiv ljudima. Može biti i korisno i razumljivo, ali nije nužno tako.

Različite industrije primjenjuju metode dubinske analize podataka, stoga je bilo potrebno razviti univerzalni model procesa dubinske analize podataka kako bi se smanjila mogućnost grešaka u procesu. Ne postoji jedinstveni ispravan način izvedbe analize podataka, ali su se kroz godine razvili modeli koji pomažu u planiranju i izvođenju takvih projekata — svi imaju isti cilj, a razlikuju se po nazivlju i opsegu pojedinih koraka. Prateći razvoj modela vezanih uz otkrivanje znanja i dubinsku analizu podataka, prema [9] mogu se izdvojiti tri glavna modela:

1. KDD (engl. *Knowledge Discovery in Databases*) prvi je model procesa dubinske analize podataka sa strukturiranim koracima. To je iterativni proces koji pokušava otkriti korisne uzorke i znanje u kontekstu velikih baza podataka. KDD proces prikazan je na slici 3.2. i obuhvaća pet faza: odabir podataka, njihovo čišćenje, transformaciju, dubinsku analizu i evaluaciju/interpretaciju.

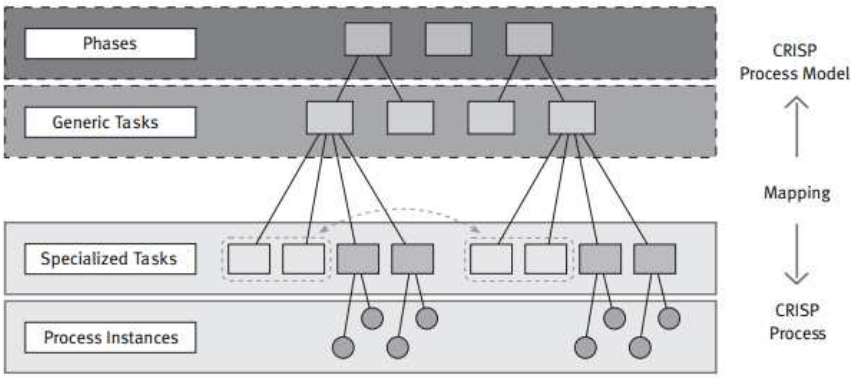
2. SEMMA (engl. *Sample, Explore, Modify, Model, Assess*) razvio je SAS 1997. godine. Naziv modela zapravo je akronim pojedinih faza projekta: uzorkovanje (*sample*), istraživanje (*explore*), prilagodbu (*modify*), modeliranje (*model*) i procjenu (*assess*). Faze su jako slične onima u KDD modelu i programski su podržane alatom SAS Enterprise Miner, stoga ovaj model neki smatraju implementacijom KDD modela[9].

3. CRISP-DM (engl. *CRoss Industry Standard Process for Data Mining*) najpoznatiji



Slika 3.2. KDD model [10]

je i prema [11] najčešće korišten model procesa dubinske analize podataka. Njegove su prednosti što je neovisan o industriji, alatima i metodama, a nedostatak što ne uključuje aktivnosti upravljanja projektom i zahtijeva puno dokumentacije. Njegova hijerarhijska struktura prikazana je na slici 3.3., a sadrži četiri razine (od općenitog do specifičnog): faza, generički zadaci, specijalizirani zadaci i instance procesa [12]. Faze i generički zadaci općeniti su i pripadaju modelu procesa, dok su specijalizirani zadaci i instance procesa konkretni i specifični za svaki projekt, odnosno definiraju sam proces. Proces je podijeljen u 6 faza, koje se po potrebi obavljaju iterativno: razumijevanje poslovnih potreba, razumijevanje podataka, priprema podataka, modeliranje, vrednovanje i puštanje u pogon (engl. *deployment*).



Slika 3.3. Hijerarhija CRISP-DM metodologije [12]

Ovaj rad ne bavi se modelima procesa, niti slijedi bilo koji od navedenih modela. U nastavku daje pregled metoda koje se koriste ili mogu koristiti u projektima dubinske analize podataka i otkrivanja znanja.

Modeli koji se koriste za prediktivne analize najčešće ne mogu koristiti sirove podatke, nego ih je potrebno obraditi prije učenja modela. Cilj je pripreme podataka transformirati početni skup podataka u oblik koji je prikladniji za modeliranje, tako da se skupom podataka što bolje predoči struktura problema prediktivnom modelu. Međutim, kada bi struktura problema bila unaprijed poznata, dubinska analiza podataka ne bi bila potrebna [13]. Priprema podataka specifična je za svaki projekt i ne postoji jedan ispravan način za dobivanje pravog skupa podataka. Ukoliko je početni skup loše kvalitete, iz njega se teško mogu dobiti korisne informacije unatoč provedenim transformacijama podataka (engl. *GIGO - garbage in, garbage out*) [8]. Na početku svakog projekta potrebno je istražiti problem i pokušati osigurati da početni podaci i njihove izvedenice što bolje mogu opisati taj problem.

U [13] definirani su najčešći standardni zadaci koji se pojavljuju tijekom pripreme podataka:

- Čišćenje podataka — identifikacija i ispravak pogrešaka u podacima.
- Odabir značajki — određivanje značajki koje su najznačajnije za rješavanje problema.
- Transformacija značajki — promjena razdiobe ili skale podataka, dobivanje novih značajki.
- Smanjenje dimenzionalnosti — projiciranje podataka u niždimenzionalni prostor, tako da sadrže što više informacije.

S obzirom na to da postoje standardne implementacije brojnih algoritama strojnog učenja u bibliotekama otvorenog koda, učenje modela postalo je rutina. Najizazovniji je dio svakog projekta prediktivnog modeliranja kako pripremiti podatke koji se koriste, upravo zato jer su jedinstveni za svaki projekt. Zbog toga je na pripremu podataka potrebno uložiti najviše vremena u cijelom projektu. Prema istraživanju Forbesa, podatkovni znanstvenici troše čak oko 80% vremena na prikupljanje, organizaciju i manipulaciju podacima [14]. U nastavku poglavlja nalazi se pregled tehnika pripreme podataka i problema na koje se primjenjuju.

3.2. Čišćenje podataka

Čišćenje podataka uključuje uočavanje i rješavanje problema u podacima. Za to je potrebno prvo istražiti dostupne podatke: tipove podataka, odrediti ciljnu značajku (ako postoji), detekciju nedostajućih i stršećih vrijednosti, uklanjanje neinformativnih značajki i duplikata,itd. Razvijeni su alati za automatiziranu eksploratornu analizu podataka, koji mogu ubrzati i olakšati proces čišćenja podataka. Ipak, (još uvijek) nije moguće u potpunosti automatizirati taj proces jer je potrebno primijeniti znanje o problemu na dostupne podatke, i kako učinjene promjene utječu na podatke [8]. U nastavku su opisani najčešći problemi u sirovim podacima i neka primjenjiva rješenja.

KONSTANTNE ZNAČAJKE

Ovakve značajke imaju jednaku vrijednost na cijelom skupu podataka. Njihova je varijanca nula i ne donose nikakvu informaciju korisnu za modeliranje problema. Takve značajke je potrebno ukloniti iz podataka.

Ukoliko postoji značajka s malim brojem vrijednosti koje poprima, ne može se odmah odbaciti, već treba razmotriti prirodu te varijable. Ukoliko je varijabla binarna ili kategorijska, vjerojatno sadrži neku korisnu informaciju i ne treba ju ukloniti. S druge strane, ako je ta značajka po prirodi numerička, a poprima mali broj vrijednosti (ima malu varijancu), može se razmotriti njezino uklanjanje.

MONOTONE ZNAČAJKE

Vrijednosti monotone značajke rastu (ili se smanjuju) bez ograničenja. Najčešće je to značajka koja označava redni broj primjera, datum, identifikacijski broj i sl. Takve značajke ne sadrže korisnu informaciju i uglavnom se mogu ukloniti. U slučaju vremenskih nizova, značajka s datumom može se transformirati u neki oblik pogodniji za analizu.

NEKONZISTENTNI PODACI

U slučaju spajanja podataka iz različitih izvora, može doći do nekonzistentnosti. Na primjer, u jednom izvoru vrijednosti neke značajke mogu biti 0/1, a u drugom DA/NE. Jednom kad se uoče, takvi problemi se lako rješavaju odabirom jedinstvene reprezentacije podataka u cijelom skupu.

STRŠEĆE VRIJEDNOSTI

Stršeći podaci su oni koji jako odudaraju od uobičajenih vrijednosti za jednu ili više značajki. Mogu biti uzrokovani pogreškom u podacima ili odražavati prirodno stanje. Ukoliko podaci strše zbog pogreške, potrebno ih je ukloniti kako bi podaci što bolje reprezentirali problem. Ne postoji univerzalno rješenje za stršeće vrijednosti, ali ih je svakako bitno detektirati u podacima. Najpoznatiji postupci otkrivanja stršećih vrijednosti su:

- vizualizacija podataka: histogram, box-plot;
- statističke metode: z-vrijednost (engl. *z-score*), interkvartilni rang;
- automatsko otkrivanje stršećih vrijednosti algoritmima:
  - LOF (engl. *local outlier factor*) — Postupak se temelji na određivanju ocjene anomalije (engl. *anomaly score*) na temelju lokalnih gustoća točaka i njihovoj usporedbi. Točke s manjom gustoćom imaju veću ocjenu anomalije i smatraju se stršećima [15]. Zbog prokletstva dimenzionalnosti metoda je primjenljiva na podatke s malo značajki.
  - IsolationForest — Algoritam slučajno odabire značajku, a zatim i vrijednost po kojoj razdvaja skup podataka na particije. Postupak je rekurzivan, a može se predstaviti strukturom stabla. Metoda pretpostavlja da se stršeće vrijednosti prije izdvoje iz skupa, odnosno da je prosječan put do njih (od korijena stabla) kraći nego do uobičajenih vrijednosti [16]. Može se koristiti i na visokodimenzionalnim skupovima podataka, ali teško uočava grupirane anomalije i anomalije poravnate s osima.

NEDOSTAJUĆE VRIJEDNOSTI

U stvarnim podacima često nedostaju vrijednosti nekih značajki. To mogu biti vrijednosti koje postoje u stvarnom procesu, ali nisu unesene u skup podataka ili vrijednosti koje su prazne ili nepoznate. Nedostajuće vrijednosti mogu biti predstavljene posebnim tipom podataka (null, None, NaN i sl.) ili nekom neobičnom vrijednosti za tu značajku (npr. -1 za značajku koja poprima pozitivne numeričke vrijednosti; ” ili ’-’ za tekstualne značajke). Najčešće se ne može pretpostaviti uzrok nedostajućim podacima, ali sam obrazac nedostajućih vrijednosti ponekad sadrži informaciju [6]. Problem nedostajućih vrijednosti potrebno je riješiti jer algoritmi koji se koriste za modeliranje ne mogu raditi

s takvim podacima.

Najjednostavniji je pristup uklanjanje primjera s nedostajućim vrijednostima. Ukoliko velik dio neke značajke nedostaje, najbolje je ukloniti cijelu značajku iz skupa podataka, a slično vrijedi i za pojedinačne primjere. Često je pretpostavljena opcija uklanjanje nedostajućih vrijednosti, osobito ako se ne može odrediti postoji li zapravo stvarna informacija o podacima koji nedostaju. Nedostatak ovog pristupa je što se gubi dio podataka, a to je osobit problem za male skupove podataka.

Alternativno je rješenje zamijeniti nedostajuće vrijednosti nekim drugim vrijednostima. Računanjem statistika (aritmetička sredina, medijan, mod) mogu se jednostavno procijeniti i zamijeniti nedostajuće vrijednosti iz preostalih vrijednosti određene značajke. Nedostajuće vrijednosti mogu se zamijeniti i proizvoljnom konstantom. Kompleksnije metode promatraju odnose između različitih značajki i pokušavaju očuvati njihovu zajedničku varijabilnost. Algoritmi strojnog učenja nastoje iz dostupnih podataka predvidjeti vrijednosti onih koji nedostaju — značajka s nedostajućim vrijednostima postaje ciljna za problem imputacije. Najčešće se koriste modeli linearne regresije i k najbližih susjeda (pod uvjetom da su značajke numeričke).

3.3. Transformacija značajki

Ponekad sirovi podaci sadrže korisne informacije, ali nisu eksplicitno iskazane. Ne postoji konkretno pravilo kojim se uvijek mogu dobiti korisne nove značajke. Za to je potrebno razumijevanje problema i dostupnih podataka, malo kreativnosti i isprobavanje različitih pristupa dok se ne dobije zadovoljavajuće rješenje. Kvalitetne značajke bolje reprezentiraju dostupne podatke i početni problem, doprinose pojednostavljenju modela, bržem učenju i fleksibilnosti modela [17].

3.3.1. Pretvorba kategoričkih značajki u numeričke

Modeli ne mogu učiti nad kategoričkim značajkama, stoga ih je potrebno pretvoriti u numeričke. Ovisno o vrsti kategoričke varijable, koriste se različiti pristupi.

Nominalne varijable nemaju definiran prirodni poredak kategorija. U tom slučaju svaka kategorija postaje binarna značajka (engl. *One-Hot Encoding*) — iz jedne značajke

dobivamo onoliko novih koliko je bilo različitih kategorija. Svaki primjer (redak) ima vrijednost 1 u značajki koja se preslikala iz pripadne kategorije. Primjer takvog preslikavanja prikazan je na slici 3.4. — kategorička značajka sadrži tri moguće vrijednosti i nakon enkodiranja nastale su tri nove varijable.

Boja očiju
Zelena
Smeđa
Smeđa
Plava
Smeđa
Plava

Boja_očiju_zelena	Boja_očiju_smeđa	Boja_očiju_plava
1	0	0
0	1	0
0	1	0
0	0	1
0	1	0
0	0	1

Slika 3.4. Primjer preslikavanja nominalne značajke

Ako postoji prirodan poredak kategorija, ta se informacija treba zadržati. Prirodno je da se kategorije redom preslikavaju u cjelobrojne pozitivne vrijednosti. Kao primjer ordinalne varijable i njezina preslikavanja mogu poslužiti dani u tjednu:
ponedjeljak → 1; utorak → 2; srijeda → 3; itd.

3.3.2. Skaliranje podataka

Modeli strojnog učenja uglavnom bolje rade ako su sve značajke na istoj mjernoj skali. U stvarnosti je to gotovo nemoguće jer najčešće nije moguće odabrati na koji način se mjere podaci, već je dostupne podatke potrebno prilagoditi. Modeli koji su osjetljivi na veličine varijabli su svi linearni modeli (linearna i logistička regresija) i modeli koji koriste mjere udaljenosti među primjerima (k najbližih susjeda, stroj potpornih vektora), dok su modeli stablastih struktura (stablo odluke, slučajna šuma) otporni na nesrazmjer prediktorskih veličina [13].

Dva su najčešća pristupa normalizacija i standardizacija. Normalizacija skalira značajku formulom 3.1 na raspon $[0, 1]$. Varijable se mogu skalirati i na neki drugi raspon, ali to je rijetkost u praksi.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

Normalizacijom se ne mijenja razdioba varijable, nego samo raspon u kojem se nalaze njezine vrijednosti.

Standardizacijom se postiže da transformirana varijabla ima srednju vrijednost nula i standardnu devijaciju jedan, a računa se formulom 3.2:

$$x_{std} = \frac{x - \overline{x}}{\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x - \overline{x})^2}} \tag{3.2}$$

Statistike potrebne za transformacije prikupljaju se iz skupa za učenje. Nakon skaliranja tog skupa, skaliraju se podaci u skupu za testiranje kako ne bi došlo do curenja informacija. Standardizacija se uglavnom koristi za normalne varijable, a normalizacija za ostale. Moguće je i provesti standardizaciju pa normalizaciju podataka kako bi sve značajke bile u istom rasponu [13].

3.3.3. Promjena razdiobe

Numeričke ulazne značajke mogu imati i nestandardnu razdiobu (zakrivljena, multimodalna, eksponencijalna). Neki algoritmi ne rade dobro s takvim podacima zbog pretpostavki o razdiobi podataka ugrađenih u modele (npr. linearna regresija) [13]. Oni drugi, koji ne pretpostavljaju razdiobu ulaznih podataka, također mogu imati problema s nestandardno distribuiranim podacima, stoga je često poželjno transformirati podatke da budu (približno) normalno ili uniformno distribuirani.

Transformacija kvantila (engl. *quantile transform*) nelinearna je transformacija koja mijenja razdiobu ulazne varijable u normalnu ili uniformnu. Iako ovakva promjena u podacima može pozitivno utjecati na rezultate modeliranja, gubi se interpretacija značajke — više nije sasvim jasno kako promjena na ulazu modela utječe na izlaz.

Drugi način na koji se može transformirati numerička varijabla je diskretizacijom njezinih vrijednosti. To je metoda kojom se bliske vrijednosti originalne varijable svrstavaju u iste grupe, tako da se smanji broj mogućih vrijednosti značajke. Broj grupa (engl. *bins*) k određuje se proizvoljno, ali mora biti znatno manji od broja različitih vrijednosti značajke da bi transformacija imala smisla. Zapravo od numeričke značajke nastaje kategorička (ordinalna), a optimizacijom hiperparametra dobiva se optimalan broj grupa. Tri su osnovna načina diskretizacije:

- Podjela na jednak broj intervala (engl. *uniform*[13], *equal-interval binning*[6]) —

raspon originalne varijable podijeljen je u k grupa, tako da svaka grupa pokriva jednak interval ($1/k$ raspona varijable).

- Podjela u grupe s jednakim brojem primjera (engl. *quantile*[13], *equal-frequency binning*[6]) — u odnosu na broj grupa određuju se granice intervala prema odgovarajućim kvantilima. Posljedica toga je da svaki interval sadrži jednak broj primjera (uniformna razdioba).

- Podjela algoritmom k srednjih vrijednosti (engl. *clustered, kmeans*) — primjeri su podijeljeni u grupe prema najbližoj od k sredina [18].

3.4. Odabir značajki

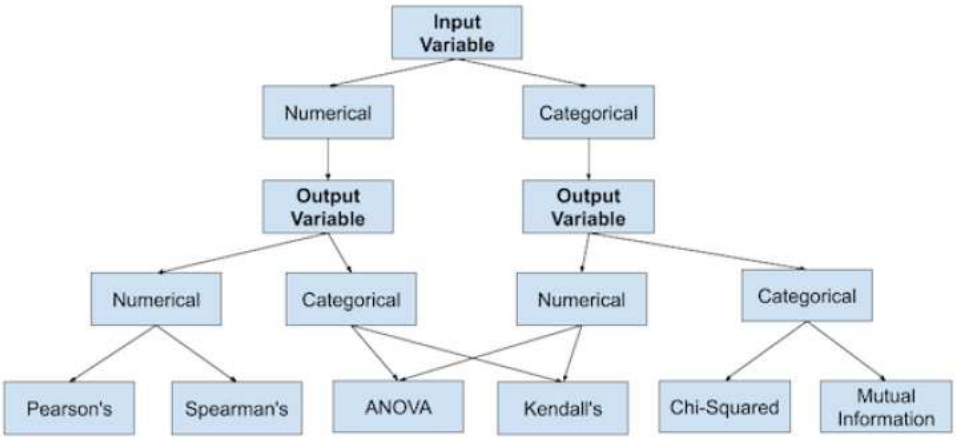
Odabir značajki postupak je smanjenja broja varijabli skupa podataka. Obuhvaća metode za identifikaciju najvažnijih značajki, odnosno onih koje pružaju najviše informacije za predviđanje ciljne varijable. Pritom se ne mijenja interpretacija odabranih značajki — izvorne značajke se ne mijenjaju (za razliku od metoda smanjenja dimenzionalnosti). Ovaj je korak važan jer redundantne i irelevantne značajke ne donose dodatne informacije o prirodi problema. Model to ne prepoznaje i dodjeljuje im određenu težinu, što potencijalno smanjuje performanse modela. Osim toga, smanjenjem broja značajki smanjuje se i vrijeme potrebno za izgradnju modela, što je ključno za neke probleme.

3.4.1. Filterski postupci

Filterskim postupcima odabiru se značajke na temelju njihove povezanosti s ciljnom značajkom. Pokušava se odrediti prediktivna moć pojedine značajke statističkim mjerama, koje se potom koriste za rangiranje značajki. Moguće je unaprijed odabrati željeni broj značajki ili ispitati ponašanje modela za podskupove s različitim brojem značajki.

Za odabir značajki postupkom filtera često se koriste statističke mjere koje odražavaju korelaciju prediktorskih značajki s ciljnom. Za svaku ulaznu značajku računa se odabrana statistika koja ju povezuje s izlaznom. Budući da ovakav pristup promatra ulazne značajke pojedinačno, njime se ne mogu otkriti interakcije (i potencijalna redundantnost) između ulaznih varijabli.

Prilikom korištenja statističkih mjera potrebno je obratiti pozornost na vrstu varijabli (numeričke/kategoričke) jer nisu sve mjere pogodne za svaki tip podataka. Za regresijske probleme može se koristiti korelacija, za kategoričke ulazne značajke najčešće se koristi χ^2 statistika, a postoje i mjere koje se mogu primijeniti u više slučajeva. Slika 3.5. prikazuje preporučene statističke mjere [13] u ovisnosti o tipu ulazne i ciljne varijable.



Slika 3.5. Statistike za odabir značajki filterskim postupkom [13]

INFORMACIJSKA DOBIT

Informacijska dobit (engl. *information gain*) mjera je koja je neovisna o vrsti ulaznih i izlazne značajke. Definirana je formulom 3.3 za diskretne varijable (analogno za kontinuirane varijable).

$$\begin{aligned} I(X;Y) &= \sum_{x\in X}\sum_{y\in Y} P(x,y) log \frac{P(x,y)}{P(x)P(y)} \\ &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \end{aligned} \tag{3.3}$$

U kontekstu odabira značajki naziva se i mjerom zajedničke informacije (engl. *mutual information*), a odnosi se na međusobnu povezanost dviju značajki. Konkretno, mjeri sadržanu informaciju o ciljnoj značajki u pojedinoj prediktorskoj značajki [19]. Ako je informacijska dobit jednaka nuli, znači da su značajke statistički nezavisne, odnosno da prediktorska značajka ne sadrži nikakvu informaciju o ciljnoj. Prediktorske značajke rangiraju se po količini zajedničke informacije, a zatim se odabiru najkorisnije značajke.

Ova mjera temelji se na računanju entropije, a ona preferira značajke s većim brojem vrijednosti i (približno) uniformnom razdiobom [20], stoga nije prikladna za svaki problem, iako formalno ne ovisi o tipu varijabli.

3.4.2. Postupci omotača

Postupci omotača (engl. *wrapper methods*) pretražuju prostor značajki kako bi pronašli optimalan podskup. Iterativno se dodaju (ili uklanjaju) značajke u novi podskup i evaluira se uspješnost tako naučenog modela. Ovakav pristup može dobro raditi s bilo kojom vrstom podataka, ali je jako spor i računalno složen. Pretraživanje cijelog prostora značajki je uglavnom presloženo pa se odabir i dodavanje značajki u novi podskup provodi pohlepnim pretraživanjem.

Dva najpoznatija algoritma implementirana su u Python paketu scikit-learn:

1. **RFE** (engl. *recursive feature elimination*) algotiram je koji rangira značajke po važnosti, a zatim rekurzivno uklanja najmanje važne značajke. Važnost značajki može se odrediti težinama koje imaju u modelu ili nekom od mjera koje se koriste u filterskim metodama. Bitno je naglasiti da značajke moraju biti na istoj skali (isti red veličine) kako bi se težine mogle ispravno dodijeliti.

   Postupak započinje s potpunim skupom značajki. Nakon odbacivanja prve, ponovo se uči model s novim (manjim) skupom značajki i preostale se značajke ponovo rangiraju. Postupak se provodi dok se podskup značajki ne smanji do unaprijed definiranog broja [6, 13, 21]. Implementiran je i algoritam koji unakrsnom validacijom automatski odabire najbolji broj značajki (RFECV — engl. *recursive feature elimination cross validation*) [22].

2. **Slijedna selekcija značajki** (engl. *sequential feature selection*) pohlepno pretražuje prostor značajki. Ovisno o početnom skupu može biti unaprijedna (engl. *forward*) ako postupak započinje s praznim skupom značajki pa dodaje jednu po jednu ili selekcija unazad (engl. *backward*) ako se značajke uklanjaju iz skupa svih značajki. Potrebno je odabrati model strojnog učenja i metriku kojom će se vrednovati kako bi se mogle birati značajke.

   Za unaprijednu selekciju algoritam u svakom koraku odabire značajku koja naj-

više doprinosi kvaliteti modela (daje najbolje rezultate unakrsnom validacijom) i dodaje ju u skup značajki. Dodavanje značajki traje sve dok se popravlja kvaliteta modela. Analogno vrijedi i za selekciju unazad — najmanje bitne značajke se uklanjaju jedna po jedna dok se ostvaruje poboljšanje. Ova metoda ne garantira pronalazak optimalnog skupa značajki za odabrani problem jer je moguće da pretraga završi u lokalnom ekstremu. [6], [23]

3.4.3. Ugrađeni postupci

Postoje modeli strojnog učenja koji određuju važnost značajki svojom unutarnjom strukturom. Ovakav postupak odabira značajki zove se ugrađeni (engl. *intrinsic*), upravo zbog "sposobnosti" određenog modela da odabere bitne značajke. To su linearni modeli (linearna i logistička regresija, stroj potpornih vektora[1]) s L1 (LASSO) regularizacijom ili modeli stablaste strukture (slučajna šuma).

L1 regularizacija postavlja težine nebitnih značajki na nulu, čime se implicitno u modelu odabiru bitnije značajke. Jačina regularizacije određuje se parametrom λ (za $\lambda = 0$ nema regularizacije). Nakon učenja takvog modela, odabiru se značajke čije su težine "preživjele", odnosno ostale su veće od 0.

Model slučajne šume sastoji se od velikog broja stabala odluke. Prilikom svakog grananja može se izračunati pripadna mjera nečistoće (grananje je čisto ako su u jednom listu primjeri samo jedne klase). Važnije značajke su u pravilu bliže korijenu, nego one bliže listovima stabla. Značajke koje prosječno (diljem slučajne šume) više smanjuju nečistoću prilikom grananja smatraju se važnijima.

3.5. Smanjenje dimenzionalnosti

Dimenzionalnost skupa podataka određuje broj značajki — primjeri su točke u višedimenzionalnom prostoru, a svaka značajka definira novu dimenziju tog prostora. Smanjenje dimenzionalnosti odnosi se na projekciju početnih podataka u neki drugi prostor tako da se zadrži što više inicijalne informacije iz podataka, uz uklanjanje dijela dimenzija (značajki). Ova se metoda razlikuje od odabira značajki jer ovim postupkom nastaju

[1] generalizirani linearni model

nove značajke koje se više ne mogu interpretirati, za razliku od odabira značajki u kojem se izvorne značajke ne mijenjaju. Uglavnom se provodi nakon čišćenja i skaliranja podataka, a prije samog modeliranja.

3.5.1. Linearne metode

Najpoznatija metoda smanjenja dimenzionalnosti je analiza glavnih komponenti (engl. *principal component analysis, PCA*). Ideja je ove metode prikazati postojeće podatke u drugom koordinatnom sustavu, koji ovisi o samim podacima. Novi koordinatni sustav mora biti ortogonalan, tj. osi svih njegovih dimenzija moraju biti međusobno okomite. Smjer osi određuje se smjerom najveće varijabilnosti u podacima — od svih mogućnosti, odabire se smjer s najvećom varijancom (određuje se prema svojstvenim vrijednostima kovarijacijske matrice skupa podataka). Ukupna varijabilnost originalnih i transformiranih značajki jednaka je, a odabirom manjeg podskupa transformiranih značajki i dalje se može zadržati velik dio informacije (varijabilnosti) podataka, uz manje dimenzija [6]. Broj komponenti može se odrediti na temelju pragova objašnjene varijance — varijanca po komponenti ili kumulativna varijanca.

Za klasifikacijske probleme može se koristi i linearna diskriminantna analiza (engl. *linear discriminant analysis, LDA*). To je statistička metoda koja razdvaja primjere na temelju njihove ciljne značajke. Cilj je pronaći linearnu kombinaciju ulaznih varijabli takvu da primjerci različitih klasa budu međusobno što udaljeniji. Nakon transformacije, odabiru se najznačajnije komponente transformacije, a broj komponenti može se odraditi unakrsnom validacijom. Prije provođenja algoritma, potrebno je skalirati sve ulazne varijable na istu mjernu skalu kako bi algoritam pravedno dodijelio težine komponentama [13].

3.5.2. Nelinearne metode

Tehnike učenja mnogostrukosti (engl. *manifold learning*) ubrajaju se u nelinearne metode smanjenja dimenzionalnosti podataka. Cilj tih metoda je prikazati visokodimenzionalne podatke u njihovom latentnom prostoru, čak i kada postoji nelinearan odnos među značajkama. Značajke se preslikavaju u latentni (nižedimenzionalni) prostor tako da se sličnost među primjercima iz višedimenzionalnog prostora odražava na udaljenosti

u latentnom prostoru (sličniji primjerci nalaze se bliže jedan drugome) [24]. Često se koriste za vizualizaciju visokodimenzionalnih skupova podataka, a neke metode su [25]:

- Višedimenzijsko skaliranje (engl. *Multidimensional Scaling, MDS*) — temelji se na mjerama sličnosti koje predstavljaju metričku udaljenost između točaka u niskodimenzionalnom prostoru; cilj je minimizirati razliku udaljenosti u višedimenzijskom i niskodimenzijskom prostoru.

- Lokalno-linearno ugrađivanje (engl. *Locally Linear Embedding, LLE*) — nastoji očuvati udaljenosti među točkama u lokalnom susjedstvu, odnosno gleda samo lokalnu strukturu najbližih podataka.

- t-SNE (engl. *t-distributed Stochastic Neighbor Embedding*) — sličnost je predstavljena vjerojatnostima udaljenosti između primjera (sličnijim primjerima pridružena je veća zajednička vjerojatnost). U originalnom prostoru sličnost je definirana Gaussovom zajedničkom vjerojatnošću, a u nižedimenzionalnom prostoru Studentovom razdiobom. Ova metoda nije primjenjiva na testni skup podataka, već uglavnom služi za vizualizaciju podataka.

4. Problem nebalansiranosti podataka u klasifikaciji

Klasifikacija je metoda nadziranog učenja u kojoj model dodjeljuje oznaku ciljne klase svakom primjeru. Prilikom učenja o klasifikatorima u pravilu se podrazumijeva da su primjeri ciljne klase uniformno distribuirani. U protivnom modeli i metrike koji se uobičajeno koriste ne rade dobro, odnosno može se činiti da daju dobre rezultate, ali u stvarnosti ne modeliraju dovoljno dobro zadani problem.

4.1. Nebalansiranost klasa

Nebalansiranost podataka (engl. *imbalance*) u klasifikaciji odnosi se na nepogodan omjer primjeraka pojedinih klasa ciljne značajke. U kontekstu strojnog učenja nepogodan omjer znači neravnotežu u omjerima primjera različitih klasa — balansiran skup ima približno uniformnu razdiobu, a nebalansiran zakrivljenu. Nebalansiranost se može odnositi na binarnu, kao i na višeklasnu klasifikaciju, ali u nastavku se podrazumijeva binarna klasifikacija.

Nebalansiranost problema ovisi o konkretnom skupu za učenje, a izražava se omjerom nebalansiranosti (engl. *imbalance ratio, IR*) [26], npr. omjer 100:1 podrazumijeva da za svaki primjer manjinske klase u skupu za učenje postoji 100 primjera većinske. Omjer nebalansiranosti može imati vrijednost od 1 do preko 1000, a na temelju njega koriste se različiti pristupi rješavanju ovog problema. Prema omjeru nebalansiranosti mogu se definirati skupine problema neuravnoteženosti [27]:

- blaga nebalansiranost ($IR < 4$)
- srednja nebalansiranost ($IR \in \langle 4, 100]$)

- velika nebalansiranost ($IR > 100$)

Kod problema s neuravnoteženim klasama obično predviđanje manjinske klase ima veći značaj. Njih je teže predvidjeti jer prirodno ima manje takvih primjera, a samim time i manje informacija kojima se mogu predstaviti i razlikovati od većinske klase. Klasifikatori u takvoj situaciji uče ponajviše iz karakteristika većinske klase (jer prevladavaju u skupu za učenje), a zanemaruju manjinsku klasu koja ima veći značaj. Glavni pristupi nebalansiranosti su ponovno uzorkovanje podataka, kako bi se smanjio omjer nebalansiranosti, i prilagodba troška pogrešne klasifikacije — pogrešno klasificirani pozitivni i negativni primjeri imaju različite težine u funkciji gubitka.

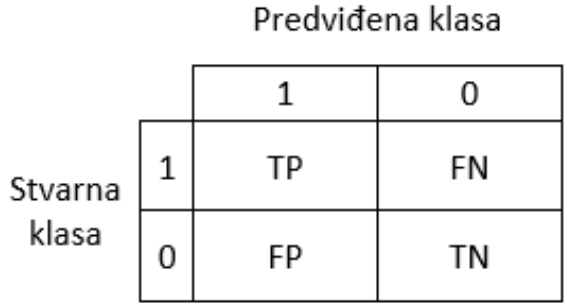
4.2. Evaluacija klasifikatora

Uspješnost klasifikatora određuje se metrikom kojom se vrednuju njegova predviđanja u usporedbi sa stvarnim oznakama. Pogrešnim odabirom metrike smanjuje se sposobnost klasifikatora da uspješno modelira stvarni problem. Odabir metrike izazov je u strojnom učenju, a osobito kod nebalansiranih skupova podataka [27].

Metrike se koriste ne samo za evaluaciju konačnog rezultata modela, nego i u prethodnim koracima: odabir značajki, odabir modela, optimizacija hiperparametara. Zbog toga je bitno da korištena metrika bude prilagođena problemu i razdiobi podataka.

Za vrednovanje klasifikatora koriste se izlazi iz modela (dobivena predviđanja) i uspoređuju s pravim oznakama primjera. Temelj svih mjera leži u matrici zabune (engl. *confusion matrix*), koja sadrži razdiobu ispravno i neispravno predviđenih pozitivnih (klasa 1) i negativnih primjera (klasa 0) [28]. Za binarnu klasifikaciju postoje četiri polja: ispravno klasificirani pozitivni primjeri (engl. *true positive, TP*), neispravno klasificirani pozitivni primjeri (engl. *false positive, FP*), ispravno klasificirani negativni primjeri (engl. *true negative, TN*) i neispravno klasificirani negativni primjeri (engl. *false negative, FN*), a njihov poredak u matrici zabune prikazan je na slici 4.1.

Iz matrice zabune izvedene su brojne metrike koje služe u evaluaciji klasifikatora [28], a one najčešće korištene su:



Slika 4.1. Matrica zabune

- **Točnost** (engl. *accuracy*) izražava udio ispravno klasificiranih primjera u cijelom testnom skupu (N je ukupan broj primjera):

$$precision = \frac{TP + TN}{N} \tag{4.1}$$

- **Preciznost** (engl. *precision*) pokazuje koliko je pozitivnih primjera zapravo ispravno klasificirano (od svih koji su predviđeni kao pozitivni):

$$precision = \frac{TP}{TP + FP} \tag{4.2}$$

- **Odziv, osjetljivost** (engl. *recall, sensitivity*) je udio pogođenih pozitivnih primjera od svih koji su zapravo pozitivni (engl. *true positive rate*):

$$recall = \frac{TP}{TP + FN} \tag{4.3}$$

- **Specifičnost** (engl. *specificity*) je ekvivalent osjetljivosti za negativnu klasu:

$$specificity = \frac{TN}{TN + FP} \tag{4.4}$$

- **F1 mjera** (engl. *F1 score*) uravnotežuje odnos preciznosti i odziva:

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{4.5}$$

- **Generalizirana F mjera,** F_β parametrom β kontrolira odnos između preciznosti i odziva — što je β manji, preciznost ima veću težinu u ukupnoj mjeri.

$$F_\beta = (1 + \beta^2)\frac{precision * recall}{\beta^2 * precision + specificity} \tag{4.6}$$

- **Udio lažno pozitivnih primjera** (engl. *false positive rate, FPR*) udio lažnih pozitivnih primjera među svim negativnim:

$$FPR = \frac{FP}{TN + FP} \tag{4.7}$$

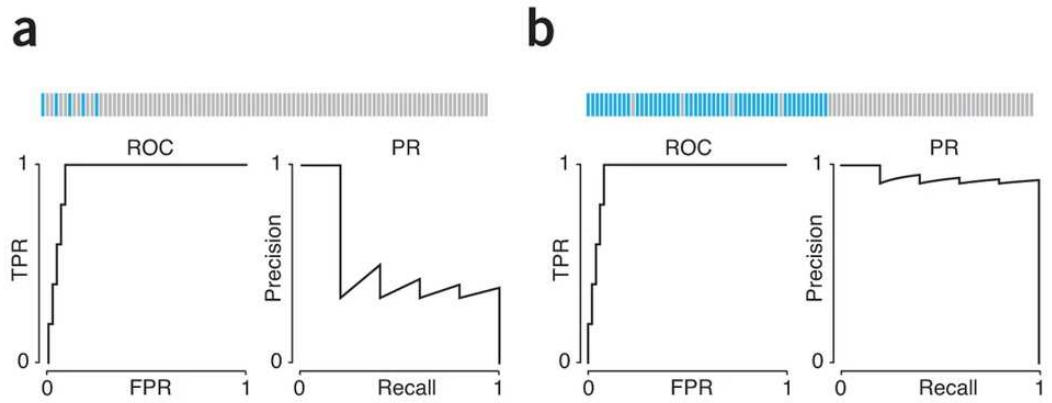
- **Stopa lažnog otkrivanja** (engl. *false discovery rate, FDR*) udio pogrešno klasificiranih pozitivnih primjera među svim pozitivno klasificiranim:

$$FDR = \frac{FP}{TP + FP} \tag{4.8}$$

Navedene metrike daju brojčanu vrijednost procjene kvalitete modela, ali često nisu dobri procjenitelji za probleme s nebalansiranim podacima, osobito ako se promatra samo jedna metrika. Postoje i metrike koje procjenjuju sposobnost razdvajanja klasa modela — za različite pragove grafički prikazuje odnos dviju metrika.

Najpoznatija je takva metrika ROC (engl. *Receiver Operating Characteristic*) krivulja [27]. Krivulju definiraju vrijednosti metrika odziva (TPR, formula 4.3) i udio lažno pozitivnih primjera (FPR, formula 4.7). ROC krivulja za beskoristan klasifikator (pridružuje većinsku klasu svim primjerima) dijagonalna je linija koja spaja donji lijevi s gornjim desnim kutom. Dobar klasifikator ima ROC krivulju "iznad" te dijagonale, a savršen je reprezentiran točkom u gornjem lijevom kutu. Iako uzima u obzir dvije metrike, ova metrika nije povoljna za nebalansirane skupove jer za njih često daje optimistične rezultate — lažno pozitivni primjeri nisu povezani sa zapravo pozitivnim [28].

Za nebalansirane skupove prikladnije je koristiti PR (engl. *Precision-Recall*) krivulju. Ona umjesto odnosa TPR i FPR oslikava odnos preciznosti (4.2) i odziva(4.3). Za razliku od ROC krivulje, za PR krivulju nije jednoznačno definiran izgled krivulje dobrog klasifikatora. To se određuje na temelju nebalansiranosti skupa — računa se krivulja



Slika 4.2. Usporedba ROC i PR krivulja za različite probleme

(horizontalni pravac) za tzv. *dummy* klasifikator i uspoređuje se s njom.

Na slici 4.2. (preuzeta iz [28]) prikazane su ROC i PR krivulje za skupove podataka različite balansiranosti: (a) je primjer nebalansiranih podataka, s 5% manjinske klase; (b) prikazuje balansirani skup podataka, u kojem obje klase imaju po 50% primjera. ROC krivulja u oba slučaja izgleda isto i sugerira da oba modela daju dobre rezultate. Ako se promotri PR krivulja, ona je puno lošija za model koji je naučen na nebalansiranim podacima i ukazuje na manjkavost tog modela, stoga je prikladnija za korištenje s nebalansiranim skupovima.

Ne postoji metrika koja je uvijek pouzdana, nego je potrebno koristiti one metrike koje su najprikladnije za konkretan problem i količinu nebalansiranosti. Smjernice za iznos odabranih metrika mogu se odrediti "beskorisnim" modelima koji predviđaju samo jednu klasu. Model koji svim primjerima pridružuje većinsku klasu imat će izvrsnu točnost i loš odziv. S druge strane, model koji klasificira sve kao manjinsku klasu za nebalansirane skupove ima jako lošu točnost, ali savršen odziv. Ukoliko dobiveni rezultati nisu bolji od rezultata ovih modela, to je znak da model ne predstavlja dobro zadani problem. Čini se da je odabir metrike pravi izazov, osobito što je omjer nebalansiranosti veći, ali ima velik značaj za dobivanje dobrih rezultata.

4.3. Ponovno uzorkovanje podataka

Uzorkovanje podataka (engl. *data sampling*) skup je tehnika kojima se transformira početni skup podataka tako da se promijeni razdioba ciljne značajke. Cilj je dobiti (više)

balansiran skup podataka kako bi se poboljšalo i olakšalo učenje modela. To je najjednostavniji pristup rješavanju problema nebalansiranosti — jednostavno se razumije i implementira, ne traži znanje o algoritmima strojnog učenja.

Uzorkovanje se provodi samo na skupu za učenje — njegova je svrha pružiti što reprezentativnije podatke za učenje kako bi model što bolje mogao odvojiti klase. Distribucija ciljne klase testnog skupa u tom slučaju nije važna, a ako se ona promijeni može doći do optimistične procjene rezultata modela [27].

4.3.1. Naduzorkovanje

Naduzorkovanje (engl. *oversampling*) podrazumijeva kopiranje ili generiranje novih primjeraka manjinske klase kako bi se povećao njihov udio u skupu za učenje. Nekoliko implementiranih i često korištenih metoda naduzorkovanja [27]:

- **Slučajno naduzorkovanje** (engl. *Random Oversampling*) nasumično odabire primjerke manjinske klase i ponovno ih dodaje u skup za učenje. Pojedini primjer može biti odabran i udvostručen više puta, što može dovesti do prenaučenosti ukoliko je skup podataka izrazito nebalansiran [27].

- **SMOTE** (engl. *Synthetic Minority Oversampling Technique*) postupkom generiraju se novi primjeri manjinske klase na temelju susjednih primjera. Za primjere manjinske klase traži se njihovih k najbližih susjeda i nove točke (primjeri) generiraju se slučajnim odabirom na liniji između njih. Ovisno o potrebnoj količini novih podataka, odabire se broj najbližih susjeda za generiranje novih primjera: npr. za naduzorkovanje od 200% uzimaju se dva susjeda [29]. Ova je metoda učinkovita za naduzorkovanje i izvor je mnogim drugim izvedenim metodama, ali je računalno dosta zahtjevna i u nekim slučajevima dolazi do preklapanja primjera različitih klasa.

- ***Borderline-SMOTE*** je metoda koja se temelji na SMOTE algoritmu, ali uz ograničenje odabira primjeraka manjinske klase. Pretpostavlja da primjeri bliže granici između klasa imaju veću šansu biti pogrešno klasificirani pa za naduzorkovanje odabire samo takve primjerke manjinske klase [30].

- ***Borderline-SMOTE SVM*** postupak sličan je prethodnom, ali za određivanje de-

cizijske granice i pogrešno klasificiranih primjera koristi stroj potpornih vektora (SVM) umjesto algoritma k najbližih susjeda. Osim toga, nastoji više popuniti dio prostora u kojemu se nalazi manje primjera većinske klase kako bi se poboljšala decizijska granica [31].

- **ADASYN** (engl. *Adaptive Synthetic Sampling*) metodom broj primjera manjinske klase koji se generiraju ovisi o okruženju u kojem se nalaze. Za primjere koje je teže naučiti (oni koji su okruženi primjercima većinske klase) generira se više novih primjera. Tamo gdje je njihova gustoća u prostoru veća, generira se mlao ili se uopće ne generiraju novi primjeri. Za svaki manjinski primjerak računa se omjer broja većinskih primjeraka u njihovom susjedstvu i prema tome se određuje koliko će se novih točaka generirati za pojedini primjer manjinske klase (što je izračunati omjer veći, dobiva se više novih točaka). Postupak generiranja je sličan kao i kod SMOTE-a: slučajno se odabiru manjinski susjedi i na liniji između njih se generira novi primjer [32].

4.3.2. Poduzorkovanje

Proces poduzorkovanja (engl. *undersampling*) uklanja dio primjeraka većinske klase iz skupa za učenje kako bi se smanjio omjer nebalansiranosti. Dva su osnovna pristupa promjene skupa podataka: zadržavanje korisnih i uklanjanje nebitnih primjera.

- **Slučajno poduzorkovanje** (engl. *Random Undersampling*) nasumično odabire primjere većinske klase koje potom uklanja iz skupa za učenje. Ova je metoda jednostavna i učinkovita, ali ne može se primijeniti na manjim skupovima podataka. Ako se pokušaju uravnotežiti jako nebalansirane klase, dolazi do prevelikog gubitka informacija i gubljenja sposobnosti učenja modela.

- **CNN** (engl. *Condensed Nearest Neighbor Rule*) tehnikom cilj je dobiti podskup podataka koji zadržava većinu informacija potrebnih za odvajanje klasa. CNN je iterativna metoda: počinje sa skupom uzoraka manjinske klase i jednim slučajnim primjerom većinske klase. U svakom koraku uči se model 1-najbližeg susjeda i dodaje po jedan primjer većinske klase koji je pogrešno klasificiran u tom koraku. Time se osigurava da samo kritični uzorci koji definiraju granice između klasa budu zadržani [33].

- **Metoda bliskog promašaja** (engl. *Near Miss Undersampling*) metode odabiru primjere koje treba zadržati u skupu podataka na temelju udaljenosti primjera većinske i manjinske klase (zadržava bliže primjere). Postoje tri varijante ove metode, a razlikuju se po udaljenostima najbližih susjeda koje promatraju za svaki primjer većinske klase [27].

- **Metoda Tomekovih poveznica** (engl. *Tomek Links Undersampling*, TL) dobra je metoda poduzorkovanja za uklanjanje šuma u podacima. Cilj je ukloniti primjere koji su blizu primjerima suprotne klase. Tomekova poveznica definirana je između dvaju primjera različitih klasa ako su jedan drugome najbliži susjed [33]. Ovom metodom može se ukloniti samo jedan ili oba člana Tomekove poveznice. U kontekstu nebalansiranih skupova uklanja se samo primjer većinske klase.

- **ENN** (engl. *Edited Nearest Neighbors Rule*) pravilo je kojim se pronalaze podaci koji potencijalno stvaraju šum. Koristeći algoritam k-najbližih susjeda zadržavaju se oni primjeri većinske klase koji imaju većinu (ili sve) svojih najbližih susjeda unutar većinske klase [33].

- **OSS** (engl. *One-Sided Selection*) metoda je koja kombinira dvije spomenute metode: metodu Tomekovih poveznica i CNN. Prvo se pomoću metode CNN pronalaze primjeri koji se teško klasificiraju, a zatim pomoću metode TL uklanjaju primjeri većinske klase koji stvaraju šum [27].

- **NCR** (engl. *Neighborhood Cleaning Rule*) slično kao i OSS, pravilo čišćenja susjedstva koristi dvije metode (za čišćenje i zgušnjavanje podataka), ali u obrnutom poretku. Prvo se metodom CNN dobiva manji i kompaktniji skup podataka, a potom se uklanja šum metodom ENN (uklanja primjerke većinske klase koji imaju više od pola najbližih susjeda manjinske klase) [27].

4.3.3. Hibridno uzorkovanje

Pristupi ponovnog uzorkovanja mogu se i kombinirati u svrhu poboljšanja rezultata. Korištenjem obiju tehnika dobiva se balansirani skup podataka, uz smanjenje vjerojatnosti pojavljivanja problema pojedinačnih metoda (prenaučenost i podnaučenost). Obično se prvo izvodi poduzorkovanje, a zatim naduzorkovanje. Mogu se kombinirati bilo koje

dvije metode poduzorkovanja i naduzorkovanja, najčešće se koristi SMOTE u paru s nekom metodom poduzorkovanja.

Najjednostavnija kombinacija je korištenje SMOTE-a i slučajnog poduzorkovanja. Prvo se slučajnim poduzorkovanjem uklanjaju primjerci većinske klase metodom tako da se smanji omjer nebalansiranosti. Nakon toga metodom SMOTE povećava se udio manjinske klase u podacima sve dok se ne dobije balansiran skup. Najčešće korištene složenije kombinacije (implementirane u paketu imbalanced-learn) su SMOTE i TL, odnosno SMOTE i ENN.

4.4. Učenje osjetljivo na cijenu

Modeli strojnog učenja podrazumijevaju jednaku cijenu pogrešne klasifikacije za primjere obiju (svih) klasa. Kod problema čiji su podaci nebalansirani to uglavnom nije tako jer je cilj otkriti primjere manjinske klase, koji se rijetko ostvaruju. Obično veću težinu ima pozitivan primjer (manjinska klasa) klasificiran kao negativan, nego obrnuto.

Dio strojnog učenja koje se bavi modelima koji mogu razlikovati težine pogrešne klasifikacije zove se učenje osjetljivo na cijenu (engl. *Cost-Sensitive Learning*). Tijekom učenja model pokušava minimizirati ukupnu pogrešku. Ako su cijene pogrešaka različite, tada je cilj modela minimizirati ukupni trošak.

Trošak pogrešne klasifikacije može se prikazati matricom cijene (engl. *cost matrix*). Ima isti format kao i matrica zabune, koja se koristi za procjenu kvalitete klasifikatora, ali njezini su elementi troškovi za sve kombinacije predviđenih i stvarnih oznaka klasa. U idealnom slučaju, matrica cijene određuje se uz pomoć stručnjaka na temelju njihova domenskog znanja. U praksi je to rijetkost — teško je doći do eksperta, a ponekad takva procjena nije ni moguća. Zbog toga se cijena pogrešne klasifikacije često određuje automatski iz podataka: cijena pogrešne klasifikacije negativnog primjera (FP) iznosi 1, a pogrešno klasificirani pozitivni primjeri (FN) imaju trošak jednak omjeru nebalansiranosti. Korištenjem ovako definiranih cijena, pretpostavlja se da je razdioba podataka jednaka na skupovima za učenje i testiranje. Na slici 4.3. prikazana je matrica cijene za skup podataka s omjerom nebalansiranosti 100:1.

Ne postoje algoritmi strojnog učenja koji su napravljeni specijalno za učenje osjetljivo

		Predviđena klasa	
		1	0
Stvarna klasa	1	0	100
	0	1	0

Slika 4.3. Matrica cijene za skup s omjerom nebalansiranosti 100:1

na cijenu, ali u tu svrhu mogu se koristiti modeli koji znaju raditi s otežanim primjercima. Najpoznatiji su modeli stroj potpornih vektora, stablo odluke i logistička regresija. Cijene (težine) se mogu dodati direktno na primjere pojedinih klasa ili se dodaju kao hiperparametar algoritmu.

5. Modeliranje problema: otkrivanje rizičnih sportskih listića

Ako se promatra iz ekonomske perspektive, rizik se može definirati i kao mogućnost kvantifikacije nesigurnosti pri donošenju poslovnih odluka [3]. U ovom radu cilj je klasificirati sportske listiće koji upućuju na nepošteno klađenje igrača (kao na primjer uplata velikog iznosa na ishod koji je malo vjerojatan). To može upućivati da igrač ima neku "prednost" u odnosu na kladionicu i očekuje veliki profit — što znači veliki gubitak za kladionicu. Ne postoji način da se sa sigurnošću utvrdi ima li igrač zapravo prednost (dojavu) ili ne, ali takvi su listići prerizični iz perspektive kladionice. Ukoliko bi se samo nekoliko takvih ekstremnih slučajeva dogodilo, cijelo poslovanje i opstanak kladioničarske kuće dolazi u pitanje, stoga treba oprezno pristupiti tom problemu.

Ne postoji neka konkretna značajka, niti jednostavno definirano pravilo koje razlikuje rizične listiće od ostalih. Neki od indikatora mogu biti količina uplaćenog novca, (ne)popularnost lige, koeficijent, itd. U dostupnim podacima, rizične listiće označili su kladioničari, na temelju svog iskustva. Korištenjem metoda dubinske analize podataka i modela strojnog učenja cilj je automatski detektirati takve listiće.

5.1. Opis početnog skupa podataka

Podaci koji se koriste dobiveni su iz baze podataka jednog poduzeća iz domene sportskog klađenja, a dostupni su na poveznici. Dio podataka maskiran je kako ne bi došlo do povrede čuvanja osobnih podataka (oznaka igrača, listića i grada). U tekstu koji slijedi, pojam *igrač* odnosi se na osobu koja se kladi na sportski rezultat, a ne na osobu koja sudjeluje u sportskom natjecanju (zbog konzistentnosti nazivlja s dobivenim podacima).

Skup podataka podijeljen je u nekoliko tablica. Podaci se odnose na igrače (njihovu

registraciju i prošlu aktivnost) te uplate sportskih listića tijekom četiriju mjeseci. Dodatno, dostupni su i podaci o udjelima uplata sportskih listića po svim sportovima u prethodnoj godini. Za tri najlikvidnija sporta dostupni su udjeli uplate po natjecanjima, također za prethodnu godinu. Ti se podaci koriste kao smjernica u definiranju novih značajki za "jačinu" sporta i natjecanja, odnosno detekciju potencijalno nepravednog klađenja (npr. oklada na utakmicu Lige prvaka nema jednaku težinu kao utakmica ekvadorske lige). Što je veća količina uplate novca na pojedino natjecanje, znači da je ono popularnije, gledanije i profesionalnije (manje sklono namještanju). Osim toga, to implicira i veću količinu "dostupnog" novca za to natjecanje — može se bez (velikog) gubitka isplatiti neki veći iznos igraču koji se kladio na malo vjerojatni ishod i pogodio.

Podatkovni skup detaljno je opisan u nastavku — za svaku tablicu navedeni su sadržani atributi i pripadni tip podataka:

1. **PlayersRegistrationDetails** — sadrži podatke o registraciji igrača na platformu. Da bi se igrač mogao kladiti, prvo se mora registrirati na platformi. Tablica sadržava podatke samo o igračima koji su bili aktivni u promatranom razdoblju na platformi za sportsko klađenje.
    - *player_id*: jedinstveni identifikator igrača
    - *registration_date*: vrijeme registracije igrača
    - *city*: grad iz kojeg dolazi igrač (naveden za vrijeme registracije)
    - *date_of_birth*: datum rođenja igrača
2. **PlayersMonthlyActivity** — tablica sadrži sumarne podatke (na razini mjeseca) o aktivnosti igrača na platformi za klađenje. Podaci su dostupni za razdoblje od siječnja 2023. do kolovoza 2023.
    - *player_id*: jedinstveni identifikator igrača
    - *transmonth*: mjesec u kojem su se dogodile promatrane transakcije
    - *sport_payin*: ukupan iznos uplata na sportsko klađenje za promatranog igrača u promatranom mjesecu

- *sport_win*: ukupan iznos dobitaka na sportsko klađenje za promatranog igrača u promatranom mjesecu
- *active_on_other_verticals*: označava aktivnost igrača na drugim igrama, različitim od sportskog klađenja (npr. Casino, Bingo). Igrač se smatra aktivnim ako je napravio barem jednu uplatu u promatranom mjesecu.
- *avg_payin_per_sport_ticket*: prosječna uplata na sportski listić za promatranog igrača u promatranom mjesecu
- *avg_ticketbets_count*: prosječan broj oklada koje igrač postavlja na sportski listić u promatranom mjesecu

3. **TicketBets** — detaljni podaci o uplatama sportskih listića u razdoblju od 1.5.2023. do 31.8.2023. Jedan redak tablice predstavlja jednu okladu na listiću. Jedan listić može se sastojati od jedne ili više oklada (nema gornje granice). Ovaj skup ukupno ima 2 773 123 oklada (redaka) raspoređenih na 602 046 listića. Tu se nalazi i ciljna značajka ovog problema: *is_rejected*.
   - *player_id*: jedinstveni identifikator igrača
   - *ticket_id*: jedinstveni identifikator sportskog listića
   - *vrijeme_uplate*: vrijeme uplate sportskog listića
   - *client_type*: platforma na kojoj je listić odigran ('MobileWebConsumer' ili 'WebConsumer')
   - *broj_betova_na_listicu*: broj odigranih oklada na listiću (može biti samo jedna ili više njih)
   - *event_name*: naziv utakmice/događaja na koji je uplaćena oklada
   - *event_start_date*: vrijeme početka utakmice/događaja
   - *prematch_live_bet*: oznaka tipa oklade — live (L) ili prematch (P); oklada live označava sve oklade koje su uplaćene za vrijeme trajanja utakmice, dok prematch označava sve oklade koje su uplaćene prije početka utakmice

- *system*: oznaka sistemskog listića (oznaka 1 za sistemski listić)
- *sport*: naziv sporta na koji je uplaćena oklada
- *category*: označava državu odigravanja natjecanja; može biti i *International*, ako natjecanje nije vezano za jednu državu
- *contest*: oznaka lige na koju se odnosi oklada
- *bettype_name*: oznaka tržišta (engl. *market*)
- *pick*: ishod na koji je uplaćena oklada; npr. za nogomet i market Golovi pick može biti 1 (domaći) ili 2 (gosti)
- *baseline*: pojedini ishodi klađenja zahtijevaju detaljniji odabir (npr. broj golova u utakmici će biti preko x), u tom slučaju taj x je definiran odabranom osnovnom ponudom
- *odds*: kvota na odabranoj okladi[1] u trenutku uplate listića
- *ukupna_uplata_na_listicu*: ukupan ulog s kojim je igrač ušao u okladu listića (jednak za sve oklade na istom listiću)
- *is_rejected*: oznaka je li listić odbijen ili uspješno uplaćen
- *rejectionground*: razlog odbijanja listića (SystemRejected ili RiskRejected)

4. **Payin by sport** — sadrži udio ukupne uplate na sportsko klađenje po svim sportovima. Podaci o uplatama prikupljeni su u razdoblju od godinu dana unazad (1.5.2022. – 30.4.2023.).

   - *Sport*: naziv sporta
   - *%GT Payin*: postotni udio uplate u ukupnoj uplati na sportsko klađenje

   Za tri najlikvidnija sporta (nogomet, košarka, tenis), dodana je raspodjela uplate po natjecanjima unutar tog sporta. Ovi podaci, zajedno s uplatom po sportu, omogućuju razlikovanje "jačine" pojedinih natjecanja na koje se igrači žele kladiti.

[1] Oklada je jedinstveno definirana kao kombinacija atributa: event-bettype-pick-baseline

5. **Payin by Contest Football/Basketball/Tennis** — sadrži udio ukupne uplate za odabrani sport po njegovim natjecanjima. Podaci o uplatama prikupljeni su u razdoblju od godinu dana unazad (1.5.2022. – 30.4.2023.) u odnosu na TicketBets tablicu.

    - *Sport*: naziv sporta
    - *contest*: naziv natjecanja
    - *category*: naziv države/kategorije u kojoj se održava natjecanje
    - *Payin%*: postotni udio uplate u ukupnoj uplati na sportsko klađenje

5.2. Priprema podataka

Prvi korak u procesu upravljanja rizikom je pregled i eksploratorna analiza dostupnih podataka. Priprema podataka uključuje čišćenje podataka, transformaciju postojećih i izgradnju novih značajki, spajanje podataka iz različitih tablica i agregaciju podataka na razinu listića. Kvalitetna priprema podataka ključna je za postizanje dobrih rezultata modela.

Za potrebe ovog projekta korišten je programski jezik Python (verzija 3.8) u interaktivnim Jupyter bilježnicama. Korišteni su sljedeći paketi: pandas za učitavanje i osnovnu manipulaciju podataka, matplotlib i seaborn za izradu vizualizacija, scikit-learn i imbalanced-learn za metode dubinske analize podataka.

5.2.1. Pregled i čišćenje podataka

Budući da su podaci podijeljeni u više tablica, prvo je potrebno zasebno pregledati te podatke u svrhu razumijevanja njihove strukture i sadržaja. Sve dok se rade manipulacije nad podacima, potrebno je provjeravati mijenja li se njihova kvaliteta.

Uz podatke dobiveni su i pripadni tipovi podataka u izvornom obliku (bazi podataka), čime je olakšan prvi korak pripreme podataka. Prilikom učitavanja podataka, zadani su odgovarajući tipovi podatka svakom stupcu iz .xlsx tablice izvora.

PODACI O OKLADAMA

Glavni izvor podataka nalazi se u tablicama TicketBets (posebna tablica za svaki od četiriju mjeseci). Nakon učitavanja i spajanja u jedinstvenu strukturu, skup podataka ima 2 773 123 primjeraka i 19 značajki. Sve te oklade podijeljene su na 602 046 listića, a listić s najviše oklada ih ima 36. U promatrana četiri mjeseca na sport su se kladila 6 532 igrača. Najviše oklada (više od 95%) na platformu za klađenje uplaćeno je s mobilnih uređaja.

Osnovne deskriptivne statistike (minimum) značajke *broj_betova_na_listicu* ukazuju na pogrešku u podacima. Postoje četiri retka koja imaju vrijednost te značajke 0, što u stvarnosti nije moguće. To su pokušaji klađenja uživo kojima većina vrijednosti nedostaje, a mogući uzrok tomu je da je ponuda uživo već bila istekla u trenutku pokušaja uplate. Ti primjeri uklonjeni su iz skupa podataka.

Igrači su stavljali oklade uživo u oko 42%, dok se preostalih 58% oklada odnosi na klađenje prije početka događaja. Svaka oklada ima i oznaku nalazi li se na sistemskom listiću ili ne. Samo malo manje od 6% oklada nalazi se na sistemskim listićima. Ta informacija nije baš relevantna jer se oznaka sistema odnosi na cijeli listić, a ne na pojedinu okladu na njemu. Ako se ta značajka promotri na razini listića, dobije se ukupno 2.5% sistemskih listića. U skupu podataka ne postoji informacija o oznaci sistema (koje kombinacije su dobitne za igrača), nego samo oznaka je li sistemski ili nije.

Igrači su se kladili na 2 389 natjecanja u 24 različita sporta. Tablica 5.1. prikazuje 5 sportova s najvećim brojem oklada, zajedno s brojem natjecanja i utakmica po svakom od njih. Samo nogomet pokriva oko 75% ukupnog broja oklada na sport, a zajedno s tenisom i košarkom nešto više od 95%.

sport	br. natjecanja	br. utakmica	br. oklada
nogomet	965	39 242	2 071 656
tenis	555	29 982	370 747
košarka	283	5 863	198 075
stolni tenis	39	12 460	28 734
bejzbol	10	1 096	27 666

Tablica 5.1. Najčešće igrani sportovi s brojem natjecanja, utakmica i oklada

Natjecanje s najviše oklada je Europska konferencijska liga (engl. UEFA Europa Conference League). Uzrok leži u razdoblju prikupljanja podataka. Od svibnja do kolovoza je smanjen opseg nogometnih natjecanja, osobito klupskih. Tablica 5.2. prikazuje razdiobu

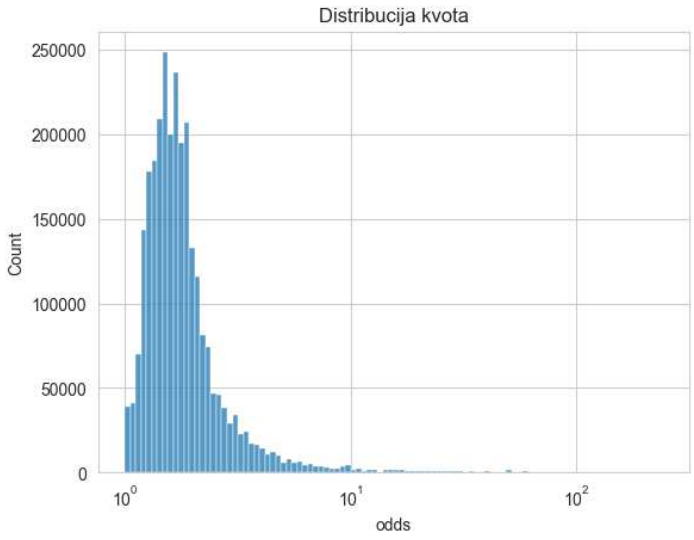
broja oklada na Konferencijsku ligu kroz mjesece koji se pojavljuju u skupu podataka. Kvalifikacije konferencijske lige igraju se u srpnju i kolovozu pa su zbog velikog broja utakmica u tom periodu izražene oklade na to natjecanje.

mjesec	broj oklada
5	5 599
6	645
7	33 550
8	82 142

Tablica 5.2. Broj oklada na Konferencijsku ligu po mjesecima

Značajke *bettype_name, pick* i *baseline* kategoričke su značajke koje zajedno definiraju okladu. Svaka od njih ima velik broj različitih vrijednosti (redom: 566, 454, 2 317), a nema prirodno definiranog poretka među njima kako bi se mogle pretvoriti u numeričke značajke. Zbog toga su te tri značajke uklonjene.

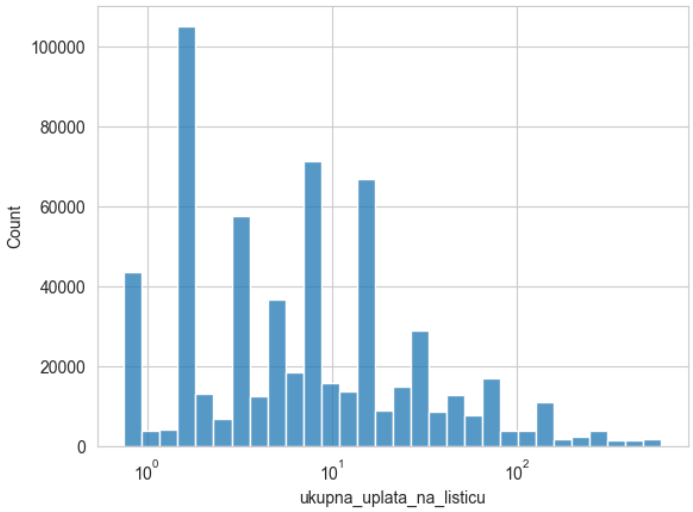
Distribucija kvota za pojedinu okladu izrazito je pozitivno zakrivljena. Prikazana je na slici 5.1., a vidi se da je velika većina kvota u intervalu $[1, 10]$. Ostale kvote imaju vrijednosti do 101, osim jedne stršeće vrijednosti za ovu značajku čija kvota iznosi 241.



Slika 5.1. Distribucija kvota (na razini oklade)

Značajku *ukupna_uplata_na_listicu* treba analizirati na razini listića, a ne oklade (trenutna granulacija podataka). Na slici 5.2. prikazana je razdioba na logaritamskoj skali. Vidi se da igrači najviše uplaćuju do iznosa 10, znatno manje do iznosa 100 i najmanje iznose veće od 100. Na histogramu su vidljivi "šiljci", odnosno iznosi koji se puno

češće uplaćuju (npr. cjelobrojni iznosi manji od 10 ili iznosi zaokruženi na 5 ili 10).



Slika 5.2. Distribucija ukupne uplate na listiću

Još preostaje pregledati značajke vezane za odbijanje listića — *is_rejected* i *rejectionground*. Prva značajka je binarna i označava je li listić odbijen, a druga sadrži razlog odbijanja. Obje značajke imaju jednake vrijednosti za sve oklade na istom listiću jer se ne može odbiti pojedina oklada na listiću, nego samo cijeli listić. U tablici 5.3. prikazane su razdiobe navedenih značajki. Iskazane postotke treba promatrati u svjetlu oklada, a ne listića (što je konačni cilj). Podaci iz tablice 5.3. daju naslutiti veliku razliku između broja odbijenih i uplaćenih listića.

is_rejected		rejectionground	
False	90.70%	NaN	90.70%
True	9.30%	SystemRejected	8.46%
		RiskRejected	0.84%

Tablica 5.3. Udjeli odbijenih oklada i razloga odbijanja

SystemRejected oznaka pridjeljena je okladama koje su automatizirano odbijene zbog neispravnih informacija (npr. opisani primjeri s 0 oklada na listiću) ili zbog ograničenja poslovanja (npr. prevelika uplata). Oklade s oznakom RiskRejected odbili su kladionički ručno i to su zapravo rizični listići koje je potrebno detektirati. U skladu s tim, iz skupa podataka treba ukloniti primjere koji imaju oznaku SystemRejected. Nakon uklanjanja tih točaka, značajka *rejectionground* postaje jednaka značajki *is_rejected* (za sve uplaćene listiće ima vrijednost NaN, a za sve odbijene RiskRejected) i treba ju ukloniti.

Sada skup podataka ima 2 538 504 primjera i 15 značajki, od kojih je 23 172 (0.91%) odbijenih primjeraka (oklada).

U skupu podataka preostalo je još 137 primjera koji imaju nedostajuće vrijednosti za značajke *event_name* i *event_start_date*. To su nestandardne oklade koje se ne odnose na jednu utakmicu, nego na cijelo natjecanje (npr. klađenje na osvajača svjetskog prvenstva prije njegova početka). One su dio listića s drugim okladama tako da se ne smiju ukloniti kako bi se očuvala cjelovitost informacije u podacima. S obzirom na to da će se ovi podaci agregirati, kasnije više neće biti tih nedostajućih vrijednosti pa se za sad ignoriraju (uz dodatan oprez pri agregaciji).

PODACI O IGRAČIMA

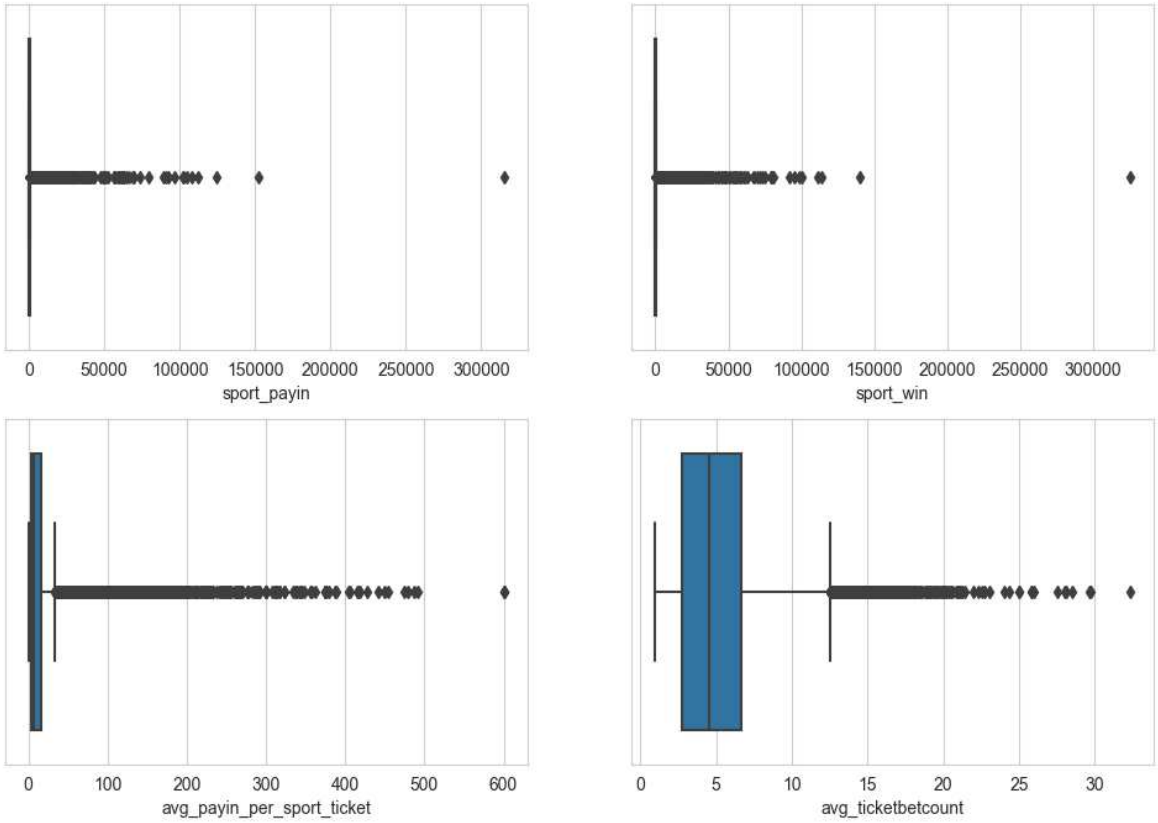
Za online klađenje potrebno je registrirati se na željenoj platformi kako bi se utvrdilo da se radi o stvarnim osobama. Dobiveni podaci odgovaraju stvarnim osobama, ali nisu dostupne informacije o njihovom identitetu.

Podaci o registraciji dobiveni su na temelju perioda aktivnosti igrača — dostupni su podaci za igrače koji su bili aktivni na platformi u promatranom razdoblju (1.1.2023. – 31.8.2023.). Takvih igrača ima 6532, svaki sa svojim jedinstvenim identifikatorom (player_id), vremenom registracije, datumom rođenja i gradom iz kojeg dolazi. Ti podaci ne znače ništa sami za sebe, ali koristit će se kasnije za izgradnju novih značajki. U ovom podskupu nema nedostajućih vrijednosti.

Podaci o prethodnoj aktivnosti igrača dostupni su za prvih osam mjeseci 2023. godine — za svakog igrača koji je bio aktivan u tom razdoblju podaci su agregirani na razini mjeseca. Postoji 30 369 takvih zapisa — za pojedinog igrača postoji zapis za mjesec u kojem je imao bilo kakvu uplatu na platformi za klađenje (na sportsko klađenje ili igre na sreću). Oni mogu poslužiti u prepoznavanju iskusnijih i vještijih igrača, koji češće pogađaju ishode na koje se klade. Takvi igrači mogu biti u nepoštenoj prednosti nad ostalima, stoga ih je dobro detektirati.

Ovi podaci će se iskoristiti za izgradnju novih značajki koje će odražavati prošlu aktivnost igrača — npr. redak o listiću koji je uplaćen u 6. mjesecu može sadržavati podatke iz ove tablice do 5. mjeseca (uključivo). Budući da ova tablica sadržava podatke i za 8. mjesec, a to je zadnji mjesec u podacima o okladama, oni se uklanjaju kako bi se

spriječilo potencijalno curenja podataka (u prošlu aktivnost ne mogu se uključiti podaci koji su nastali u isto vrijeme ili kasnije od trenutnih). Nakon ovog uklanjanja, u skupu podataka ostaje 25 526 redaka, od kojih 4 569 s nedostajućim vrijednostima značajki *avg_payin_per_sport_ticket* i *avg_ticketbetcount*. To su slučajevi kada igrač nije imao nijednu uplatu na sport u tom mjesecu, nego neku drugu aktivnost na platformi za klađenje. U slučaju kada igrač nije bio aktivan ni na nekoj drugoj vrsti klađenja, takvi se reci (njih 112) mogu ukloniti iz skupa jer ne sadrže korisnu informaciju i kako ne bi stvarali šum. Preostale nedostajuće vrijednosti zamjenjuju se konstantom 0, što u ovom slučaju odgovara stvarnom značenju nedostajuće vrijednosti.



Slika 5.3. Razdiobe numeričkih značajki podataka o prethodnoj aktivnosti igrača

Razdiobe numeričkih značajki prikazane su na slici 5.3. Sve četiri značajke imaju zakrivljenu razdiobu, dok *sport_payin* i *sport_win* imaju po jednu izrazito stršeću vrijednost. Ručnom provjerom utvrđeno je da je to isti redak, a vrijednosti svih numeričkih značajki prikazane su u tablici 5.4. Budući da nije sigurno sadrži li ova stršeća vrijednost bitnu informaciju, nije uklonjena iz podataka. Na kraju ovog koraka ovaj skup podataka ima 25 414 primjera i 7 značajki.

sport_payin	sport_win	avg_payin_per_sport_ticket	avg_ticketbetcount
314909.07	325372.71	283.19	1.05

Tablica 5.4. Stršeća vrijednost

DISTRIBUCIJE UPLATA PO SPORTOVIMA

Ovi "pomoćni" skupovi podataka služe za dobivanje uvida u popularnost pojedinog sporta, a za neke sportove i popularnost njihovih natjecanja. U sljedećem koraku cilj je iz ovih podataka izvući korisne informacije i pretočiti ih u nove značajke. Da bi to bilo moguće, potrebno je analizirati svaki od četiriju dostupnih skupova.

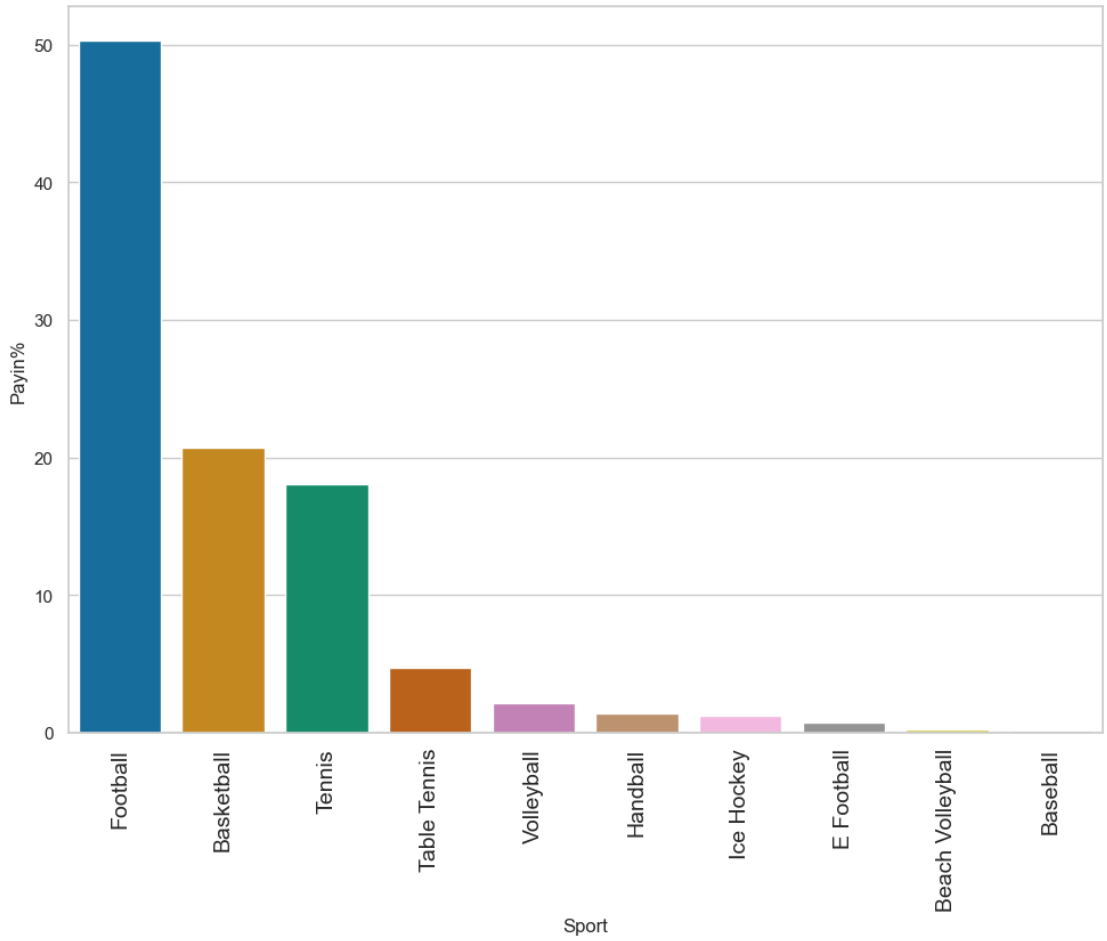
U prvoj datoteci nalaze se podaci o uplatama po svim sportovima. U promatranom razdoblju igrači su se kladili na 28 različitih sportova. Na slici 5.4. prikazan je stupčasti graf s postotkom od ukupne uplate[2] za 10 sportova s najviše uplata. Nogomet se značajno izdvaja od ostalih sportova, s oko 50% iznosa ukupne uplate na sportsko klađenje. Slijede ga košarka i tenis s po oko 20%, a stupići su vidljivi još za stolni tenis, odbojku, rukomet, hokej na ledu i E-nogomet (oko 5% i manje). Ostali sportovi[3] imaju neznatan udio u ukupnoj uplati: pojedinačno svaki od njih ima manje od 0.3%, a svi skupa osigurali su ukupno manje od 1%.

S obzirom na to da tri najpopularnija sporta nose gotovo 90% iznosa ukupnog klađenja na sport, za njih je dostupno još podataka. Uplata po svakom od sportova podijeljena je na sva dostupna natjecanja. Za svaki od tih sportova napravljena je slična analiza. Za svaki sport dodan je redundantni stupac *Payin%* koji izražava uplatu u postotnom obliku. Intuitivnije je značenje stupca i jednostavnije je baratati njime jer je većina izvornih vrijednosti izrazito mala.

Za nogomet postoje zapisi o 1 459 natjecanja u 165 država. Naziv natjecanja ne određuje jednoznačno pojedino natjecanje, nego u kombinaciji s državom u kojoj se održava (npr. postoje 34 naziva natjecanja 'Premier League'). Postoji 9 redaka s natjecanjima koji u svom nazivu sadrže znakovni niz 'FALŠI', što vjerojatno označava neku grešku u podacima. Provjerom naziva natjecanja u TicketBets tablici, nije pronađeno nijedno natjecanje koje sadrži taj znakovni niz. S obzirom na to da ovaj skup služi samo kao po-

[2]Ukupna uplata na sportsko klađenje u razdoblju 1.5.2022. – 30.4.2023.

[3]Ostali sportovi: odbojka na pijesku, bejzbol, vaterpolo, pikado, američki nogomet, futsal, badminton, biljar, boks, ragbi, formula 1, košarka 3x3, nogomet na pijesku, australski nogomet, dvoranski hokej, kriket, skijanje, hokej na travi, curling i kategorija 'Specials'

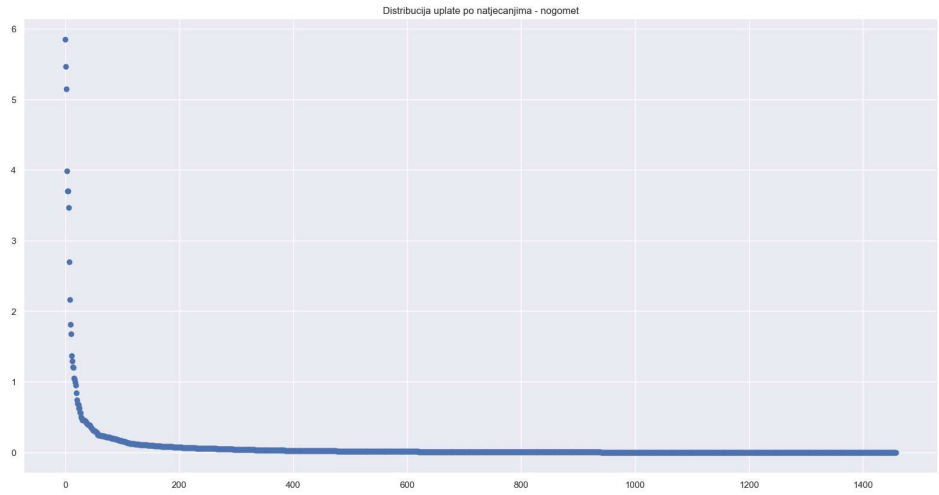


Slika 5.4. Razdioba uplate na sportsko klađenje za 10 sportova s najviše uplata

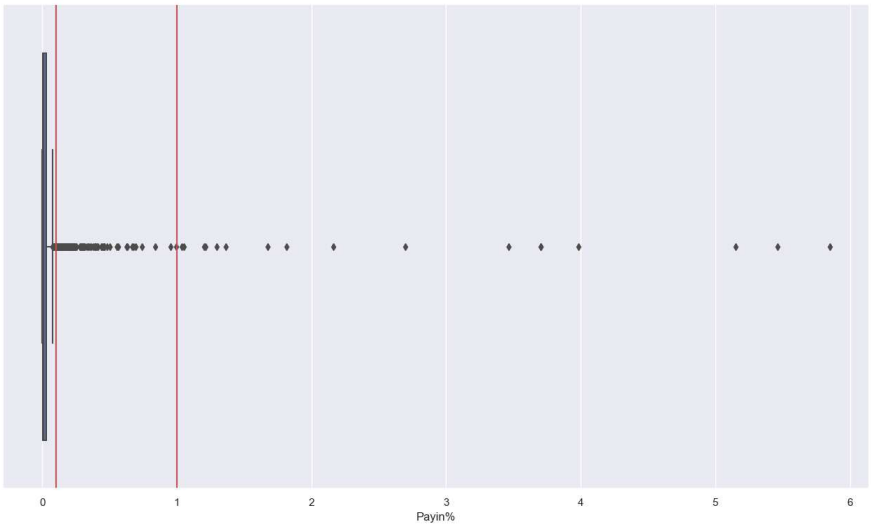
moć za izgradnju novih značajki, a u glavnom skupu podataka se ne pojavljuju navedene greške, tih 9 redaka nisu uklonjeni kako bi se očuvala mjera udjela[4].

Postotci uplate za sva nogometna natjecanja prikazani su na slici 5.5. — svaka točka predstavlja jedno natjecanje, a pripadni postotak u ukupnoj uplati prikazan je na y-osi. Većina natjecanja ima jako mali postotak u ukupnoj uplati ($< 1\%$), što se vidi iz prikaza na slici 5.5. Eksponencijalna razdioba značajke postotka uplate je prikazana box-plotom na slici 5.6. Crvene vertikalne linije postavljene su na vrijednosti 0.1% i 1% i otprilike odjeljuju gušće i rjeđe zgusnute stršeće vrijednosti od ostalih podataka. Najmanje je natjecanja koja nose veći dio uplate (jako stršeće vrijednosti). Na temelju toga u potpoglavlju 5.2.2. će se natjecanja podijeliti u grupe i izgraditi nove značajke kako bi se očuvala informacija o pojedinim natjecanjima na listiću.

[4]Sva natjecanja zajedno daju 100% uplate u nogometu; ako se ti reci uklone, postotci se ne mogu preračunati i suma ne bi odgovarala 100%.



Slika 5.5. Prikaz udjela u ukupnoj uplati za pojedino nogometno natjecanje



Slika 5.6. Distribucija postotka uplate na nogometna natjecanja

Košarkaških natjecanja ima manje nego nogometnih, njih 501. Distribucija postotka uplate po natjecanju slična je onoj za nogomet. Razlika je što u košarci tri natjecanja više odstupaju od ostalih (imaju veći postotak uplate) — NBA donosi oko 19% uplaćenog iznosa na košarku, Euroliga 14%, a ABA liga nešto manje od 8%. Kod košarke, kao i kod nogometa, oznaka države (*category*) sadrži korisnu informaciju jedino u kontekstu jednoznačnog definiranja natjecanja.

Dostupno je 2 464 teniskih natjecanja u skupu podataka. Postoje samo tri različite kategorije — sva natjecanja su međunarodna osim dvaju, koja imaju pogrešnu oznaku. I to su međunarodni turniri kojima je unesena država u kojoj se održavaju umjesto kategorije International. Te se dvije pogreške mogu ignorirati jer je ta značajka (category)

konstantna.

Naziv gotovo svih natjecanja sadrži oznaku natjecanja: ATP, WTA, ITF. Preostala natjecanja su Grand Slamovi (GS) ili neka manje poznata natjecanja. Prema tim odrednicama razine natjecanja, promijenjena je značajka *category* tako da odražava kategoriju turnira. Efektivno je uklonjena konstantna značajka i izgrađena nova: s informacijom o vrsti turnira.

U tablici 5.5. prikazana je raspodjela broja turnira (značajka contest) i udio u ukupnoj uplati po kategoriji natjecanja. Najviše natjecanja postoji na ITF razini, što je ujedno i najniža razina[5], ali najveći postotak uplate nose turniri ATP razine. To su profesionalni turniri u muškoj konkurenciji koja je općenito više praćena. Osim toga, značajno većoj uplati doprinosi to što se ti turniri odigravaju tijekom cijele godine, za razliku od Grand Slam turnira. Oni nose manje od 3% ukupne uplate, ali to su četiri najveća i najjača turnira u godini. Postoji 20 GS različitih natjecanja jer se razlikuju muška i ženska pojedinačna konkurencija te muški, ženski i miks parovi.

category	contest	Payin%
ATP	499	51.48
ITF	1 633[6]	20.27
WTA	156	15.62
Grand Slam	20	9.70
Other	155	2.92

Tablica 5.5. Distribucija turnira i uplata po kategorijama teniskih turnira

5.2.2. Izgradnja značajki i spajanje izvora

Podaci se nalaze u različitim datotekama i sadrže različitu razinu detalja. Problem klasifikacije odnosi se na cijeli listić, a ne na pojedinu okladu na njemu. Cilj je ovog koraka dobiti skup podataka jednake granulacije, odnosno da svaki entitet bude predstavljen jednim retkom i s istim značajkama. Procesi izgradnje značajki i spajanja izvora u ovom su projektu isprepleteni jer neke nove značajke ovise samo o jednom izvoru, a neke se izgrađene na temelju više tablica.

Prvi korak uključuje učitavanje osnovnog skupa podataka (TicketBets), kojem se postupno dodaju ostali skupovi podataka kako bi ga obogatili. Najprije se dodaju podaci o

[5]osim kategorije Other; ona je samo dodijeljena preostalim turnirima
[6]zapravo ima 1634 retka; jedan naziv se ponavlja dvaput

sportovima i natjecanjima jer njihove značajke treba izgraditi prije agregacije, a kasnije se dodaju podaci iz skupova podataka o igračima. Prema razdiobama uplata opisanih u 5.2., izgrađene su kategoričke značajke koje dijele sportove, odnosno natjecanja u grupe. Značajka *payin_cat* definirana je sukladno opisu razdiobe (slika 5.2.) i ima četiri različite vrijednosti:

- *top_sport_payin*: nogomet,
- *sport_high_payin*: košarka i tenis,
- *sport_medium_payin*: stolni tenis, odbojka, rukomet, hokejna ledu i E-nogomet,
- *sport_low_payin*: ostali.

U tablici 5.6. prikazane su nove značajke za natjecanja triju najpopularnijih sportova: njihov naziv, granice značajke *Payin%* (na temelju koje su izgrađene) za pojedinu vrijednost i broj natjecanja koja uključuju. Teniska natjecanja imaju definirane razine jačine turnira i prirodno je iskoristiti tu podjelu za definiranje novih značajki. U tom slučaju ne treba se voditi isključivo uplatama jer su Grand Slam turniri najpoznatiji i predstavljaju najviši rang natjecanja u tenisu, ali zbog malog broja turnira godišnje, uplata je manja. Natjecanja u tenisu zbog toga su podijeljena u dvije kategorije: pro (GS, ATP i WTA turniri) i other (ITF i Other turniri).

naziv značajke	vrijednost značajke	granice Payin%[7]	br. natjecanja
football_payin_cat	football_high_payin	$\langle 1, 100]$	17
	football_medium_payin	$\langle 0.1, 1]$	133
	football_low_payin	$\langle 0, 0.1]$	1309
basketball_payin_cat	basketball_high_payin	$\langle 1, 100]$	14
	basketball_medium_payin	$\langle 0.1, 1]$	108
	basketball_low_payin	$\langle 0, 0.1]$	379
tennis_cat	tennis_pro	-	675
	tennis_other	-	1789

Tablica 5.6. Definicija novih kategoričkih značajki vezanih za natjecanja i opis njihovih vrijednosti

Nakon određivanja vrijednosti kategoričkih značajki za svaki sport i natjecanje, napravljeno je preslikavanje svih vrijednosti u zasebne binarne značajke (One-Hot Encoding), a zatim spajanje s TicketBets tablicom. Dodavanjem novih značajki broj dimenzija

[7] postotak se promatra za svaki sport pojedinačno

skupa porastao je na 27. Novi stupci, s kategorijama po uplati, imaju vrijednost NaN u slučaju kada se radi o drugom sportu. Na mjesto nedostajućih vrijednosti treba staviti False u odgovarajuće stupce (npr. nogometno natjecanje će treba imati vrijednost False u stupcima koji se tiču košarke i tenisa).

Sljedeći je korak sažimanje podataka na razinu listića. Sada se mogu ukloniti tekstualne značajke vezane za sportove i natjecanja jer su na temelju njih izgrađene nove numeričke značajke. Uklanjaju se sljedeće značajke: *sport, category, contest, event_name, event_start_date*. Ovime je riješen i problem nedostajućih vrijednosti jer su jedino nedostajale vrijednosti za 137 utakmica (*event_name, event_start_date*). Podaci se prvo grupiraju po vrijednosti *ticket_id*, a zatim se ovisno o značajki primjenjuju različite funkcije. Značajke koje imaju istu vrijednost za sve oklade na listiću koriste funkciju 'first' (odabire se prvo pojavljivanje značajke u grupi), a to su: *player_id*, *vrijeme_uplate*, *client_type*, *broj_betova_na_listicu* (*ticket_bets_count*), *prematch_live_bet*, *system_ticket*, *is_rejected*, *ukupna_uplata_na_listicu*. Kvote pojedinih oklada pomnožene su i daju ukupnu kvotu listića (*total_ticket_odds*). Značajke s kategorijama sportova i natjecanja sažete su tako da odražavaju udio oklada u toj kategoriji u odnosu na sve oklade (pomoću funkcije srednje vrijednosti) — npr. ako su na listiću uplaćene dvije oklade na nogomet i dvije na košarku, vrijednost značajki *top_sport_payin_rate* i *sport_high_payin_rate* bit će 0.5. Nakon sažimanja, dodane su još tri nove značajke:

- *potential_payout*: Iznos potencijalne isplate igraču u slučaju da je listić dobitan (formula 2.1).
- *has_low_payin_group_bet*: Binarna varijabla koja označava postoji li na listiću oklada iz bilo koje grupe s rijetkim uplatama. Da bi varijabla imala vrijednost True, barem jedna od četiriju značajki s udjelima (*sport_low_payin_rate*, *football_low_payin_rate*, *basketball_low_payin_rate*, *tennis_other_rate*) mora imati vrijednost veću od 0.
- *has_similar_ticket*: "Sličan/identičan listić" definiran je kao listić koji ima isti broj oklada, isti ukupni koeficijent i istu potencijalnu isplatu. Uspoređuju se uplaćeni listići za svakog pojedinog igrača, a ne svi (redom uplaćeni) listići u skupu podataka.

Trenutni skup podataka sadrži 554 872 redaka (listića) i 25 značajki. Još je potrebno

pridružiti podatke o igračima, a podacima o aktivnosti igrača dodane su neke nove značajke kako bi se što bolje opisale karakteristike igrača. Jedan od ključnih pokazatelja aktivnosti igrača jest broj uplaćenih listića, a iz dostupnih podataka računa se na sljedeći način:

$$ticket\_cnt = \frac{sport\_payin}{avg\_payin\_per\_sport\_ticket}. \tag{5.1}$$

Vrijednost NaN pojavljuje se ako igrač nije bio aktivan u navedenom mjesecu (i brojnik i nazivnik jednaki su nuli), ali one se mogu bez gubitka informacije zamijeniti konstantom 0.

Sljedeći problem koji se pojavio je taj što nisu svi igrači bili aktivni svaki mjesec, a za dosljedan izračun novih značajki potrebno je u skupu podataka imati zapis za svakog registriranog igrača za promatrane mjesece. Za svaki mjesec u kojem igrač nije bio aktivan numeričkim značajkama se pridružuje vrijednost 0, a bool varijablama vrijednost False. Nakon toga mogu se izračunati sljedeće značajke za mjesece 5 – 8 i ukloniti originalne značajke na temelju kojih su nastale[8]:

- *active_on_sport_lm* — je li igrač bio aktivan u sportskom klađenju u tekućem mjesecu (True/False)
- *win_rate_lm* — postotak dobitka u odnosu na uloženi iznos igrača:

$$win\_rate = \frac{sport\_win - sport\_payin}{sport\_payin} \tag{5.2}$$

- *sport_payin_lm4* — ukupan ulog igrača u sportskom klađenju u posljednja 4 mjeseca
- *sport_win_lm4* — ukupan dobitak u sportskom klađenju u posljednja 4 mjeseca
- *tickets_lm4* — ukupan broj odigranih listića u posljednja 4 mjeseca
- *active_on_sport_lm4* — aktivnost igrača (br. mjeseci u kojima je bio aktivan) u sportskom klađenju posljednja 4 mjeseca (0–4)
- *active_on_other_verticals_lm4* — aktivnost igrača na drugim igrama na sreću pos-

[8] Ukoliko se originalne značajke ne bi uklonile došlo bi do curenja podataka; ne uklanjaju se značajke *player_id* i *transmonth*.

ljednja 4 mjeseca (0–4)

Prilikom ovog koraka pojavila su se dva primjera koja imaju beskonačnu vrijednost *win_rate_lm*. To su rubni slučajevi kada igrač u nekom mjesecu nije bio aktivan (nije imao uplatu), ali je na temelju uspješne aktivnosti s kraja mjeseca koji prethodi osvojio neki iznos. Budući da postoje samo dva takva primjera u ovom skupu, a u skupu podataka s listićima postoji samo jedan listić koji im odgovara, ti se reci mogu ukloniti iz obaju skupova. To neće uzrokovati velik gubitak informacije jer je to jedan zanemariv slučaj. Još jedan problem sa značajkom *win_rate_lm* je što se vrijednost 0 pojavljuje u dva slučaja čije se značenje ne može razlikovati:

1. igrač je uložio i osvojio jednak iznos (brojnik iz formule 5.2 iznosi 0),
2. igrač nije bio aktivan u prošlom mjesecu (prilikom računanja nastala je vrijednost NaN pa zamijenjena s 0).

Drugi je slučaj češći, ali nema druge smislene vrijednosti kojom bi se mogle zamijeniti nedostajuće vrijednosti, a bilo ih je previše da bi se potpuno uklonile.

Sada kad su sve značajke izgrađene, mogu se pridodati temeljnom skupu podataka s listićima. Podaci o aktivnosti igrača su dodani tek nakon agregacije kako bi taj proces bio manje računalne složenosti. Nakon spajanja podataka o registraciji, iz njih su izgrađene još tri značajke:

- *days_since_registration* — broj dana proteklih od registracije igrača do trenutka obrade listića
- *months_since_registration* — broj mjeseci proteklih od registracije igrača do trenutka obrade listića
- *age* — starosna dob igrača.

Prve dvije značajke izražavaju gotovo istu stvar, ali nije sigurno koja bi značajka mogla biti korisnija (prva poprima više različitih vrijednosti od druge) pa su obje ostavljene.

Na samom kraju transformacije podataka još je potrebno prilagoditi kategoričke značajke *client_type* i *prematch_live_bet*. To su tekstualne značajke s po dvije različite vrijed-

nosti i mogu se pretvoriti u binarne značajke bez gubitka informacije. Kako bi nazivi što bolje odražavali sadržanu informaciju, značajke su preimenovane u *client_type_mobile*, odnosno *prematch_ticket*. Trenutni podatkovni skup sadrži podatke o 554 871 sportskih listića opisanih s 45 značajki.

Prije primjene bilo kakvih metoda potrebno je odvojiti podatke u skup za učenje i za testiranje kako se ne bi unijela pristranost i došlo do boljih rezultata modela nego što oni zapravo jesu. U skup za učenje dodani su svi listići koji su pristigli na platformu u razdoblju 1.5.2023. – 31.7.2023. U testnom skupu preostaju listići iz kolovoza. Distribucija listića i ciljne značajke po mjesecima prikazana je u tablici 5.7. Omjer skupova za učenje i testiranje je 70:30, a oba skupa imaju približno jednak udio pozitivnih primjera ciljne značajke. Ciljna značajka ima visok stupanj nebalansiranosti: omjer nebalansiranosti ovog problema približno je jednak 100:1.

	mjesec	br. listića	odbijeni listići	udio odbijenih listića
skup za učenje	5	169530	1294	0.76%
	6	114635	2098	1.83%
	7	106674	2054	1.93%
	Σ	390839	5446	1.39%
testni skup	8	164032	2388	1.46%

Tablica 5.7. Razdioba ciljne značajke u skupovima za učenje i testiranje

Vizualizacije razdioba svih značajki skupa za učenje (osim identifikatora *ticket_id* i *player_id*) nalaze se u dodatku A . Zbog velike zakrivljenosti razdioba i potencijalnih stršećih vrijednosti dodatno su provjerene značajke *win_rate_lm*, *total_ticket_odds* i *potential_payout*. Prva među njima ima jednu stršeću vrijednost (265.33; iduća najveća vrijednost iznosi 34.22) koja se pojavljuje 21 put. Niti jedan od listića sa stršećom vrijednosti *win_rate_lm* nije odbijen — igrač je u prethodnom mjesecu ostvario velik dobitak u odnosu na uložen novac, ali to je izniman slučaj. Ti primjeri uklonjeni su iz skupa za učenje. Značajke *total_ticket_odds* i *potential_payout* imaju jako velike raspone vrijednosti, ali i jako zakrivljene razdiobe. Kod tih značajki ne može se izdvojiti jedna (ili nekoliko) stršećih vrijednosti. Velike vrijednosti tih značajki nisu neobične, nego ovise o broju oklada na listiću i o vjerojatnostima, odnosno kvotama pojedinih ishoda.

Značajke koje su nebitne, redundantne ili irelevantne treba ukloniti iz podataka: *ticket_id*, *vrijeme_uplate*, *client_type_mobile* (konstanta), *player_id*, *city* (identifikator mjesta),

date_of_birth, *registration_date* (datumi; izgrađene su numeričke značajke), *month* (pomoćna značajka za spajanje podataka iz različitih izvora). Preostalo je 37 značajki i 390 818 primjeraka u skupu za učenje.

5.2.3. Skaliranje značajki

Brojni modeli strojnog učenja daju bolje rezultate ako ne postoje velike razlike u amplitudama značajki. Najjednostavniji je pristup standardizacija svih značajki, ali u slučaju jako zakrivljenih i eksponencijalnih razdioba takva transformacija možda neće biti dovoljno dobra.

Ovisno o razdiobi značajke, primijenjena je jedna od dviju transformacija: standardizacija ili transformacija kvantila. Binarne značajke nisu transformirane. Standardizirane su sljedeće značajke: *days_since_player_registration*, *months_since_player_registration*, *player_age*, *active_on_sport_lm4*, *active_on_other_verticals_lm4*.[9] Obje transformacije implementirane su u Python paketu sklearn.preprocessing.

Ovaj je korak evaluiran na dvama modelima: Bayesov klasifikator i slučajna šuma kako bi se usporedio značaj transformacija na različitim vrstama modela (generativni, odnosno stablasti).

5.3. Odabir značajki i modela

U prethodnom koraku izgrađene su brojne značajke s ciljem da budu što relevantnije za klasifikaciju rizičnih listića. S druge strane, ako se podaci prostiru u puno dimenzija oni postaju rijetki (udaljeni). S porastom dimenzionalnosti potreban je puno veći broj primjera za modeliranje problema. Budući da se rijetko može nabaviti još podataka, smanjuje dimenzionalnost podataka odabirom samo najrelevantnijih značajki.

Isprobane su dvije metode za odabir najrelevantnijih značajki: rangiranje i filtriranje značajki mjerom zajedničke informacije i odabir rekurzivnom eliminacijom značajki (RFE) modelom slučajne šume. Prilikom filtriranja značajke su rangirane po mjeri zajedničke informacije, a zatim je optimalan broj značajki (3 – 20) koje treba uzeti u obzir određen unakrsnom validacijom. Unakrsna validacija napravljena je tako da je skup za

[9] Na preostale nebinarne značajke primijenjena je transformacija kvantila.

učenje podijeljen na 5 dijelova — svaki od njih ima jednake omjere ciljne klase kao cijeli skup za učenje (engl. *stratified 5-fold cross validation*). Evaluacija modela provedena je mjerom F2 (formula 4.6; $\beta = 2$). Minimalan broj značajki koji se može odabrati rekurzivnom eliminacijom postavljen je na tri.

S obzirom na to da je ciljna klasa jako nebalansirana, proveden je postupak ponovnog uzorkovanja kombnacijom metoda SMOTE za naduzorkovanje i Tomekovih poveznica za poduzorkovanje. Nakon ovoga koraka skup za učenje postao je dvostruko veći, s podjednakim udjelima primjera obiju klasa.

Nakon svakog koraka trenirani su modeli Bayesov klasifikator i slučajna šuma kako bi se mogli usporediti njihovi rezultati. Za učenje Bayesova klasifikatora, ulazni podaci su skalirani prethodno opisanom transformacijom (5.2.3.). Za model slučajne šume skaliranje podataka nije značajno, stoga ulazni podaci nisu transformirani.

Za optimizaciju hiperparametara modela također je korištena 5-struka unakrsna validacija. Za Bayesov klasifikator jedini hiperparametar je *var_smoothing*[10]. Pretragom po rešetci (engl. *grid search*) odabire se najbolja od ponuđenih vrijednosti: $10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}$. Model slučajne šume ima više hiperparametara, a podešeni su sljedeći s nekom od ponuđenih vrijednosti:

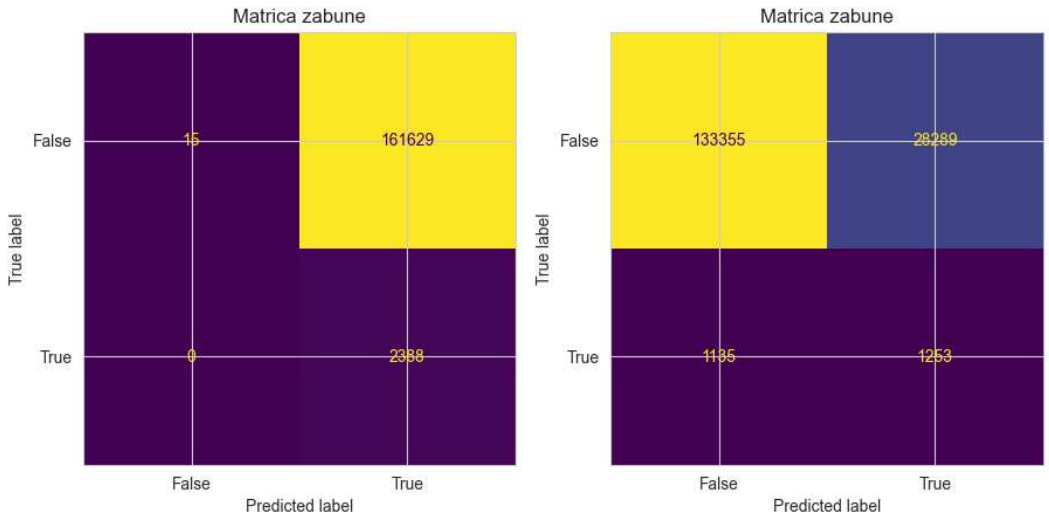
- broj stabala u šumi (n_estimators) — 50, 100 ili 200
- maksimalna dubina pojedinog stabla (max_depth) — 5, 10, 15, 20 ili 25
- minimalan broj primjera potrebnih za nastavak grananja čvora (min_samples_split) — 2, 5, 10 ili 20
- udio podataka koji se koriste za učenje svakog stabla (max_samples) — 0.1, 0.2, ..., 0.9

[10] Podešava varijance značajki zbog postizanja numeričke stabilnosti

6. Rezultati i rasprava

U ovom poglavlju prikazani su dobiveni rezultati za različite kombinacije primijenjenih metoda dubinske analize podataka. Nakon primjene svake metode, napravljena je evaluacija i usporedba s rezultatima prethodnog koraka.

Na početku su naučeni i evaluirani modeli korištenjem svih značajki, skaliranih i neskaliranih. Na slici 6.1. prikazane su matrice zabune za Bayesov klasifikator[1]. U prvom slučaju (6.1.b) klasifikator ne uspijeva naučiti razliku između klasa i predviđa gotovo sve listiće iz skupa za testiranje kao rizične. Iako rizičnih listića ima samo oko 1%, zbog evaluacije metrikom F2 (koja daje veću težinu odzivu), prevladala je pozitivna klasa u predviđanju. Nakon skaliranja značajki dobiju se nešto bolji rezultati — većina listića ne smatra se rizičnima, ali ne detektira se otprilike polovina rizičnih listića. Postoji i velik broj lažno pozitivnih primjera, a vrijednosti točnosti, preciznosti, odziva i F2 mjere nalaze se u tablici 6.3.

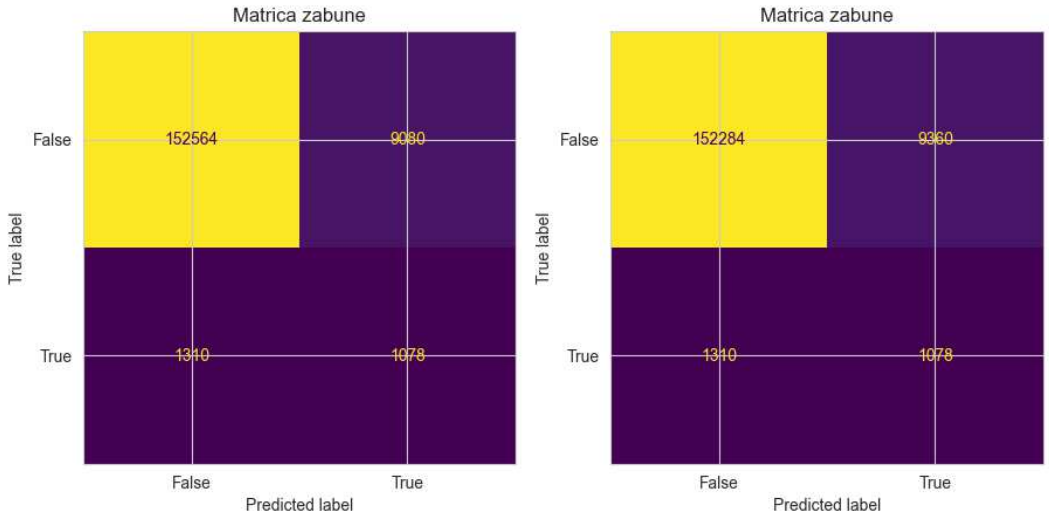


(a) Originalne značajke **(b)** Skalirane značajke

Slika 6.1. Matrice zabune za Bayesov klasifikator sa svim značajkama

[1]U tablicama se koristi oznaka NB (engl. *Naive Bayes*)

Kod modela slučajne šume[2] dobiveni rezultati gotovo se ne razlikuju, što je u skladu s očekivanjima ako se uzme u obzir vrsta modela. U odnosu na prvi model, ovaj bolje predviđa negativnu klasu (nema toliko lažno negativnih primjera), ali osjetljivost i dalje iznosi približno 50%. Matrice zabune prikazane su na slici 6.2., a tablica 6.3. prikazuje izračunate metrike za sve navedene kombinacije u ovom koraku.



(a) Originalne značajke **(b)** Skalirane značajke

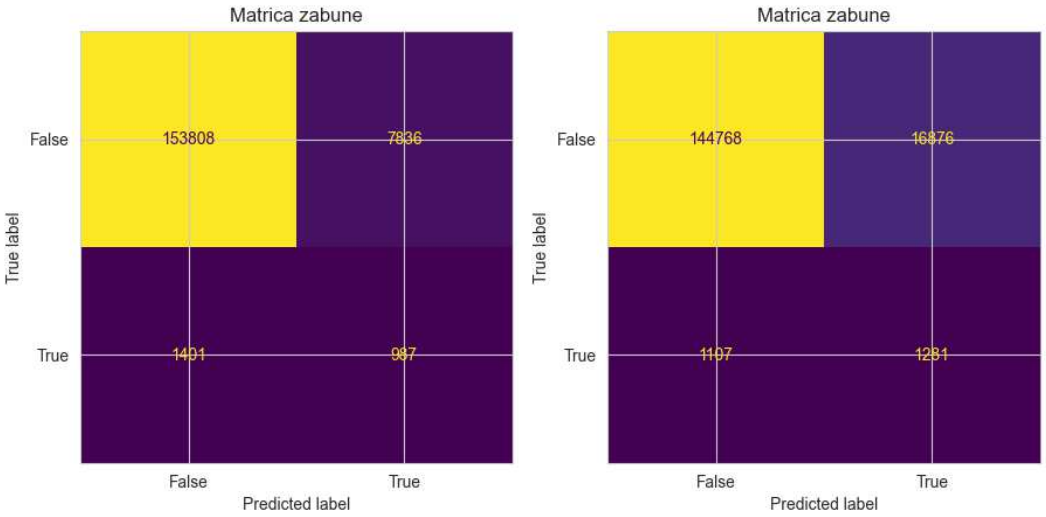
Slika 6.2. Matrice zabune za model slučajne šume sa svim značajkama

		točnost	preciznost	odziv	F2 mjera
NB	originalne značajke	0.0146	0.0146	1	0.0688
	skalirane značajke	0.8206	0.0424	0.5247	0.1603
RF	originalne značajke	0.9367	0.1061	0.4514	0.2735
	skalirane značajke	0.9346	0.1074	0.4782	0.2829

Tablica 6.1. Evaluacijske metrike modela

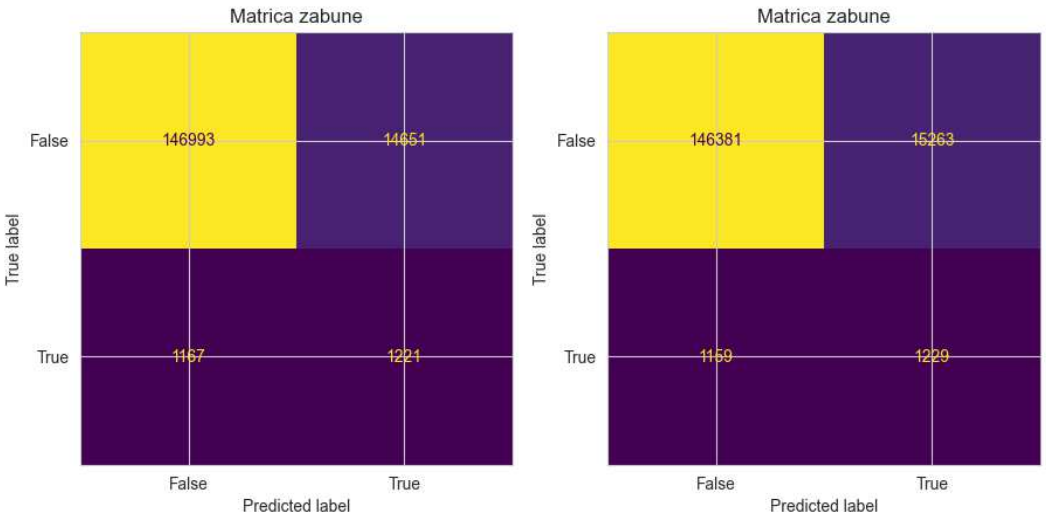
Prilikom filtriranja značajki mjerom zajedničke informacije, za unakrsnu validaciju prilikom odabira optimalnog broja značajki, korišten je model Bayesovog klasifikatora. Zbog toga su značajke prethodno skalirane kao što je već opisano. Dobiven je podskup od 7 značajki: *sport_medium_payin_rate*, *sport_low_payin_rate*, *basketball_high_payin_rate*, *basketball_medium_payin_rate*, *basketball_low_payin_rate*, *active_on_other_verticals_lm* i *tennis_other_rate*. Te su značajke najviše povezane s ciljnom značajkom (promatrajući svaku značajku zasebno).

[2]U tablicama se koristi oznaka RF (engl. *Random Forest*)



(a) Nebalansirani skup **(b)** Balansirani skup

Slika 6.3. Matrice zabune za model NB s odabranim podskupom značajki filtriranjem



(a) Nebalansirani skup **(b)** Balansirani skup

Slika 6.4. Matrice zabune za model RF s odabranim podskupom značajki filtriranjem

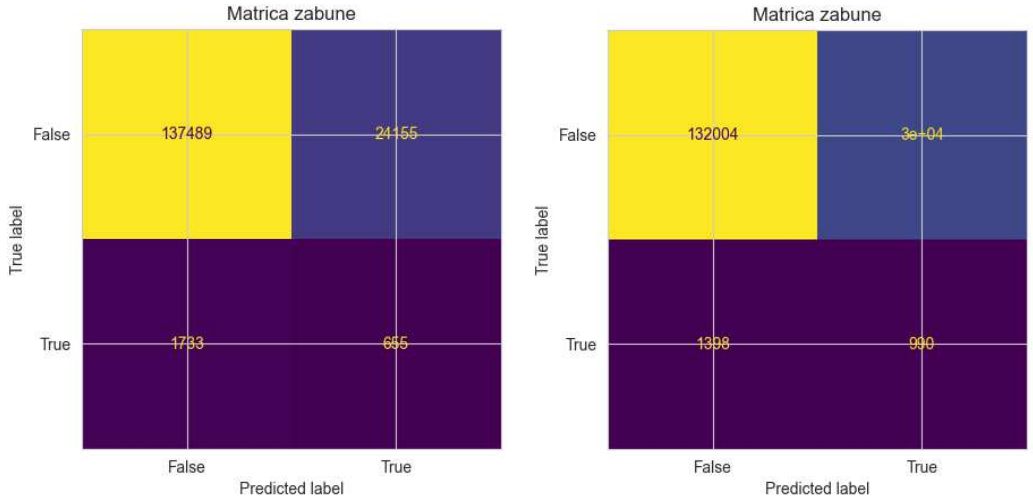
		točnost	preciznost	odziv	F2 mjera
NB	nebalansirani podskup	0.9437	0.1119	0.4133	0.2686
	balansirani podskup	0.8904	0.0706	0.5364	0.2312
RF	nebalansirani podskup	0.9036	0.0769	0.5113	0.2401
	balansirani podskup	0.8999	0.0745	0.5147	0.2359

Tablica 6.2. Evaluacijske metrike modela — filtriranje značajki mjerom zajedničke informacije

U usporedbi sa skupom svih značajki, Bayesov klasifikator daje nešto bolje rezultate, a model slučajne šume malo lošije rezultate koristeći filtrirani podskup značajki. Korištenje metoda za ponovno uzorkovanje nije ostvarilo poboljšanje modelima. Naprotiv,

modeli koji su učeni nad balansiranim skupovima, koji bi trebali biti pogodniji za učenje, u ovom slučaju daju nešto lošije rezultate.

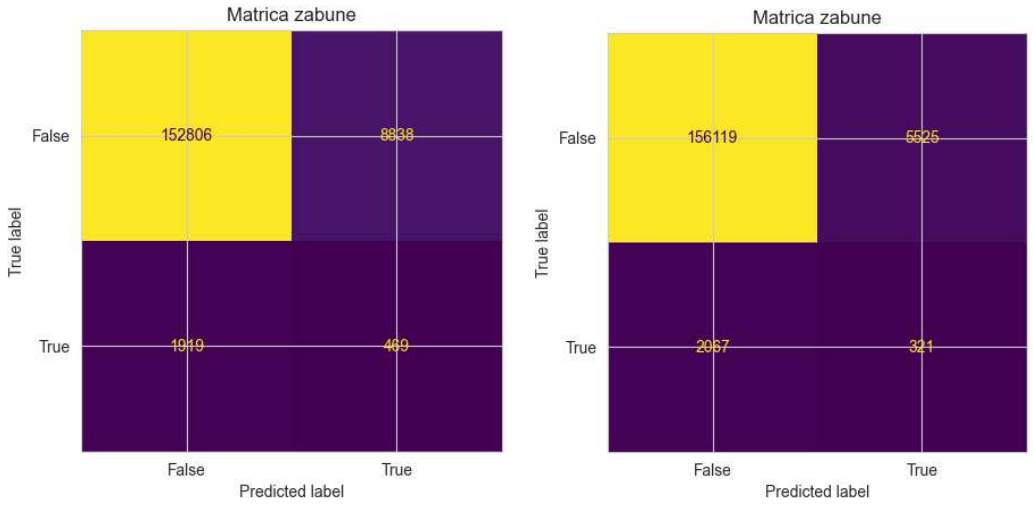
Rekurzivnom eliminacijom na skupu za učenje najboljim se pokazao podskup triju značajki: *total_ticket_odds*, *avg_payin_per_sport_ticket_lm* i *avg_ticketbetcount_lm*. Nakon izdvajanja odabranih značajki, trenirani su modeli na nebalansiranom i balansiranom skupu podataka. Podaci za učenje korišteni u Bayesovom klasifikatoru skalirani su, a slučajna šuma je trenirana na podacima s izvornim vrijednostima. Balansirani skup dobiven je ponovnim uzorkovanjem skupa s trima odabranim značajkama. Rezultati su prikazani na jednak način kao i u prethodnim koracima, prvo za model Bayesova klasifikatora pa za slučajnu šumu.



(a) Nebalansirani skup **(b)** Balansirani skup

Slika 6.5. Matrice zabune za model NB s odabranim podskupom značajki metodom RFE

Smanjivanjem dimenzionalnosti skupa odabirom triju predloženih značajki, dodatno se smanjila mogućnost modela da uspješno prepozna rizične listiće. Svaka od četiriju kombinacija odabranih značajki i modela je lošija u odnosu na klasifikatore učene na svim značajkama. Podskup značajki dobiven metodom rekurzivne eliminacije nije optimalan i ne sadrži dovoljno informacije za kvalitetnu detekciju rizičnih sportskih listića.



(a) Nebalansirani skup **(b)** Balansirani skup

Slika 6.6. Matrice zabune za model RF s odabranim podskupom značajki metodom RFE

		točnost	preciznost	odziv	F2 mjera
NB	nebalansirani podskup	0.8422	0.0264	0.2743	0.0953
	balansirani podskup	0.8108	0.0323	0.4146	0.1232
RF	nebalansirani podskup	0.9344	0.0504	0.1964	0.1243
	balansirani podskup	0.9537	0.0549	0.1344	0.1042

Tablica 6.3. Evaluacijske metrike modela — odabir značajki rekurzivnom eliminacijom

Najbolje rezultate daje model slučajne šume sa svim značajkama, koje su prethodno skalirane. Metoda filtriranja ne daje poboljšanje rezultata jer se važnost značajki određuje neovisno za svaku značajku — tim pristupom ne mogu se uhvatiti zavisnosti među značajkama. Rekurzivnom eliminacijom značajki dobije se podskup s premalo informacija o ukupnom problemu i modeli ne mogu naučiti iz tih podataka. Možda bi bolje rezultate dala kombinacija RFE i generativnog modela za unakrsnu validaciju, ali ta kombinacija je resursno puno složenija i nije provedena u sklopu ovog rada.

Na kraju su u tablici 6.4. prikazani rangovi svih značajki, ovisno o korištenoj metodi za odabir optimalnog podskupa značajki. Za dvije metode rangovi se značajno razlikuju. Značajke koje su proglašene bitnima u jednoj metodi, nisu značajne u drugoj. Može se izdvojiti nekoliko značajki koje niti u jednoj metodi nisu bile značajne: *system_ticket*, *months_since_player_registration*, *active_on_sport_lm4*, *active_on_other_verticals_lm4*.

značajka	rang (filtriranje)	rang (RFE)
ticket_bets_count	31	12
prematch_ticket	19	21
system_ticket	36	33
total_ticket_odds	29	1
ticket_payin	30	2
top_sport_payin_rate	9	3
sport_high_payin_rate	11	18
sport_medium_payin_rate	2	5
sport_low_payin_rate	7	32
football_high_payin_rate	12	28
football_medium_payin_rate	15	23
football_low_payin_rate	13	25
basketball_high_payin_rate	3	19
basketball_medium_payin_rate	5	31
basketball_low_payin_rate	1	34
tennis_pro_rate	10	22
tennis_other_rate	6	30
potential_payout	27	4
has_low_payin_group_bet	14	29
has_similar_ticket	35	15
days_since_player_registration	32	7
months_since_player_registration	33	20
player_age	34	13
sport_payin_lm	21	10
sport_win_lm	20	17
active_on_other_verticals_lm	4	24
avg_payin_per_sport_ticket_lm	23	1
avg_ticketbetcount_lm	26	1
tickets_lm	28	6
active_on_sport_lm	8	26
win_rate_lm	22	9
sport_payin_lm4	17	11
sport_win_lm4	18	14
tickets_lm4	25	8
active_on_sport_lm4	16	27
active_on_other_verticals_lm4	24	16

Tablica 6.4. Rangiranje značajki različitim metodama

7. Zaključak

Ne postoji jedinstveni pristup korištenju metoda dubinske analize podataka za rješavanje svih problema. To se jasno vidi prilikom korištenja različitih metoda za odabir najboljeg podskupa značajki. Filtriranje značajki temelji se na određivanju povezanosti svake pojedine značajke sa ciljnom i samim time ne uzima u obzir interakcije među prediktivnim značajkama. Metoda rekurzivne eliminacije ima drugačiji pristup i u svakoj iteraciji odbacuje najmanje važnu značajku. Tako dobiveni rezultati ne ovise samo o samim značajkama, nego i o modelu koji se koristi u iteracijama odbacivanja značajki.

Dobiveni rezultati pokazuju značajna odstupanja u procijenjenoj važnosti značajki za klasifikaciju rizičnih sportskih listića. Ne može se odrediti skup značajki koji značajno doprinosi ispravnim predviđanjima, ali uočene su značajke koje imaju nizak značaj u oba pristupa i nisu jako važne za rješavanje problema. Uzrok tomu može biti u primijenjenim metodama, ali i u nedovoljnoj informativnosti pojedinih značajki, odnosno kompleksnosti samog problema.

Drugi uzrok loših rezultata svih metoda može biti zbog pristranosti u podacima. Rizični listići (ciljna značajka) označeni su ručno, od strane više osoba, na temelju njihove osobne procjene rizičnosti. Ne postoji točna definicija što je rizik, nego se njegova procjena temelji na intuiciji i iskustvu pojedinih kladioničara. Ukoliko kladioničari nisu dosljedni ili nemaju iste kriterije za procjenu, zbog toga može doći do velikog šuma u podacima.

Uz pomoć stručnjaka iz domene sportskog klađenja bi se potencijalno mogle izgraditi kvalitetnije značajke. Sve izgrađene značajke imale su za cilj što bolje opisati ponašanje i karakteristike pojedinog igrača i događaja na koje se kladi. U dostupnom skupu podataka bilo je dosta nekonzistentnih igrača na platformi — onih koji nisu bili aktivni barem

jednim dijelom promatranog perioda. Bilo bi korisno segmentirati igrače u nekoliko skupina i na temelju toga opisati njihove karakteristike, ali za to nije bilo dovoljno podataka. Što se tiče procjene jačine sportova i natjecanja, numerička bi značajka možda bila bolja opcija, ali samo na temelju udjela u ukupnoj uplati teško je kvalitetno izraziti formulu kojom bi se izračunala.

Osim poboljšanja kvalitete podataka, uvijek postoji mogućnost dodatnog istraživanja drugih metoda dubinske analize podataka. Tako ovaj rad može poslužiti kladionicama (ili onima koji održavaju njihove sustave) kao dobar temelj za razvoj sustava za automatizirano odbijanje rizičnih sportskih listića.

Literatura

[1] Hrvatska Lutrija, "Opća pravila klađenja", izvor: https://www.lutrija.hr/static/assets/Korporativni/Dokumentacija_Web/Pravila_Web/Opca_pravila/2023/Opca_Pravila_kladenja_2023.pdf, 2023., preuzeto: 18.1.2024.

[2] PSK, "Što je Betbuilder?" izvor: https://help.psk.hr/article/1047, 2023., pristupljeno: 23.6.2024.2024.

[3] rizik, *Hrvatska enciklopedija, mrežno izdanje*. Leksikografski zavod Miroslav Krleža, 2013 – 2024., pristupljeno: 22.1.2024.

[4] C. Zins, "Conceptual approaches for defining data, information, and knowledge", *JASIST*, sv. 58, str. 479–493, 02 2007. https://doi.org/10.1002/asi.20508

[5] M. Frické, "The Knowledge Pyramid: the DIKW Hierarchy", *Knowledge Organization*, sv. 46, str. 33–46, 01 2019., dostupno i na: https://www.isko.org/cyclo/dikw, pristupljeno: 31.5.2024.

[6] I. H. Witten, E. Frank, M. A. Hall, i C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4. izd., ser. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers, 2016.

[7] IBM, "What is data mining?" izvor: https://www.ibm.com/topics/data-mining, pristupljeno: 25.5.2024.

[8] D. Pyle, *Data Preparation for Data Mining*, 1. izd., ser. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, 1999.

[9] A. Rotondo i F. Quilligan, "Evolution Paths for Knowledge Discovery and Data Mining Process Models", *SN Computer Science*, sv. 1, 2020.

[10] H. Hamilton, “Overview of the KDD Process”, https://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1_kdd.html, pristupljeno: 1.6.2024.

[11] N. Hotz, “What is CRISP DM?” izvor: https://www.datascience-pm.com/crisp-dm-2/, pristupljeno: 2.6.2024.

[12] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, i R. Wirth, *CRISP-DM 1.0: step-by-step data mining guide*, 2000.

[13] J. Brownlee, *Data Preparation for Machine Learning-Data Cleaning, Feature Selection, and Data Transforms in Python*, ser. Machine Learning Mastery. Independently Published, 2020.

[14] G. Press, “Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says”, izvor: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/, 2016., pristupljeno: 5.6.2024.

[15] scikit learn, “LocalOutlierFactor”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html, pristupljeno: 5.6.2024.

[16] ——, “IsolationForest”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html#isolationforest, pristupljeno: 5.6.2024.

[17] J. Brownlee, “Discover Feature Engineering, How to Engineer Features and How to Get Good at It”, izvor: https://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/, pristupljeno: 10.6.2024.

[18] scikit learn, “KBinsDiscretizer”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.KBinsDiscretizer.html#sklearn.preprocessing.KBinsDiscretizer, pristupljeno: 8.6.2024.

[19] Wikipedia, “Mutual Information”, izvor: https://en.wikipedia.org/wiki/Mutual_information, pristupljeno: 7.6.2024.

[20] J. Brownlee, “Information Gain and Mutual Information for Machine Learning”, izvor: https://machinelearningmastery.com/information-gain-and-mutual-information/, pristupljeno: 29.5.2024.

[21] scikit learn, “RFE”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html#sklearn.feature_selection.RFE, pristupljeno: 5.6.2024.

[22] ——, “RFECV”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFECV.html#sklearn.feature_selection.RFECV, pristupljeno: 5.6.2024.

[23] ——, “SequentialFeatureSelector”, izvor: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SequentialFeatureSelector.html#sklearn.feature_selection.SequentialFeatureSelector, pristupljeno: 5.6.2024.

[24] Wikipedia, “Latent space — Wikipedia, the free encyclopedia”, https://en.wikipedia.org/w/index.php?title=Latent_space&oldid=1193128291, 2024., pristupljeno: 10.6.2024.

[25] scikit learn, “Manifold Learning”, izvor: https://scikit-learn.org/stable/modules/manifold.html#multidimensional-scaling, pristupljeno: 10.6.2024.

[26] Y. Feng, M. Zhou, i X. Tong, “Imbalanced classification: a paradigm-based review”, 2021.

[27] J. Brownlee, *Imbalanced Classification with Python: Better Metrics, Balance Skewed Classes, Cost-Sensitive Learning*, 1. izd., ser. Machine Learning Mastery. Independently Published, 2020.

[28] J. Lever, M. Krzywinski, i N. Altman, “Classification evaluation”, *Nature Methods*, sv. 13, br. 8, str. 603–604, 8 2016. https://doi.org/10.1038/nmeth.3945

[29] K. W. Bowyer, N. V. Chawla, L. O. Hall, i W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique”, *CoRR*, sv. abs/1106.1813, 2011. [Mrežno]. Adresa: http://arxiv.org/abs/1106.1813

[30] H. Han, W.-Y. Wang, i B.-H. Mao, “Borderline-smote: A new over-sampling method in imbalanced data sets learning”, u *Advances in Intelligent Computing*, D.-S. Huang, X.-P. Zhang, i G.-B. Huang, Ur. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005., str. 878–887.

[31] H. M. Nguyen, E. W. Cooper, i K. Kamei, “Borderline over-sampling for imbalanced data classification”, u *Proceedings of the Fifth International Workshop on Computational Intelligence & Applications*, sv. 2009, br. 1. Higashi-Hiroshima City: IEEE SMC Hiroshima Chapter, studeni 2009., str. 24–29. [Mrežno]. Adresa: https://ousar.lib.okayama-u.ac.jp/en/19617

[32] H. He, Y. Bai, E. A. Garcia, i S. Li, “Adasyn: Adaptive synthetic sampling approach for imbalanced learning”, u *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008., str. 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969

[33] imbalanced learn, “Under-sampling”, izvor: https://imbalanced-learn.org/stable/under_sampling.html, pristupljeno: 12.6.2024.

Sažetak

Uporaba metoda dubinske analize podataka za potrebe upravljanja rizikom u sportskom klađenju

Ana Šutalo

Ovaj rad istražuje metode dubinske analize podataka i njihovu primjenu za potrebe upravljanja rizikom u sportskom klađenju. Upravljanje rizikom ključno je za uspjeh kladionica, a korištenje naprednih analitičkih metoda može pomoći u automatizaciji procesa prepoznavanja rizičnih sportskih listića. Analiza uključuje čišćenje podataka, transformaciju značajki, skaliranje i odabir značajki te evaluaciju modela. Poseban fokus stavljen je na problem nebalansiranosti podataka: korištenje odgovarajućih metrika i primjenu tehnika ponovnog uzorkovanja kako bi se poboljšala točnost klasifikatora. Rezultati istraživanja nisu dali dovoljno dobar klasifikator za primjenu u industriji koji bi pouzdano upravljao rizikom, ali može poslužiti kao baza za razvoj automatiziranog odbijanja rizičnih sportskih listića. Prepoznata je potreba za poboljšanjem kvalitete podataka i izgradnjom boljih značajki za daljnji napredak u ovom području.

Ključne riječi: sportsko klađenje; upravljanje rizikom; dubinska analiza podataka; nebalansiranost podataka; klasifikacija

Abstract

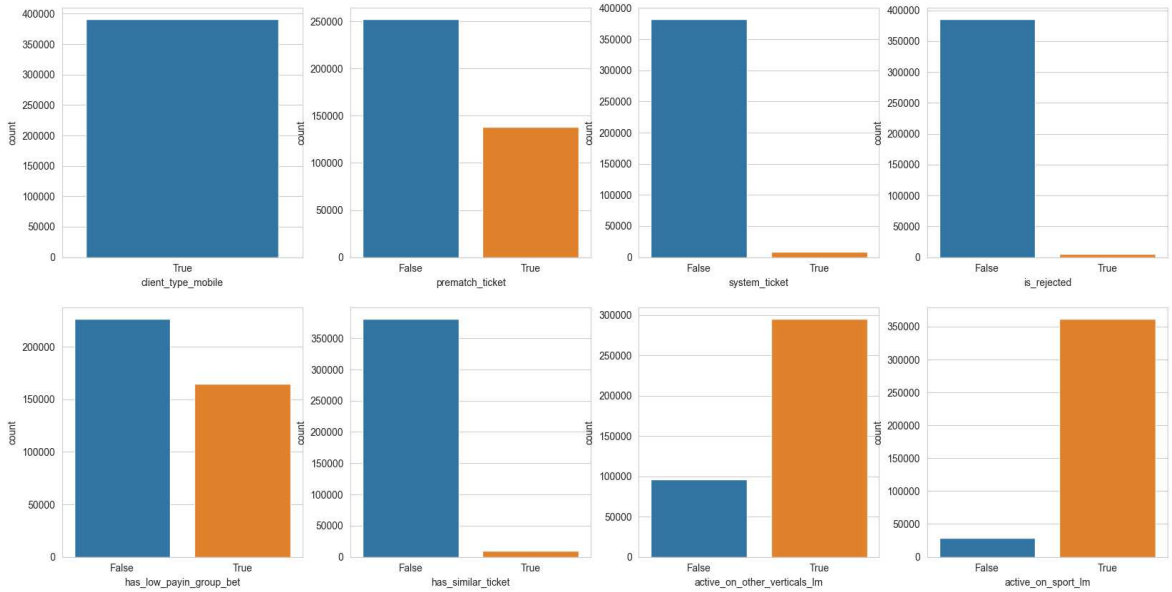
Using Data Mining Methods for Risk Management in Sports Betting

Ana Šutalo

This study investigates data mining techniques and their use in risk management for sports betting. Effective risk management is essential for the profitability of betting companies, and advanced analytical methods can help automate the identification of high-risk betting tickets. The study involves data cleaning, feature transformation, scaling and selection, and model evaluation. The issue of data imbalance is addressed using suitable metrics and resampling techniques to enhance classifier accuracy. Although the research did not yield a sufficiently reliable classifier for industry use, it provides a basis for developing automated systems to reject high-risk betting tickets. The importance of improving data quality and creating better features for future progress in this area has been highlighted.

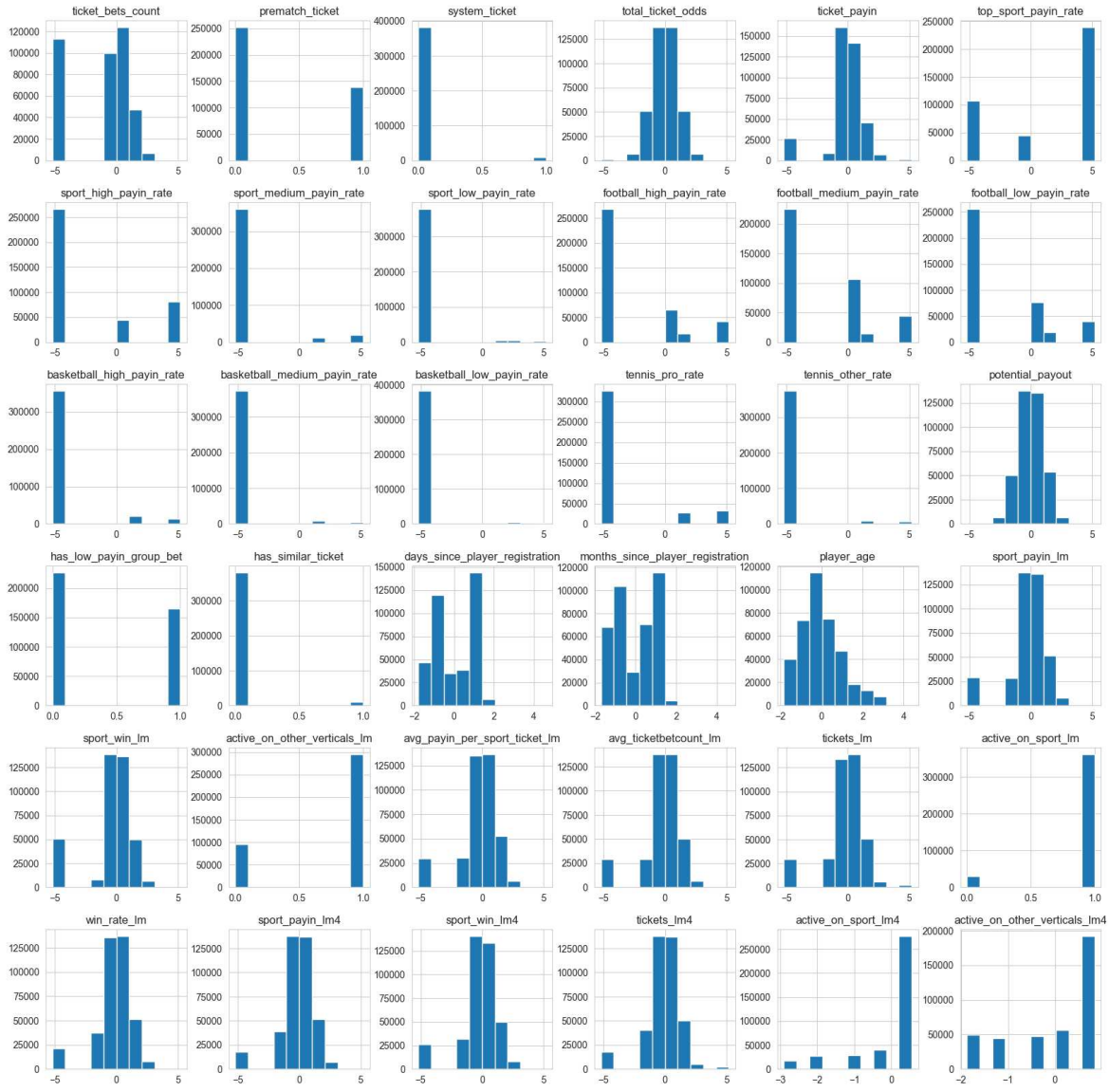
Keywords: sports betting; risk management; data mining; data imbalance; classification

Privitak A: Razdiobe značajki



Slika A1. Razdiobe binarnih značajki

Slika A2. Razdiobe numeričkih značajki



Slika A3. Numeričke značajke nakon primjene odgovarajućih transformacija