Univerzitet Union
Računarski fakultet

Metode mašinskog učenja za predviđanje ishoda sportskih događaja

Uglješa Stojanović

mentor: prof. dr Dragan Urošević

Sadržaj

Uvod

Engleska Premijer liga (EPL) smatra se za jedno od najpopularnijih fudbalskih takmičenja na svetu. Po podacima iz [1], tokom sezone 2013-14 procenjeno je da je utakmice pratilo preko 1.1 milijardi fanova širom sveta. Televizijski prenosi našli su se na malim ekranima 645 miliona domova u 175 država i 212 teritorija, a prava na njih su procenjena na preko milijardu britanskih funti po sezoni. U EPL za trofej se takmiči 20 timova, od kojih tri najgore plasirana svake sezone ispadaju iz takmičenja i bivaju zamenjena najboljim timovima iz lige nižeg ranga. Svaki tim igra protiv svih ostalih dva puta, jednom na domaćem terenu i jednom kao gost. Dakle, tokom sezone odigra se ukupno $2\binom{20}{2} = 380$ susreta. Sezona traje od avgusta do maja naredne godine.

Predmet ovog rada je primena metoda mašinskog učenja (eng. machine learning) u predviđanju krajnjeg ishoda utakmica EPL. Mogući ishodi podrazumevaju pobedu domaćeg tima, pobedu gostujućeg tima, kao i nerešen rezultat. Kako je navedeni skup diskretan, u pitanju je problem koji podrazumeva dodeljivanje klase (kategorije) odre enom skupu podataka. Ovakvi problemi se u teoriji mašinskog učenja nazivaju problemi klasi kacije. Veliki broj susreta završenih nerešenim ishodom predstavlja jedan od ključnih izazova u ovom radu, budući da oni drastično povećavaju neodre enost modela.

Problem klasi kacije predstavlja jedan od ključnih problema u oblasti istraživanja podataka i mašinskog učenja. Metode za klasi kaciju nalaze široke primene kao glavni ili pomoćni mehanizmi u sistemima za podršku odlučivanju, obradi signala, medicinskoj dijagnostici, obradi multimedijalnih sadržaja itd. S obzirom na praktični, ali i teorijski značaj klasi kacije, razvijen je veliki broj metoda (klasi katora) koje se bave ovim problemom. Neke od često primenjivanih metoda klasi kacije su: metoda podržavajućih vektora, metoda najbližih suseda, klasi kacija korišćenjem stabla odlučivanja, veštačke neuronske mreže i dr.

Klasi katori predstavljaju nadgledanu tehniku učenja, što znači da se u fazi učenja klasi kator snabdeva ulaznim vrednostima i očekivanim izlaznim vrednostima, odnosno očekivanim klasama. Tokom procesa učenja klasi katora, nailazi se na različite probleme koji su vezani za kvalitativne i/ili kvantitativne karakteristike ulaznih i izlaznih podataka, ili stanje parametara klasi katora. Jedan od problema vezanih za kvalitativne i kvantitativne karakteristike ulaznih podataka je tzv. problem odabira atributa. Neka je dat skup od N atributa. Budući da svaki atribut može da bude uključen ili isključen iz skupa razmatranih atributa, postoji $2^N - 1$ različitih načina da se odabere neprazan podskup skupa svih atributa, odnosno podskup atributa koji će učestvovati u procesu klasi kacije. Odabir adekvatnih atributa ima ključni uticaj, ne samo na kvalitet, već i na e kasnost klasi kacije, jer dimenzija upotrebljenog podskupa atributa utiče na dužinu vremena izvršavanja i količinu upotrebljenog memorijskog prostora. Srodan problem, ali na realnom domenu, predstavlja problem odre ivanja težina atributa, gde se težina interpretira kao značaj atributa. Za razliku od problema odabira atributa, kod ovog problema atribut ne mora da bude samo uključen ili isključen, već može da bude uključen sa nekim stepenom značajnosti. U nekim slučajevima se dešava da, i pored adekvatnog odabira atributa ili njihovih težina, kvalitet klasi kacije nije na zadovoljavajućem nivou. Uzrok ovog problema može biti loš odabir parametara metode

za klasi kaciju. S obzirom na to da se parametri obično pretražuju na domenu realnih vrednosti, tradicionalne tehnike za rešavanje problema podešavanja parametara, poput pretrage mreže (eng. grid search), ne uspevaju da proizvedu zadovoljavajuće rezultate kada je broj ovih parametara veliki.

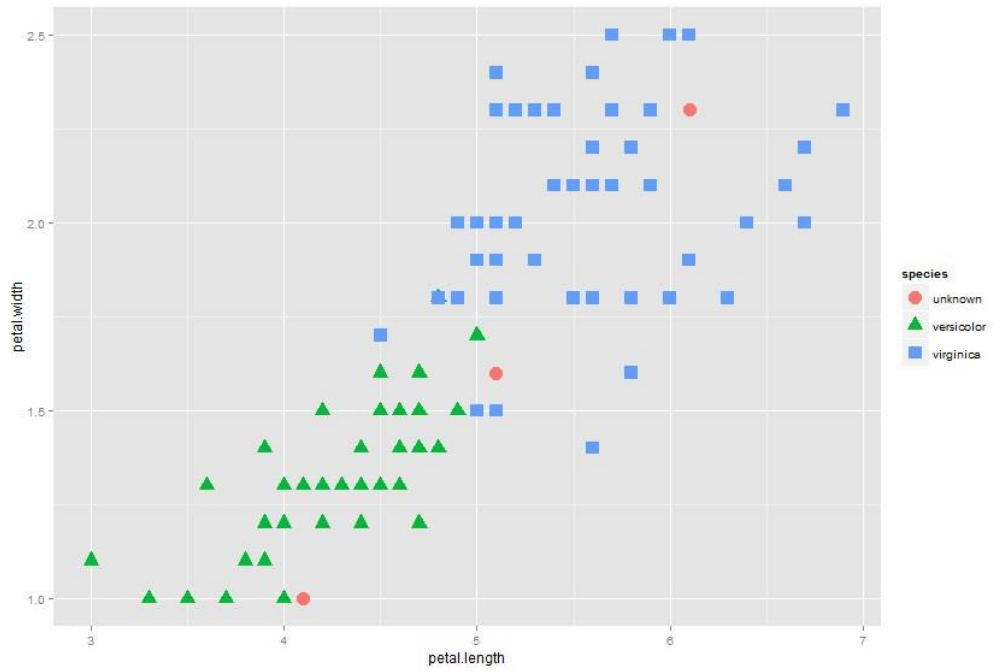
Rad se sastoji iz pet poglavlja, a većina poglavlja iz većeg broja sekcija. U drugom poglavlju se izlažu osnovni pojmovi i koncepti koji će se koristiti u daljem izlaganju: problem klasi kacije sa svim relevantnim aspektima, uključujući i probleme podešavanja parametara klasi katora, odabira atributa, podešavanja težina atributa, kao i analizu kvaliteta klasi kacije. U trećem poglavlju dat je pregled popularnih metoda klasi kacije koje se koriste u nauci i industriji. Potom, u četvrtom poglavlju predloženi su načini primene poznatih klasi katora na predviđanje ishoda mečeva EPL uz ocenu kvaliteta svakog od pristupa. Poslednje, peto poglavlje, sadrži zaključak, u kome je rekapituliran predmet rada i primenjenih metodologija uz zaključke koji slede iz rezultata dobijenih u četvrtom poglavlju.

Klasi kacija - osnovni koncepti

Klasi kacija se bavi problemom dodeljivanja klase (kategorije) nekom objektu, pri čemu je broj mogućih klasa konačan i unapred poznat. Sledeći primer ilustruje problem klasi kacije u kojoj su moguće dve klase.

Primer 1. Primer je zasnovan na skupu podataka pod nazivom Iris. Reč je o struktuiranom skupu podataka o biljci (cvetu) Iris koji se često koristi kao test problem za potrebe klasi kacije u literaturi. Iris skup podataka se može preuzeti sa Repozitorijuma za mašinsko učenje UCI [2]. Podaci su podeljeni u tri kategorije koje predstavljaju tip Iris cveta: iris setosa, iris versicolor i iris virginica. Za svaku od kategorija postoji po 50 podataka, a svaki podatak ima sledeće informacije: dužina čašice (eng. sepal length), širina čašice (eng. sepal width), dužina latice (eng. petal length) i širina latice (eng. petal width). Na Slici 2.1 je prikazan podskup skupa ovih podataka. Horizontalna osa odgovara dužini latice, a vertikalna njenoj širini. Zbog preglednosti vizuelne ilustracije, preostale dve informacije, o dužini i širini čašice, su izostavljene. Dodatno pojednostavljenje je napravljeno i po pitanju broja klasa time što su izostavljeni podaci za tip iris setosa.

Trouglovima su predstavljeni podaci koji odgovaraju tipu iris versicolor, dok su podaci tipa iris virginica označeni kvadratima. Može se primetiti da postoji odre ena geometrijska pravilnost po pitanju grupisanja dva različita tipa, naime, iris versicolor je pretežno raspore en u donjem levom uglu gde su niže vrednosti oba posmatrana svojstva, dok je drugi tip Irisa pretežno raspore en u gornjem desnom uglu. Ova pravilnost upućuje na zaključak da se može postaviti intuitivna granica izme u dve klase cvetova. Neformalno gledano, odre ivanje pravilnosti, po kojoj se podaci mogu razvrstati u klase, upravo predstavlja klasi kaciju. Ukoliko bi to pravilo bilo predstavljeno pravom linijom, ono ne bi bilo u mogućnosti savršeno da razgraniči sve podatke, jer bi se neki podaci nalazili sa pogrešne strane prave. Sa druge strane, upotrebom npr. polinomske funkcije dovoljno velikog stepena, bilo bi moguće



Slika 2.1: Problem binarne klasi kacije - Iris

razgraničiti podatke u potpunosti. Me utim, postavlja se pitanje: da li bi takva funkcija uspešno klasi kovala nove podatke, koji se ne nalaze u poznatom skupu? Pored podataka koji odgovaraju poznatim klasama cvetova, na slici su prikazani i krugovi koji se odnose na podatke čija klasa nije poznata, odnosno na nove podatke koji imaju sve informacije osim klase. Intuitivno bi bilo klasi kovati krug bliži donjem levom uglu kao trougao, a drugi bliži gornjem desnom uglu kao kvadrat. Postavlja se dilema po pitanju klase kruga koji se nalazi bliže margini izme u dve oblasti. Već i kod ovakvog jednostavnog primera se mogu uočiti odre eni izazovi, što sugeriše da je rešavanje klasi kacionih problema vrlo težak problem, posebno kada je broj podataka, njihovih svojstava i klasa veliki.

U prethodnom primeru su se pominjali pojmovi poput: tip cveta, dužina latice cveta, pravilo razdvajanja i dr. Uopštenja svih pominjanih relevantnih pojmova su data na sledeći način. Skup podataka sa poznatim tipovima cvetova koji se koriste u procesu klasi kacije predstavlja trening podatke. Trening podaci ponekad predstavljaju samo podskup skupa svih poznatih podataka. Motivacija za upotrebu samo podskupa skupa svih podataka će biti opisana u narednim sekcijama. Dužina i širina latice predstavljaju atribut podatka, dok se tipovi cvetova iris versicolor i iris irginica nazivaju klasama podatka. Zajednički, skup atributa i klasa formira jedan podatak. Tako e je bitno razgraničiti pojam podatka, koji uključuje sve informacije (atributi i klasa), od podatka za koji nije poznata klasa. Iz tog razloga će se drugi pojam nazivati vektor atributa u daljem tekstu. Pravilo koje razgraničava podatke jedne i druge klase, o kojem je bilo reči (linearna funkcija, polinomijalna itd.), se naziva klasi kaciona funkcija, a funkcija dodeljivanja klase se tako e naziva i funkcija odlučivanja. Funkcija odlučivanja je, obično, vrlo usko povezana sa klasi kacionom funkcijom. U Primeru 1 je ta

veza zasnovana na proveri sa koje strane prostora podeljenog pravom ili polinomskom funkcijom se nepoznati podatak nalazi. Pored ovih termina, u daljem izlaganju biće po potrebi uvedeni još neki termini vezani za klasi kaciju.

U literaturi je problem klasi kacije i metoda za njegovo rešavanje detaljno izučavan, a samo neki od detaljnih i sveobuhvatnih resursa su [3], [4] i [5].

Sledi opšta de nicija problema tzv. binarne klasi kacije, odnosno klasikacije u kojoj postoje samo dve moguće klase. U daljem tekstu su date i de nicije problema klasi kacije prilago ene konkretnim klasi katorima.

De nicija 1. Neka je dat skup trening podataka D_{tr} koji se sastoji od N_{tr} ureenih parova oblika $(\mathbf{x}^{(i)},y^{(i)}),i = 1,\ldots,N_{tr}$, gde je $\mathbf{x}^{(i)} \in \mathbf{R}^N$ N-dimenzionalni vektor atributa, a $y^{(i)} \in \{-1,1\}$ odgovarajuća klasa. U literaturi se vektori sa klasom 1 nazivaju i pozitivni primeri, tj. vektori sa pozitivnom klasom, a vektori klase -1 se nazivaju negativni primeri. Klasi kacija podrazumeva formiranje klasi kacione funkcije $f : \mathbf{R}^N \rightarrow \mathbf{R}$. Na osnovu klasi kacione funkcije se odre uje funkcija odlučivanja $c : \mathbf{R}^N \rightarrow \{-1,1\}$ koja je u stanju da za novi vektor atributa $\mathbf{x} \in \mathbf{R}^N$, koji nije pripadao trening skupu, odredi klasu $c(\mathbf{x}) = \hat{y}$, tako da ona bude jednaka pravoj klasi datog vektora atributa $y = \hat{y}$.

Prema [3], izdvajaju se tri interpretacije problema klasi kacije, a samim tim i tri različita pristupa u njegovom rešavanju: 1) statistički pristup; 2) pristup zasnovan na mašinskom učenju; 3) pristup zasnovan na veštačkim neuronskim mrežama.

Klasičan statistički pristup je zasnovan na Bajesovom pravilu odlučivanja, a nešto moderniji pristupi koriste i dodatna poboljšanja, kao npr. mešovite raspodele atributa. Zajedničko za sve statističke pristupe je da se klasi kacija ne vrši direktno, već implicitno, odre ivanjem verovatnoća da posmatrani podatak pripada svakoj od mogućih klasa.

U mašinskom učenju se problem klasi kacije svodi na odre ivanje automatizovanih procedura koje su u stanju da nauče da klasi kuju konzumacijom dovoljnog broja trening podataka. Ovaj pristup je obično u potpunosti vo en podacima i automatizovan, te ne zahteva nikakve dodatne intervencije čoveka. Problem je što količina podataka potrebnih za učenje klasi katora može biti velika. Najveći broj metoda u ovom pristupu je zasnovan na stablima odlučivanja.

Veštačke neuronske mreže predstavljaju kombinaciju prethodna dva pristupa i zasnovane su na strukturalnoj i funkcionalnoj imitaciji ljudskog mozga. S obzirom da spada u grupu univerzalnih aproksimacionih metoda, problem klasi kacije, iz perspektive veštačke neuronske mreže, je jednostavno postavljen kao problem učenja klasi kacione funkcije. Jedan od problema ovog pristupa je potpuno odsustvo transparentnosti klasi kacionog modela prema korisniku, što ne važi za prethodna dva pristupa.

Sekcija koja sledi prikazuje neke koncepte Bajesove teorije odlučivanja koji motivišu upotrebu klasi kacionih metoda u slučaju dve klase (u [5] se može naći detaljan pregled Bajesove teorije odlučivanja). Proširenja na slučaj više klasa se neće razmatrati u statističkom smislu već samo iz opšte perspektive. U pitanju su različiti načini svo enja problema višeklasne klasikacije na višestruke probleme binarne klasi kacije i o tome će biti reči nakon Sekcije 2.1.

2.1 Bajesova teorija odlučivanja

Bajesova teorija odlučivanja predstavlja osnovni statistički aparat u rešavanju problema klasi kacije. Klasi kacija, tačnije donošenje odluke o pripadnosti klasi se sprovodi ocenjivanjem vrednosti verovatnoća za svaku od klasa. Neka se razmatra problem binarne klasi kacije (De nicija 1) vektora atributa označenog sa $\mathbf{x} = (x_1,\ldots,x_n)$, i neka je ω_1 događaj kada je pripadajuća klasa -1, a ω_2 događaj kada je klasa 1. Neka su $P(\omega_1)$ i $P(\omega_2)$ redom verovatnoće koje odgovaraju realizacijama prvog i drugog događaja. Kada ne postoje drugi mogući događaji, onda važi da je $P(\omega_1) + P(\omega_2) = 1$. Ako ne postoje nikakve dodatne informacije, tj. ako su poznate samo bezuslovne verovatnoće $P(\omega_1)$ i $P(\omega_2)$, logičan izbor za funkciju odlučivanja je

$$c(\mathbf{x}) = \begin{cases} -1, & P(\omega_1) \geq P(\omega_2) \\ 1, & P(\omega_1) < P(\omega_2) \end{cases}. \qquad (2.1)$$

Gore prikazani klasi kator je smislen u slučaju da ne postoje nikakve informacije o vektoru atributa koji se klasi kuje. Sa druge strane, može se primetiti da će atributi zavisiti od njegove klase, tj. da ukoliko znamo klasu nekog podatka, možemo iz toga zaključiti nešto o njegovim atributima. Zarad pojednostavljenja notacije, umesto razmatranja slučajeva ω_1 i ω_2 koristiće se samo oznaka za događaj ω_j. Vektor atributa se može posmatrati kao vektor slučajnih promenljivih, a $p(\mathbf{x}|\omega_j)$ kao gustina uslovne verovatnoće. S obzirom na to da važi $p(\omega_j,\mathbf{x}) = p(\mathbf{x}|\omega_j)P(\omega_j) = P(\omega_j|\mathbf{x})p(\mathbf{x})$, dolazimo do poznatog rezultata, tzv. Bajesove formule

$$P(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)P(\omega_j)}{p(\mathbf{x})}. \qquad (2.2)$$

Budući da postoje dva moguća događaja, važi da je $p(\mathbf{x}) = p(\mathbf{x}|\omega_1)P(\omega_1)+ p(\mathbf{x}|\omega_2)P(\omega_2)$, pa se uvrštavanjem u (2.2) dobija

$$P(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)P(\omega_j)}{p(\mathbf{x}|\omega_1)P(\omega_1) + p(\mathbf{x}|\omega_2)P(\omega_2)}. \qquad (2.3)$$

U skladu sa (2.3), prethodna funkcija odlučivanja (2.1) se može unaprediti tako da uzima u obzir i informacije iz vektora atributa. Sledeća relacija se zove Bajesovo pravilo odlučivanja

$$c(\mathbf{x}) = \begin{cases} -1, & P(\omega_1|\mathbf{x}) \geq P(\omega_2|\mathbf{x}) \\ 1, & P(\omega_1|\mathbf{x}) < P(\omega_2|\mathbf{x}) \end{cases}. \qquad (2.4)$$

Može se pokazati da Bajesovo pravilo odlučivanja minimizuje prosečnu grešku funkcije odlučivanja. Naime, u skladu sa datim pravilom odlučivanja, verovatnoća greške je data kao:

$$P(\text{greška}|\mathbf{x}) = \begin{cases} P(\omega_1|\mathbf{x}), & \text{ako je odabrana klasa 1} \\ P(\omega_2|\mathbf{x}), & \text{ako je odabrana klasa -1} \end{cases}. \qquad (2.5)$$

Za svako dato $\mathbf{x}$ se može minimizovati greška tako što se primeni Bajesovo pravilo odlučivanja. Prosečna greška je predstavljena izrazom

$$Z$$

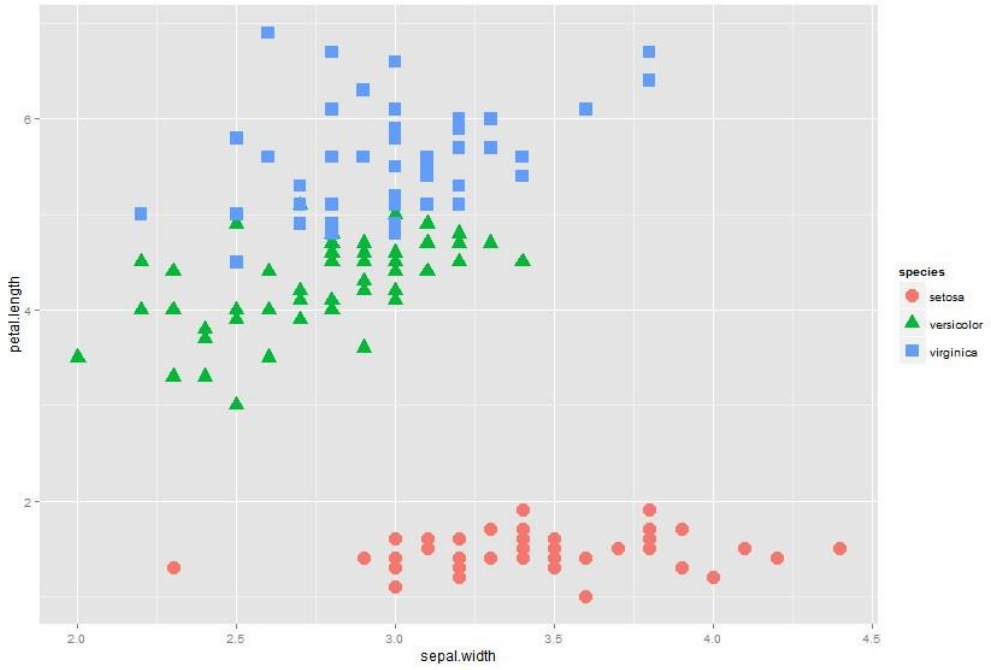
$$P(\text{greška}) = \int_{D(c)} P(\text{greška},\mathbf{x})\,d\mathbf{x}, \qquad (2.6)$$

gde je $D(c)$ oblast de nisanosti funkcije odlučivanja c.

Iz relacije (2.6) se može zaključiti da, ako je za svako pojedinačno $\mathbf{x}$ greška najmanja moguća, onda će i dati integral biti najmanji moguć. Budući da se Bajesovim pravilom odlučivanja minimizuje svaka pojedinačna greška, zaključuje se da će i integral biti najmanji moguć, a samim tim i prosečna greška odlučivanja (klasi kacije).

2.2 Višeklasni klasi kacioni problemi

U Primeru 1 je pomenuto da je originalni skup podataka cveta Iris sačinjen od podataka o cvetovima tri tipa. Na Slici 2.2 je prikazan potpun skup podataka sa dodatim cvetovima tipa iris setosa. Zarad ilustrativnijeg rasporeda podataka, na horizontalnoj osi je sada vrednost dužine čašice, a na vertikalnoj širina latice. Evidentno je da se ne može povući linearna granica koja bi razdvajala sva tri regiona. Me utim, granica je ipak intuitivna, i može se opisati nekim složenijim pravilom razdvajanja. Pre nego što se nastavi sa prikazom nekih poznatih metoda za rešavanje ovog problema, biće uvedena formalna de nicija problema višeklasne klasi kacije.



Slika 2.2: Problem višeklasne klasi kacije - Iris

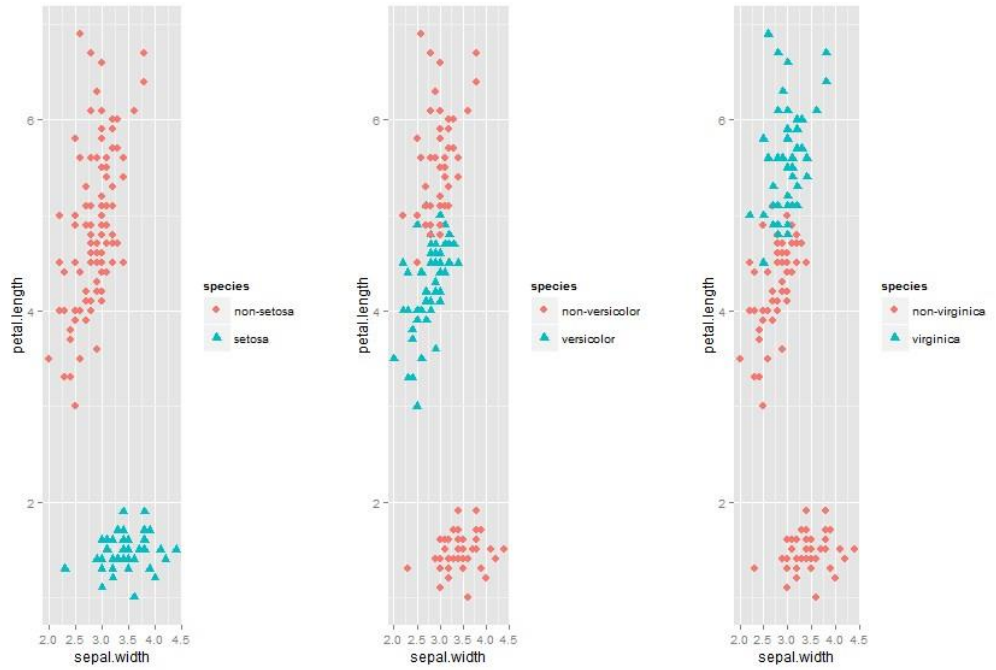
De nicija 2. Neka je dat skup trening podataka D_{tr} koji se sastoji od N_{tr} ureenih parova oblika $(\mathbf{x}^{(i)},y^{(i)}), i = 1,\ldots,N_{tr}$, gde je $\mathbf{x}^{(i)} \in \mathbf{R}^N$ N-dimenzionalni vektor atributa, a $y^{(i)} \in \{1,\ldots,N_c\}$ odgovarajuća klasa. N_c je ukupan broj klasa, a klase su, bez gubitka opštosti, označene prirodnim brojevima od 1 do N_c. Klasi kacija podrazumeva formiranje klasi kacione funkcije f

: $\mathbf{R}^N \rightarrow \mathbf{R}$. Na osnovu klasi kacione funkcije se odre uje funkcija odlučivanja $c : \mathbf{R}^N \rightarrow \{1,\ldots,N_c\}$ koja je u stanju da za novi vektor atributa $\mathbf{x} \in \mathbf{R}^N$, koji nije pripadao trening skupu, odredi klasu $c(\mathbf{x}) = \hat{y}$, tako da ona bude jednaka pravoj klasi datog vektora atributa $y = \hat{y}$.

Ako se prethodna de nicija uporedi sa De nicijom 1, može se primetiti da je jedina razlika u kardinalnosti skupa klasa, koja sada može biti i veća od 2. De nicija višeklasne klasi kacije je, dakle, uopštenje binarne klasi kacije. Postoji vrlo srodan problem koji se zove višeznačna klasi kacija (eng. multilabel classi cation), gde je cilj dodeliti vektorima atributa jednu ili više klasa. Kod višeklasne klasi kacije (eng. mutliclass classi cation) svaki objekat pripada tačno jednoj klasi.

Ovde su prikazane dve osnovne tehnike rešavanja problema višeklasne klasi kacije. Obe tehnike su zasnovane na svo enju problema višeklasne klasi kacije na veći broj problema binarne klasi kacije. U literaturi su ova dva pristupa najčešće korišćena. Postoje i drugi predloženi pristupi, ali oni nisu razmotreni u ovom radu.

2.2.1 Jedan-protiv-Svih (eng. One-vs-All)



Slika 2.3: Jedan-protiv-Svih pristup

Ovaj pristup zasnovan je na ideji da se za N_c klasa, pravi N_c nezavisnih binarnih klasi katora, gde se i-ti klasi kator koristi za razdvajanje i-te klase od svih ostalih klasa, tj. i-ti klasi kator klasi kuje vektore klase i kao pozitivne primere, a sve ostale kao negativne. Ako se sa f_i označi klasi kaciona funkcija i-tog klasi katora, a sa $\mathbf{x}$ vektor atributa koji je potrebno klasi kovati, onda se funkcija odlučivanja za problem višeklasne klasi kacije dobija kao

$$c(\mathbf{x}) = \operatorname{argmax} f_i(\mathbf{x}),\ i = 1,\ldots,N_c. \qquad (2.7)$$

i

U izrazu (2.7), $\text{argmax}_i f_i(\mathbf{x})$ znači da se vrši maksimizacija izraza $f_i(\mathbf{x})$, ali da se potom kao vrednost vraća i za koje je maksimum dostignut.

Slika 2.3 ilustruje ovaj pristup. Za primer sa 3 tipa Iris cvetova se formiraju 3 klasi kacione funkcije. Na slici te funkcije nisu prikazane, ali je jasno da postoji intuitivna granica u svakom od slučajeva. Novom vektoru atributa $\mathbf{x}$ se dodeljuje klasa onog klasi katora koji postigne najveću moć diskriminacije prema ostalim klasama. Na Slici 2.3 se tako e vidi da u slučaju kada se podaci podele na grupe versicolor i non-versicolor, granica izme u dve novoformirane klase podataka nije linearna. U takvoj situaciji bi se mogla primeniti nelinearna klasi kaciona funkcija, ili transformacija skupa ulaznih podataka koja bi omogućila da podaci mogu biti razdvojeni linearnom funkcijom. O ovim, i o drugim problemima će biti reči kada se budu razmatrali konkretni klasi katori.

2.2.2 Svi-protiv-Svih (eng. All-vs-All)

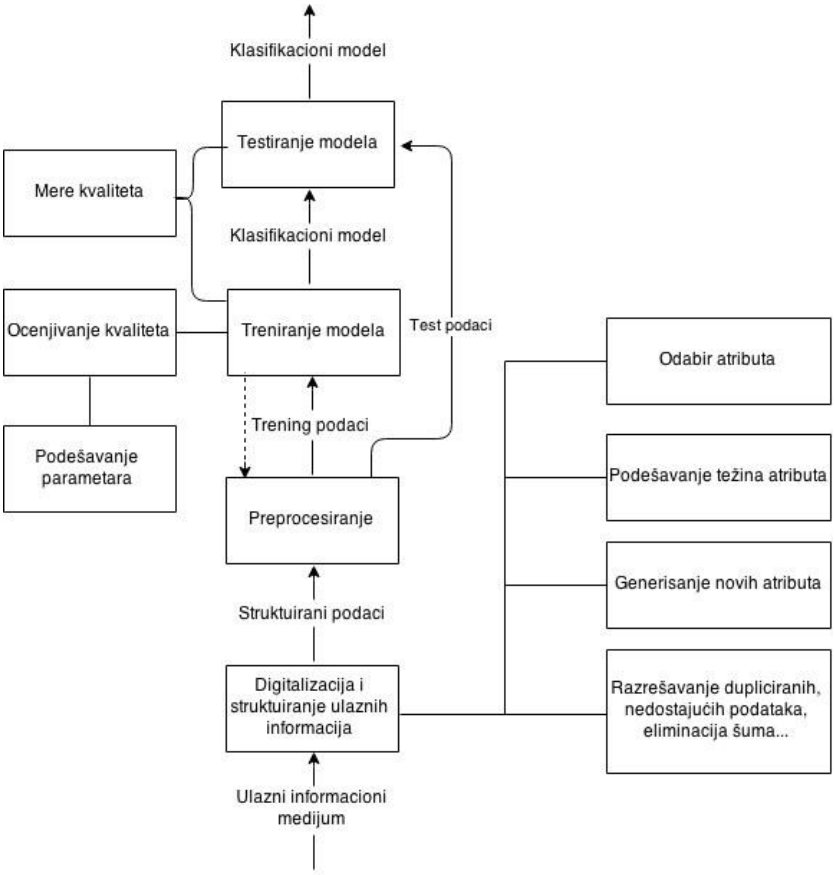
Ovaj pristup podrazumeva formiranje $N_c(N_c - 1)$ klasi katora, za svaki par klasa po jedan. Neka je sa f_{ij} označena klasi kaciona funkcija koja sve podatke klase i klasi kuje kao pozitivne primere, a sve podatke klase j kao negativne. Primeri ostalih klasa se ne razmatraju u formiranju klasi kacione funkcije f_{ij} . Primećuje se da je $f_{ij} = -f_{ji}$. Funkcija odlučivanja se računa putem izraza

$$c(\mathbf{x}) = \text{argmax}_i \left(\sum_{i \neq j} f_{ij}(\mathbf{x}) \right), \; i = 1, ..., N_c, \; j = 1, ..., N_c \qquad (2.8)$$

U literaturi se često koriste oba prikazana pristupa, i mnogobrojna empirijska istraživanja su pokazala da se ova dva metoda, iako jednostavna, dobro pokazuju u praksi. Vremenska e kasnost zavisi od veličine trening skupa podatka, broja klasa i klasi katora koji se koristi. Prvi pristup koristi samo N_c klasi katora, ali je trening skup sačinjen od celog početnog skupa, jer se razmatraju sve klase. U drugom pristupu, Svi-protiv-Svih, formira se $N_c(N_c-1)$ klasi katora. Me utim, složenost procesa učenja svakog od njih je manja, jer je i broj podataka po jednom klasi katoru manji. Ovo je posebno primetno u slučajevima kada je broj klasa velik, a njihova raspodela ravnomerna.

2.3 Ostali aspekti klasi kacije

U ovoj sekciji su opisani još neki prateći elementi klasi kacionog problema. Prvo od pitanja se tiče problema utvr ivanja kvaliteta klasi kacije. Zatim su izložene tehnike za izbegavanje prekomernog prilago avanja (eng. over tting) klasi katora, odnosno povećanja moći uopštavanja klasi kacionog modela. Ovde će biti reči i o problemu podešavanja parametara klasi katora. Na kraju su izložene procedure preprocesiranja podataka koje mogu doprineti poboljšanju e kasnosti i kvaliteta klasi kacije. Posebno su od interesa problem odabira atributa i podešavanja njihovih težina.



Slika 2.4: Proširena šema klasi kacije

Na Slici 2.4 je prikazana proširena šema klasi kacionog procesa. Na početku procesa je prikupljanje podataka i njihovo skladištenje u struktuiranom formatu. Na primer, ako se vrši detekcija nepoželjnih poruka, potrebno je prvo obraditi poruke elektronske pošte i dodatne informacije o njima. Ako se vrši klasi kacija tumora na osnovu slika dobijenih putem ultrazvuka ili magnetne rezonance, potrebno je obraditi slike i iz njih izvući relevantne informacije. Nakon što su podaci pripremljeni, prelazi se u fazu preprocesiranja. Preprocesiranje podrazumeva odabir atributa, podešavanje težina atributa, rešavanje problema nedostajućih vrednosti, šuma itd. Posle preprocesiranja, ulazi se u fazu formiranja klasi kacionog modela. Razmatra se i slučaj kada postoji i povratna veza sa fazom preprocesiranja, jer se odabir atributa i podešavanje njihovih težina izvršava u fazi treniranja modela. Bitan aspekt faze treninga je način ocenjivanja kvaliteta klasi kacije na trening podacima. Postoje različiti pristupi ocenjivanja kvaliteta koji omogućavaju visok stepen uopštavanja klasi katora i samim tim mogućnost njegove primene na novim podacima van trening skupa. Neki klasi kacioni modeli imaju svoju internu parametarsku strukturu, koja, ako se adekvatno podesi, može poboljšati ocenu kvaliteta u fazi treninga. Ovo, kasnije, ukoliko je ocena kvaliteta nepristrasna i ima dovoljnu moć generalizacije, dovodi do poboljšanja kvaliteta klasi kacije i na novim podacima. Mera kvaliteta je funkcija koja se koristi za opisivanje kvaliteta klasi katora, kao i za pore enje sa drugim metodama. Ona se koristi i u fazi treninga, kao i u fazi testiranja klasi kacionog modela. U fazi treninga se

ocenjuje buduća vrednost mere kvaliteta na novim podacima, a u fazi testiranja se ta buduća vrednost mere kvaliteta proverava.

2.3.1 Mere kvaliteta

Mera kvaliteta klasi kacionog modela omogućuje njegovu evaluaciju i pore enje sa drugim modelima. Mera kvaliteta klasi kacije obično predstavlja potencijal modela da korektno predvidi klasu novog podatka i/ili trening podataka. Matrica konfuzije (eng. confusion matrix) predstavlja pregledan i detaljan način da se taj potencijal prikaže. Ona prikazuje uporedni odnos izme u broja predvi enih i pravih klasa za neki skup vektora atributa. Pretpostavimo da je neki klasi kator proizveo klasi kacije predstavljene matricom konfuzije u Tabeli 2.1.

		Predvi ena setosa	versicolor	virginica
Prava	setosa	43	5	2
	versicolor	5	34	11
	virginica	1	1	48

Tabela 2.1: Matrica konfuzije

Redovi matrice odgovaraju pravim vrednostima klasa vektora atributa, dok kolone predstavljaju klase dodeljene od strane modela. Za red označen nazivom versicolor i kolonu označenu sa virginica, matrica prikazuje broj vektora atributa koje je model klasi kovao kao tip virginica, a ustvari je bila reč o cvetovima tipa versicolor. Elementi matrice koji se nalaze na glavnoj dijagonali prikazuju korektna predviđanja klasi kacionog modela. U opštem slučaju matrica konfuzije ima strukturu datu u Tabeli 2.2

		Predvi ena			
		1	2	$\cdots$	N_c
Prava	1	n_{11}	n_{12}	$\cdots$	n_{1N_c}
	2	n_{21}	n_{22}	$\cdots$	n_{2N_c}
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	N_c	n_{N_c1}	n_{N_c2}	$\cdots$	$n_{N_cN_c}$

Tabela 2.2: Matrica konfuzije - opšta struktura

Tačnost (eng. accuracy) je mera koja predstavlja odnos ukupnog broja korektnih predviđanja i ukupnog broja predviđanja. U skladu sa prethodno uvedenom notacijom, može se izračunati kao

$$Acc = \frac{\sum_{i=1}^{N_c} n_{ii}}{\sum_{i=1}^{N_c} \sum_{j=1}^{N_c} n_{ij}}. \tag{2.9}$$

Preciznost (eng. precision) je slična meri tačnosti, ali se odnosi samo na jednu posmatranu klasu. Ona predstavlja odnos tačnih pozitivnih predviđanja (eng. true positives) i ukupnog broja slučajeva u kojima je klasi kator predvideo posmatranu klasu. Računa se prema formuli

$$P_i = \frac{n_{ii}}{\sum_{j=1}^{N_c} n_{ji}},\ i = 1, ..., N_c \quad . \tag{2.10}$$

Odziv (eng. recall) prikazuje odnos korektno predvi enih vektora atributa neke klase i ukupnog broja pravih pojavljivanja te klase u skupu podataka

$$R_i = \frac{n_{ii}}{\sum_{j=1}^{N_c} n_{ij}},\ i = 1, ..., N_c \quad . \tag{2.11}$$

F-mera (eng. F-measure) je kombinovana mera dobijena kao harmonijska sredina preciznosti i odziva

$$F_i = \frac{2P_i R_i}{P_i + R_i},\ i = 1, ..., N_c \quad . \tag{2.12}$$

U Tabeli 2.3 su prikazane vrednosti ove četiri mere za matricu konfuzije datu u Tabeli 2.1.

	Mera kvaliteta		
Klasa	P_i	R_i	F_i
i = 1	0.88	0.86	0.87
i = 2	0.85	0.68	0.76
i = 3	0.79	0.96	0.77
Acc		0.83	

Tabela 2.3: Vrednosti klasi kacionih mera za matricu konfuzije iz Tabele 2.1

2.3.2 Ocene kvaliteta

Izgradnja klasi katora podrazumeva dve ključne faze: treniranje klasikacionog modela i njegovo testiranje. Treniranje podrazumeva formiranje klasi kacione funkcije (klasi katora) na osnovu jednog dela podataka. U fazi testiranja, kroz klasi kator se propuštaju test podaci, odnosno podaci koji nisu bili poznati u fazi treninga. Na ovaj način se simulira prava namena klasi katora, a to je klasi kacija budućih, trenutno nepoznatih podataka. Jedan od osnovnih problema u fazi treniranja klasi katora je ocenjivanje kvaliteta klasi kacije izražene prethodno prikazanim ili nekim drugim merama. Potrebno je da ta ocena bude nepristrasna i da se ponaša dobro na test skupu podataka. Problem koji se često javlja kod klasi kacije je prekomerno prilago avanje modela trening podacima što uzrokuje nezadovoljavajuće ponašanje klasi katora na test skupu i na budućim nepoznatim podacima. Na primer, ukoliko bismo u slučaju klasi kacije stablom odlučivanja dopustili da stablo naraste dok god se ne izvrši korektna klasi kacija svih trening podataka, postoji velika mogućnost da bi to izazvalo prekomerno prilago avanje. Evidentno je da je u praksi potrebno napraviti kompromis po pitanju složenosti modela. Previše složeni modeli mogu dovesti do prekomernog prilago avanja, dok sa druge strane, previše jednostavni modeli mogu dovesti do nedovoljnog prilago avanja, situacije u kojoj model ne nauči dovoljno dobro ciljnu klasi kacionu funkciju. Rešavanje problema ocenjivanja kvaliteta se svodi na maksimizaciju budućeg kvaliteta klasi kacije, što u praksi znači maksimizaciju kvaliteta klasi kacije na test skupu podataka. U svim

opisanim pristupima, test skup se odvaja od trening skupa. Na kraju, kada je klasi kacioni model izgra en, vrši se jednokratna primena modela nad test skupom što proizvodi konačnu meru kvaliteta izgra enog klasi katora. Uobičajene podele skupa su: 2/3 za trening skup, 1/3 za test ili 1/2 za trening i 1/2 za test. Me utim, taj odnos zavisi i od tipa klasi katora. Kod metode podržavajućih vektora je dovoljan relativno mali trening skup, što joj kasnije omogućava visok stepen generalizacije na novim podacima. Podaci se biraju na slučajan način, što u teoriji, za dovoljno velike početne skupove podataka, omogućava da raspodela klasa i atributa bude ista u trening i u test skupu.

Ocena na trening skupu podatka (eng. hold-out estimation) je osnovni pristup. Ocena kvaliteta klasi kacije je jednostavno jednaka oceni kvaliteta na trening skupu. Ovo je vrlo optimistična pretpostavka, i često se pokazuje kao pristrasna ocena kvaliteta. Potrebno je napraviti kompromis po pitanju veličine trening skupa, jer preveliki trening skup može dovesti do prekomernog prilago avanja, a premali do nedovoljnog prilago avanja.

Iterativna podela na trening skup i skup za proveru (eng. random subsampling) se zasniva na višestrukoj slučajnoj podeli trening skupa podataka na dva dela. Prvi deo se u svakoj od tih podela koristi za treniranje modela, a drugi, pod nazivom skup za proveru, se koristi za proveru modela. Konačna ocena kvaliteta se dobija kao prosečna ocena kvaliteta nad svim skupovima za proveru. Ovo ima za cilj povećavanje sposobnosti uopštavanja klasi katora, ali ipak, dovodi do drugih problema. Očigledni problem je e kasnost, a drugi bitan nedostatak je to što ne postoji kontrola nad brojem pojavljivanja nekog podatka u skupu za proveru (ne postoje ni gornja ni donja granica u broju pojavljivanja nekog podatka u skupu za proveru).

Unakrsna provera (eng. cross-validation) je posebno značajna tehnika kada su polazni skupovi podataka male ili srednje veličine. Trening skup se deli na k disjunktnih podskupova (skupova za proveru) iste ili približne kardinalnosti. Ustaljene vrednosti za parametar k su iz opsega $2{-}10$. Klasa svakog vektora atributa se odre uje propuštanjem kroz klasi kator koji je formiran korišćenjem svih komponenti osim one kojoj posmatrani vektor atributa pripada. Nakon što se odredi kvalitet klasi kacije za sve skupove za proveru, ukupan kvalitet se dobija kao prosek tih vrednosti. Može se primetiti da se ovim pristupom rešava gore pomenuti problem sa kontrolom broja pojavljivanja podatka u skupu za proveru, jer se svaki podatak koristi tačno jednom za proveru. Me utim, problem e kasnosti i dalje je prisutan. Granični scenario primene unakrsne provere, kada je kardinalnost skupa za proveru jednaka 1, zove se izostavi-1-provera (eng. leave-one-out validation LOO). Iako daje skoro nepristrasnu ocenu kvaliteta klasi kacije, računarski je još intenzivnija od unakrsne provere sa ustaljenim vrednostima parametra k.

Butstrep ocena (eng. bootstrap estimation) za razliku od iterativne podele na trening i skup za proveru, i unakrsne provere, koristi slučajni odabir sa vraćanjem. Ako je trening skup dimenzije N_{tr}, vrši se N_{tr} slučajnih odabira podataka sa vraćanjem. Nakon što se izvrše svi slučajni odabiri, podaci koji nisu bili nijednom odabrani ulaze u skup za proveru, dok se preostali koriste za trening. Proces se potom ponavlja nekoliko puta, a ukupna ocena kvaliteta se dobija kao prosečna ocena na svim tako formiranim skupovima za proveru. S obzirom da je verovatnoća odabira jednog podatka $1/N_{tr}$, verovatnoća da podatak neće biti odabran nijedanput u N_{tr} pokušaja je $(1 - N_{tr})^{N_{tr}}$. Za dovoljno veliko N_{tr}, pomenuti izraz se asimptotski približava e^{-1}, što znači da će veličina skupa za proveru biti ≈ 0.368.

2.3.3 Optimizacija parametara klasi katora

Neke metode za klasi kaciju poseduju parametarsku strukturu, sačinjenu od jednog ili više parametara. Vrednosti parametara često imaju visok uticaj na kvalitet klasi kacije. Tradicionalni pristupi u rešavanju ovog problema su: 1) iterativno ručno podešavanje parametara; 2) sistematska pretraga parametara po mreži vrednosti (eng. grid search). Drugi pristup je zasnovan na podeli prostora mogućih parametara na ujednačene regione, a potom proveri kvaliteta klasi kacije u granicama tih regiona. Ograničenje tog pristupa je vremenska nee kasnost koja je posebno izražena u slučaju da je broj parametara, opseg vrednosti parametara ili preciznost podele na regione, visoka. Jedan od poznatijih problema podešavanja parametara klasi katora je potkresivanje stabla u slučaju klasi kacije stablom odlučivanja. Adekvatno podešavanje može dovesti do poboljšanja kvaliteta klasi kacije na test skupu, tj. do sprečavanja prekomernog prilago avanja klasi katora trening podacima. U kontekstu veštačkih neuronskih mreža, pojavljuje se problem podešavanja broja skrivenih čvorova u unutrašnjim slojevima, prag vrednosti čvorova, brzine učenja i drugi. Ako je broj skrivenih čvorova suviše mali, postoji mogućnost da mreža neće biti u stanju da aproksimira klasi kacionu funkciju. Za preveliki broj skrivenih čvorova, mreža će moći da nauči funkciju, ali će joj u tom procesu biti potrebno mnogo više vremena. Kod metode podržavajućih vektora, pojavljuje se problem podešavanja parametara kernelske funkcije, kao i regularizacionog parametra. Pravilnim odabirom ovih parametara, kvalitet klasi kacije se može značajno poboljšati.

2.3.4 Preprocesiranje podataka

Ulazni skupovi podataka obično imaju nedostatke koji ih čine neadekvatnim, a ponekad i neupotrebljivim za namene klasi kacije. Odre enim tehnikama preprocesiranja i optimizacije moguće je nadomestiti nedostatke i poboljšati kvalitet klasi kacije. Sada će biti izloženi samo neki od poznatijih problema koji se pojavljuju u fazi preprocesiranja.

Nedostajuće vrednosti podrazumevaju nepostojanje vrednosti atributa za podskup skupa ulaznih podataka. Razlozi za nepostojanje nekih vrednosti mogu biti različiti: greške u radu mernog instrumenta ako se radi o merenju zičkih fenomena, voljno uskraćivanje odgovora u slučaju da se radi o ispitanicima i dr. Načini razrešavanja ovih problema su tako e raznovrsni. Najjednostavniji pristup je eliminacija podataka. Postoje horizontalna i vertikalna eliminacija. Prva podrazumeva izbacivanje pojedinačnih podataka koji imaju nedostajuće atribute. Vertikalna eliminacija se odnosi na uklanjanje nekog atributa, i najčešće se primenjuje u slučajevima kada su nedostajuće vrednosti skoncentrisane oko tog atributa. Drugi pristup u rešavanju problema nedostajućih vrednosti je dodeljivanje vrednosti: nedostajuća vrednost atributa se može popuniti prosečnom vrednošću tog atributa na celom skupu u slučaju realnih vrednosti, medijanom u slučaju rednih atributa, ili najčešćom vrednošću u slučaju kategoričkih. Umesto vrednosti zasnovane na celom skupu, može se koristiti i vrednost zasnovana na skupu najbližih suseda. Treći pristup podrazumeva zadržavanje podataka sa nedostajućim vrednostima u skupu podataka, ali i ignorisanje nedostajućih vrednosti u slučaju da se javi potreba za njihovim korišćenjem. Ovo znači da se u procesu treniranja ili testiranja klasi kacionog modela, mere kvaliteta i druge potrebne vrednosti izračunavaju

različito za različite podatke. U slučajevima kada je broj ovakvih podataka mali, to ne mora dovesti do narušavanja kvaliteta klasi kacije.

Odabir atributa je značajan aspekt u preprocesiranju podataka i pripada široj klasi metoda koje se koriste za dimenzionu redukciju. Odabir atributa ima dvojaku ulogu. Prva uloga je smanjivanje dimenzije ulaznog problema što kao posledicu može imati drastično smanjenje vremena potrebnog za treniranje klasi kacionog modela. Druga uloga je da smanjivanjem broja atributa i sam klasi kacioni model postaje intuitivniji. Problem odabira atributa podrazumeva izdvajanje podskupa atributa koji su relevantni za proces klasi kacije. Jedan od alternativnih naziva ovog procesa je i vertikalna restrikcija podataka, jer ako se skup podataka posmatra tabelarno, atributi su predstavljeni kolonama. Postoji veliki broj metoda za rešavanje problema odabira atributa i sve one se mogu grupisati u tri kategorije: 1) lter metode (eng. lter methods), 2) omotač metode (eng. wrapper methods) i 3) ugnježdene metode (eng. embedded methods) koji formiraju podskup odabranih atributa u fazi treniranja klasi kacionog algoritma.

Filter metode su vrlo e kasne i obično koriste samo nekoliko prolaza kroz skup podataka kako bi izvršile eliminaciju nepotrebnih atributa. esto su zasnovane na svojstvima ulaznog skupa podataka: entropiji, informacionom kriterijumu, simetričnost, normalnosti podataka itd. Filter metode su preprocesirajuće u pravom smislu te reči. To se ne može reći za drugu grupu metoda, tzv. omotač metode. Odabir atributa kod ove grupe metoda je drugačiji. Koristi se optimizaciona tehnika kao omotač oko klasi kacionog modela. Optimizaciona tehnika potom traži takav podskup atributa koji, kada se prosledi kao ulaz u klasi kacioni model, maksimizuje funkciju cilja. Pokazuje se da su u praksi omotač metode kvalitetnije od lter metoda, ali imaju problem nasle ene (inherentne) vremenske nee kasnosti zbog načina na koji rešavaju problem odabira atributa.

Podešavanje težina atributa (eng. feature weighting) je problem odreivanja optimalnog stepena uticaja pojedinačnih atributa. Prema standardnoj strukturi klasi kacionog modela, svi uključeni atributi imaju isti značaj. Me utim, često neki od atributa nisu relevantni, ili poseduju šum, koji može da naruši kvalitet klasi kacije. U idealnom slučaju, podešavanjem težina atributa, irelevantni atributi dobijaju težine bliske ili jednake nuli, dok su vrednosti težina relevantnih atributa veće od nule i usaglašene sa njihovim relativnim značajem. Prema pristupu rešavanja ovog problema, postoje dve grupe metoda [6]: jednoprolazni metodi (eng. online optimization) koji uzimaju u obzir svojstva skupa podataka i višeprolazni (eng. batch optimization) koji koriste povratnu informaciju klasi katora. Prvom grupom metoda vrednosti težina atributa se ažuriraju sekvencijalnim prolaskom kroz trening skup. Problem koji se u tom procesu javlja je velika zavisnost konačnih težina od redosleda podataka u trening skupu. Drugom grupom metoda vrše se višestruki prolasci kroz trening skup, što eliminiše prethodni problem zavisnosti od redosleda podataka po ceni manje e kasnosti. U literaturi su predloženi različiti pristupi zasnovani na gradijentnim tehnikama, simuliranom kaljenju, genetskim algoritmima itd.

Odabir podataka (eng. instance selection) podrazumeva odabir najmanjeg podskupa podataka na osnovu kojeg je klasi kator u stanju da proizvede viši ili bar isti kvalitet klasi kacije. U literaturi se ovaj problem naziva još i problem horizontalne redukcije ulaznog skupa, jer se vrši eliminacija irelevantnih podataka koji su u tabeli podataka predstavljeni

kao redovi. Dobit od procesa odabira podataka je dvojaka: 1) dimenzija ulaznog skupa podataka je smanjena što može dovesti do velikog poboljšanja vremenske e kasnosti klasi katora; 2) eliminisani su podaci sa šumom, greškama, podaci van granica (eng. outliers) koji mogu umanjiti kvalitet klasi kacije. Prema [7], uobičajena su dva osnovna pristupa u rešavanju ovog problema:

1. Poboljšavanje uklanjanjem. Uklanjanjem odre enih podataka često je moguće povećati kvalitet klasi kacije. Ovo je moguće jer se uklanjaju podaci koji imaju šum, grešku ili jednostavno nisu dovoljno reprezentativni.

2. Nenarušavanje uklanjanjem. Uklanjanjem podatka zadržava se kvalitet klasi kacije. Ovo znači da je podatak nepotreban, odnosno redudantan, tako da se uklanjanjem doprinosi smanjenju dimenzije ulaznog problema.

U literaturi su predloženi i neki pristupi u kojima se integrisano posmatra horizontalna i vertikalna restrikciju skupa podataka, odnosno problem odabira atributa i problem odabira podataka.

Neravnomerna raspodela klasa. Za skup podataka sa dve klase se kaže da ima neravnomernu raspodelu klasa (eng. class imbalance) ako je odnos broja podataka jedne klase značajno manji od broja podataka druge klase. Ovo je čest problem u realnim skupovima podataka: podaci o retkim bolestima, otkrivanje prevara, nepoželjnih poruka, itd. Prema [8] postoje četiri osnovna pristupa u rešavanju ovog problema:

1. pristup zasnovan na ponovnom uzorkovanju (eng. resampling) u cilju balansiranja klasa;

2. izmena postojećeg algoritma učenja (izmena na algoritamskog nivou);

3. izmena u načinu merenja kvaliteta klasi kacije (prilago avanje mere kvaliteta);

4. pristupi zasnovani na vezama izme u neravnomernosti raspodele klasa i ostalih karakteristika složenosti.

Pregled poznatih klasi katora

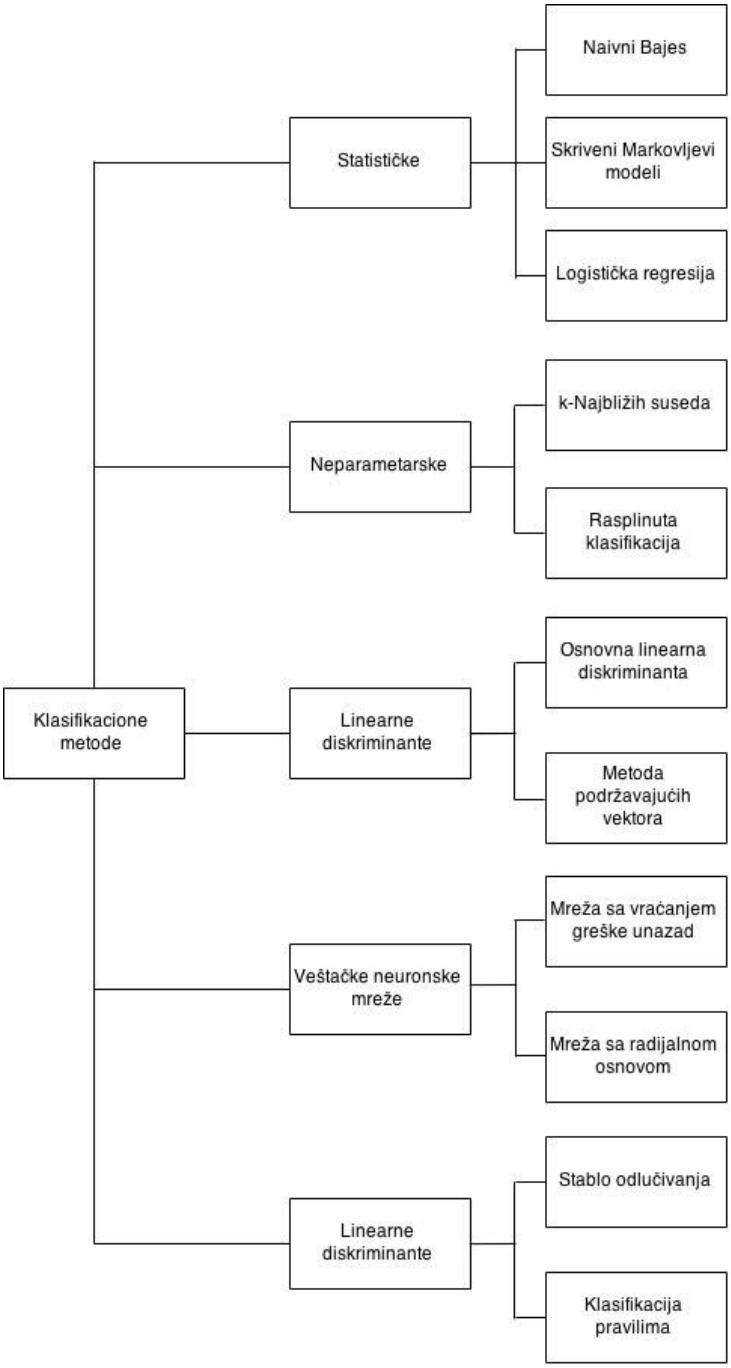
U [5] predložena je podela metoda za klasi kaciju koja je šematski prikazana na Slici 3.1. Za svaku grupu metoda navedeno je po nekoliko važnih i/ili popularnih predstavnika od kojih je deo kasnije detaljnije predstavljen u ovom poglavlju.

Kada su u pitanju metode statističke klasi kacije, u Sekciji 2.1 su prikazani neki aspekti Bajesove teorije odlučivanja. Oni su podrazumevali da je raspodela verovatnoće klasa unapred poznata. Kada su takve informacije dostupne, Bajesov klasi kator se može koristiti kao zlatni standard za pore enje sa drugim metodama. U praksi je, me utim, češća situacija da raspodela verovatnoće klasa nije dostupna, ali i tada je moguće primeniti odre ene statističke tehnike. Na primer, ako je poznat tip raspodele (funkcionalna forma), onda se parametri raspodele mogu aproksimirati tehnikama poput metode maksimalne

verodostojnosti, ili drugim egzaktnim i neegzaktnim optimizacionim tehnikama. Metoda logističke regresije zasnovana je na ovom konceptu i biće prikazana u sekciji koja sledi.

Kada ni funkcionalna forma raspodele nije poznata, tj. kada postoji potpuno odsustvo bilo kakvih informacija o strukturi raspodele verovatnoće klasi kacionog problema, mogu se primeniti neparametarske tehnike. Jedna od poznatijih neparametarskih metoda je metoda k-najbližih suseda o kojoj će biti reči u narednoj sekciji ovog poglavlja.

Sledeća grupa metoda se zovu linearne diskriminantne funkcije. Kod njih se koristi linearna funkcionalna forma kao klasi kacioni model. Potom se vrši optimizacija parametara te funkcionalne forme. U narednim sekcijama ovog poglavlja će biti predstavljene dve metode iz ove grupe, osnovna linearna diskriminantna funkcija i metoda podražavajućih vektora, obe na primerima binarne klasi kacije.



Slika 3.1: Podela klasi kacionih metoda

Veštačke neuronske mreže predstavljaju zasebnu grupu metoda kod kojih je funkcija učenja (klasi kacije) nelinearna. Veštačka neuronska mreža je moćna aproksimativna metoda sposobna da nauči funkcije koje u osnovi imaju visok stepen nelinearnosti. Detaljniji pregled ove tehnike izlazi iz okvira ovog rada.

Nemetričke metode nemaju jasnu funkcionalnu formu, niti statističke elemente. One se najbolje mogu opisati kao skupovi logičkih pravila. U jednoj od narednih sekcija je opisana i opšta struktura algoritma zasnovanog na stablu odlučivanja. U ovoj poglavlju se dalje izlažu

neke od klasi kacionih tehnika koje pripadaju različitim klasi kacionim grupama iz podele prikazane na Slici 3.1.

3.1 Metoda *k*-najbližih suseda (k-NN)

Metoda k-najbližih suseda (eng. k-nearest neighbors k-NN) predstavlja neparametarsku klasi kacionu tehniku koja klasi kuje zadati vektor atributa na osnovu skupa od k najbližih suseda tog vektora. Pri tom se pod najbližim susedima misli na podatke iz trening skupa podataka koji imaju najviši stepen sličnosti vektora atributa sa posmatranim vektorom atributa. Nakon što se odredi k takvih podataka, posmatrani vektor atributa se klasi kuje u skladu sa klasom koja preovladava u skupu suseda. Kada se primenjuje u rešavanju binarnih klasi kacionih problema, vrednosti parametra k obično uzimaju neparne vrednosti. Ovo omogućava da se uvek donese jednoznačna odluka po pitanju klase.

Formalna de nicija klasi kacionog problema u kontekstu metode najbližih suseda glasi:

De nicija 3. Neka je dat skup trening podataka D_{tr} koji se sastoji od N_{tr} ureenih parova oblika $(\mathbf{x}^{(i)},y^{(i)}),i = 1,\ldots,N_{tr}$, gde je $\mathbf{x}^{(i)} \in \mathbf{R}^N$ N-dimenzionalni vektor atributa, a $y^{(i)} \in \{1,\ldots,N_c\}$ odgovarajuća klasa. Za novi vektor atributa $\mathbf{x}$, k-NN pronalazi skup sačinjen od k trening podataka $\{(\mathbf{x}^{(i_1)},y^{(i_1)}), \ldots,(\mathbf{x}^{(i_k)},y^{(i_k)})\}$, koji imaju najmanje vrednosti funkcije udaljenosti $dist$: $\mathbf{R}^N \times \mathbf{R}^N \rightarrow \mathbf{R}$ u odnosu na vektor atributa. Funkcija udaljenosti (različitosti) se računa za svaki par formiran od svih trening vektora atributa sa jedne strane i novog vektora atributa sa druge: $dist(\mathbf{x},\mathbf{x}^{(i)}), i = 1,\ldots,N_{tr}$. Nakon što se utvrde sve vrednosti funkcije udaljenosti, vrši se ure ivanje po toj vrednosti u rastućem poretku i nakon toga prvih k trening vektora atributa se bira u skup najbližih suseda. Funkcija odre ivanja klase novog vektora atributa se potom dobija pomoću formule

$$c(\mathbf{x}) = \arg\max_{m} \left(\sum_{j=1}^{k} \mathbb{1}\{y^{(i_j)} = m\} \right), m = 1, ..., N_c \quad . \qquad (3.1)$$

Izrazom $1\{y^{(i_j)} = m\}$ je predstavljena indikatorska funkcija koja uzima vrednost 1 ako je $y^{(i_j)} = m$, a u suprotnom uzima vrednost 0. Kao što se može videti iz prikazanog, k-NN metoda je relativno jednostavna za implementaciju i za razumevanje. Faza učenja klasi katora ne postoji u klasičnom smislu. Sve relevantne operacije, koje čine klasi kacioni model, se izvršavaju tek kada započne primena klasi katora nad konkretnim vektorom atributa. Ovo svojstvo ima za posledicu da vremenska e kasnost zavisi od dimenzije trening skupa podataka i broja atributa svakog od podataka, jer je npr. složenost izvršavanja 1-NN za svaki pojedinačni trening vektor jednaka $O(N_{tr}N)$. U cilju poboljšavanja vremenske e kasnosti, u literaturi su predložena mnogobrojna poboljšanja osnovnog k-NN algoritma, poput paralelizacije pretrage najbližih suseda, parcijalnog računanja funkcije udaljenosti i drugih.

3.2 Linearna diskriminantna funkcija

U linearnim diskriminantnim modelima se pravi pretpostavka da je adekvatna forma funkcionalnog klasi katora linearna funkcija. Potom se vrši optimizacija funkcije cilja po

parametrima te funkcionalne forme. Funkcija cilja je najčešće predstavljena greškom klasi kacije na trening podacima, što može dovesti do prekomernog prilago avanja klasi kacione funkcije trening skupu. Posledica prekomernog prilago avanja je da se klasi kator ne ponaša dobro na podacima van trening skupa, tzv. test podacima. Opšta forma linearne diskriminativne funkcije je

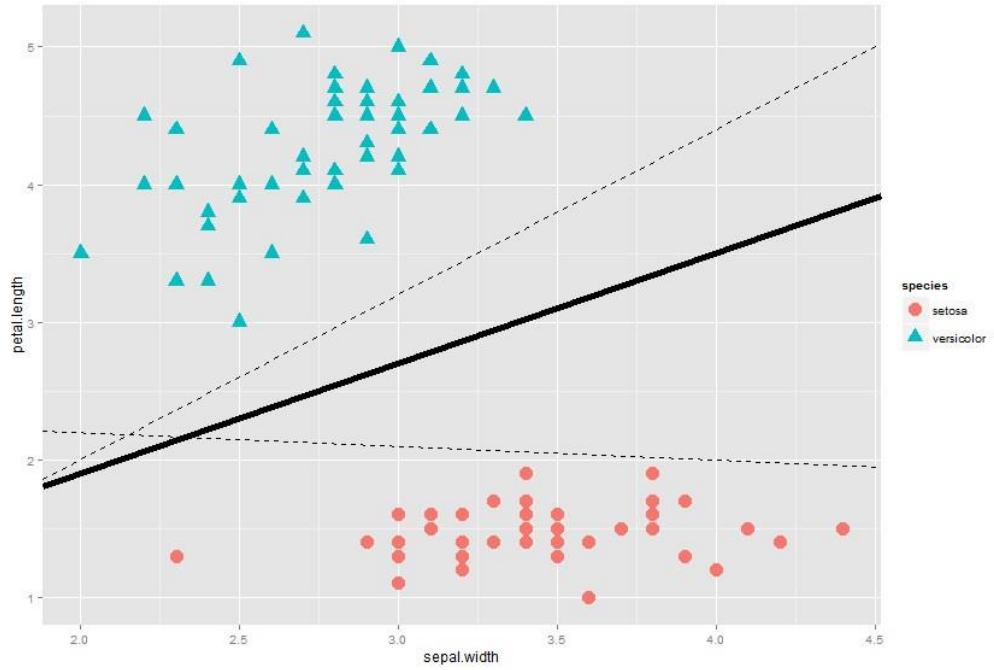
$$f(\mathbf{x}) = \omega_0 + \omega_1 x_1 + \cdots + \omega_N x_N,\ \omega_i \in \mathbf{R},\ i = 1,\ldots,N, \tag{3.2}$$

gde su ω_i, $i = 1,\ldots,N$ težine dodeljene atributima vektora, dok je ω_0 slobodni član ili težinski prag (eng. threshold weight). Funkcija f de niše hiperravan (marginu) koja razdvaja oblasti pripadnosti dve klase. Uobičajena pridružena funkcija odlučivanja za (3.2) je data kao

$$c(\mathbf{x}) = \begin{cases} -1, & f(\mathbf{x}) < 0 \\ 1, & f(\mathbf{x}) \geq 0 \end{cases}. \tag{3.3}$$

Razmotrimo sada slučaj kada podaci imaju svojstvo linearne razdvojivosti na trening skupu. Ovo znači da postoji takav vektor koe cijenata $\omega = (\omega_0,\ldots,\omega_N)$ za koji važi da, nakon što se uvrsti u funkciju f, pridružena funkcija odlučivanja daje korektne klasi kacije za sve trening podatke.

Rezultujući vektor koe cijenata nije jedinstven kao što se može videti na Slici 3.2. Iako sve tri razdvajajuće hiperravni korektno klasi kuju sve trening vektore atributa, pojačana hiperravan ispunjava i dodatni kriterijum jer minimizuje sumu rastojanja trening vektora od razdvajajuće hiperravni. Ispostavlja se da je ovo svojstvo ključno za klasi katore koji pripadaju grupi klasi katora u kojima se maksimizuje margina (eng. maximal margin classi ers). Najpoznatiji u ovoj grupi su metoda podržavajućih vektora i AdaBoost metod.



Slika 3.2: Primeri razdvajajućih hiperravni na Iris skupu sa dve klase

U osnovnom modelu linearne diskriminativne funkcije dodatni uslov o maksimalnosti margine ne postoji. Problem optimizacije koe cijenata stoga razmatra samo uslov linearne razdvojivosti, i rešenja za ovaj problem se mogu naći upotrebom osnovnih numeričkih algoritama poput gradijentnog spusta, Njutnove metode i dr. Ovakav pristup koristi metoda Logističke regresije.

3.3 Logistička regresija (LR)

Logistička regresija (eng. logistic regression - LR) je jedna od najpopularnijih tehnika klasi kacije koje se danas koriste. Odlikuju je jednostavnost modela na kome je zasnovana, primenjivost u velikom broju klasi kacionih problema, kao i dostupnost velikog broja e kasnih i skalabilnih implementacija u javno dostupnim (eng. open-source) bibliotekama. U okviru ove sekcije biće razmatran matematički model reševanja problema binarne klasi kacije predstavljen u [13].

U osnovi, LR pretpostavlja parametarski oblik raspodele $P(y|\mathbf{x})$, a onda vrši direktnu procenu tih parametara pomoću trening podataka. Pretpostavljena raspodela je data kao

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{w_0+\mathbf{w}\cdot\mathbf{x}}} \tag{3.4}$$

i

$$P(y = -1|\mathbf{x}) = \frac{e^{w_0+\mathbf{w}\cdot\mathbf{x}}}{1 + e^{w_0+\mathbf{w}\cdot\mathbf{x}}}. \tag{3.5}$$

U prethodnim izrazima $\mathbf{w}$ predstavlja vektor realnih parametara dimenzije N, dok je $w_0 \in \mathbf{R}$ slobodni koe cijent. Jednačina (3.5) sledi direktno iz (3.4), jer suma ove dve verovatnoće mora biti jednaka 1.

Po Bajesovoj teoriji odlučivanja, a u datim uslovima, iz (2.4) lako se dolazi do funkcije odlučivanja

$$c(\mathbf{x}) = sgn(w_0 + \mathbf{w} \cdot \mathbf{x}). \tag{3.6}$$

Vrednosti parametara w_0 i $\mathbf{w}$ procenjuju se upotrebom prinicipa maksimalne verodostojnosti (eng. maximum likelihood). Parametri se biraju tako da verovatnoća dobijanja posmatranih trening podataka bude maksimalna, odnosno

$$(w_0,\mathbf{w}) = \operatorname{argmax}^{(w_0,\mathbf{w})} \left(\prod_{i=1}^{N_c} P(y^{(i)}|\mathbf{x}^{(i)}, (w_0, \mathbf{w})) \right). \tag{3.7}$$

Primenom logaritamske funkcije na obe strane prethodnog izraza, kao i smenom pretpostavljenih raspodela iz (3.4) i (3.5) dolazi se do oblika pogodnog za rešavanje poznatom tehnikom gradijentnog spusta

$$(w_0,\mathbf{w}) = \text{argmax}^{(w_0,\mathbf{w})} \left(\sum_{i=1}^{N_c} y^{(i)}(w_0 + \mathbf{w}\cdot\mathbf{x}) - ln(1 + e^{w_0+\mathbf{w}\cdot\mathbf{x}}) \right) \quad . \qquad (3.8)$$

3.4 Klasi kacija metodom podržavajućih vektora (SVM)

Metoda podržavajućih vektora (eng. support vector machine - SVM) je tehnika mašinskog učenja sa širokim primenama. SVM pripada grupi linearno diskriminativnih metoda u kojima postoji i dodatni uslov o maksimalnosti margine. Dva glavna domena primene su: 1) klasi kacioni problemi u kojima se vrši predviđanje diskretne promenljive i 2) odre ivanje funkcionalnih formi u problemima regresije u kojima se vrši predviđanje neprekidne promenljive.

Teorijske osnove SVM su date u [9] i [10]. Motivacija za primenu SVM u klasi kaciji potiče od rezultata statističke teorije učenja u kojoj je razvijena gornja granica greške generalizacije ove metode. Gornja granica greške je minimalna kada je rastojanje izme u vektora i razdvajajuće hiperravni maksimalno. Važno praktično svojstvo gornje granice je njena nezavisnost od dimenzije prostora atributa. Me utim, formiranje razdvajajuće hiperravni nije uvek moguće i u takvim sitacijama se kaže da prostor atributa nije linearno razdvojiv. Ovo svojstvo nerazdvojivosti se može zaobići preslikavanjem originalnog prostora atributa u drugi prostor koji poseduje svojstvo linearne razdvojivosti. Transformacija prostora dovodi do povećanja dimenzije problema, me utim, to ne utiče na e kasnost SVM metode, s obzirom na činjenicu da SVM ne koristi vektore atributa direktno. Umesto direktne upotrebe trening vektora, SVM koristi funkciju sličnosti izme u parova vektora. Funkcija sličnosti se naziva kernel (jezgro), i njeno svojstvo od visokog praktičnog značaja je da se može izračunati u originalnom prostoru atributa. Stoga, bilo kakvo povećanje dimenzije prostora nema uticaja na složenost izračunavanja SVM algoritma.

Sada će biti prikazana matematička formulacija problema binarne klasikacije prilago ena rešavanju SVM algoritmom. Za klasi kacione probleme sa više od dve moguće klase, pored opštih pristupa prikazanih u Sekciji 2.2, u literaturi je predstavljen uni kovani pristup za klasi katore zasnovane na marginama, me u kojima je i SVM.

De nicija 4. Neka je dat skup trening podataka D_{tr} sačinjen od N_{tr} ure enih parova oblika $(\mathbf{x}^{(i)},y^{(i)}), i = 1,\ldots,N_{tr}$, gde je $\mathbf{x}^{(i)} \in \mathbf{R}^N$ N-dimenzionalni vektor atributa, a $y^{(i)} \in \{-1,1\}$ njegova pripadajuća klasa. SVM koristi trening skup D_{tr} u cilju pronalaženja razdvajajuće hiperravni $\mathbf{w}\cdot\mathbf{x}+b = 0$, $\mathbf{w} \in \mathbf{R}^N$, $b \in \mathbf{R}$ koja je maksimalno udaljena od svih trening podataka (vektora) sa obe strane. Razdvajajuća hiperravan (margina) se lako transformiše u funkciju odlučivanja $c(\mathbf{x}) = sgn(\mathbf{w}\cdot\mathbf{x}+b)$ koja računa klasu za dati ulazni vektor atributa.

Striktno linearno razdvojivi podaci su vrlo retki u praksi, stoga postoji potreba za dopuštanjem odre ene greške po pitanju razdvojivosti podataka. Manje striktna varijanta problema, tzv. klasi kacija bazirana na mekoj margini (eng. soft margin classi er) je predložena u [11]. Umesto postojanja striktne margine izme u trening vektora različite klase, meka varijanta problema dopušta vektorima da se nalaze sa pogrešne strane, ali ipak dovoljno blizu margine: $y^{(i)}(\mathbf{w}\cdot\mathbf{x}^{(i)} + b) \geq 1 - \zeta_i$, $i = 1,\ldots,N_{tr}$, a ζ_i je nenegativna promenljiva koja predstavlja grešku prelaska vektora $x^{(i)}$ na pogrešnu stranu hiperravni. Na ovako modi

kovanoj de niciji razdvajajuće hiperravni, optimalne vrednosti za w i b mogu biti prona ene rešavanjem optimizacionog problema

$$min\left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N_{tr}}\zeta_i\right), \tag{3.9}$$

uz ograničenja

$$y^{(i)}(\mathbf{w}\cdot\mathbf{x}^{(i)} + b) \geq 1 - \zeta_i,\ \zeta_i \geq 0,\ i = 1,\ldots,N_{tr}. \tag{3.10}$$

C je regularizacioni (kazneni) parametar koji kontroliše uticaj grešaka prelaženja.

SVM koristi pogodnu dualnu reprezentaciju u cilju odre ivanja margine. Dualna formulacija je data bez izvo enja (u [12] je detaljno opisan postupak njenog dobijanja na osnovu primalne formulacije)

$$max\left(\sum_{i=1}^{N_{tr}}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N_{tr}} y^{(i)}y^{(j)}\alpha_i\alpha_j\mathbf{x}^{(i)}\mathbf{x}^{(j)}\right), \tag{3.11}$$

uz ograničenja

$$\alpha_i \in [0,C], i = 1,\ldots,N_{tr} \tag{3.12}$$

i

$$\sum_{i=1}^{N_{tr}} \alpha_i y^{(i)} = 0 \tag{3.13}$$

Vrednosti α_i, $i = 1,\ldots,N_{tr}$ predstavljaju Lagranžove množioce, odnosno koe cijente, dok C u ovoj formulaciji predstavlja njihovu gornju granicu. Sledstveno, C kontroliše sveukupni maksimizacioni izraz pravljenjem kompromisa izme u maksimizacije margine i minimizacije greške. Funkcija odlučivanja za datu formulaciju se računa kao

$$c(\mathbf{x}) = sgn\left(\sum_{i=1}^{N_{tr}}\alpha_i y^{(i)}\mathbf{x}^{(i)}\mathbf{x} + b\right). \tag{3.14}$$

3.4.1 Kernel funkcije

U prethodnom odeljku su opisani osnovni elementi metode podržavajućih vektora sa pratećim formulacijama klasi kacionih problema u slučajevima da su podaci linearno razdvojivi, ili skoro linearno razdvojivi (slučaj meke margine). U slučaju da podaci nisu linearno razdvojivi i da upotreba meke margine nije dovoljna, primenjuje se transformacija nad prostorom atributa.

U tom svojstvu se vrši zamena svakog vektora $\mathbf{x}$ sa $\mathbf{\Phi}(\mathbf{x})$, gde $\mathbf{\Phi} : \mathbf{R}^N \rightarrow \mathbf{R}^N$ preslikava originalni prostor atributa u prostor više dimenzije, u kojem je linearna razdvojivost podataka moguća.

Preslikani prostor atributa je veće dimenzije, pa se nameće pitanje e kasnosti. Najpre je potrebno preslikati originalne vektore atributa u novi prostor, a potom u novom prostoru, koji je nužno veće dimenzije, izvršiti sva potrebna izračunavanja. Sva izračunavanja nad podacima u preslikanom prostoru su zasnovana na skalarnom proizvodu. Ovo predstavlja bitnu polaznu pretpostavku za uvo enje mehanizma koji se naziva kernelski trik (eng. kernel trick). Kernelski trik je mehanizam koji omogućava da se preslikavanje i dimenzija preslikanog prostora zanemari sa aspekta vremenske e kasnosti. Suština leži u upotrebi funkcionalne forme koja se naziva kernel:
$K(\mathbf{u},\mathbf{v}) = \mathbf{\Phi}(\mathbf{u})\cdot\mathbf{\Phi}(\mathbf{v})$. Ključno svojstvo kernela je da se on primenjuje nad vektorima u preslikanom prostoru atributa, ali se istovremeno može izračunati i u originalnom prostoru atributa. Ovo omogućava da nikad ne dolazi do direktne upotrebe funkcije $\mathbf{\Phi}$ i, samim tim, dimenzija ciljnog prostora ne igra ulogu u e kasnosti izračunavanja.

3.5 Klasi kacija pomoću stabla odlučivanja

Stablo odlučivanja klasi kuje posmatrani vektor atributa tako što propušta taj vektor kroz unutrašnje čvorove sve do listova. Svakom od unutrašnjih čvorova je pridruženo pitanje koje usmerava vektor atributa na neki od svojih čvorova potomaka. Pitanje se odnosi na vrednost nekog od mogućih atributa. Neophodno je da odgovor na pitanje bude jednoznačan, odnosno da usmerava ulazni vektor atributa na tačno jednog potomka. Vektoru atributa klasa biva dodeljena kada dostigne neki od listova stabla. Neke od poznatijih metoda iz ove grupe su: ID3, C4.5, C5.0, CART, CHAID i dr. Svi ovi algoritmi koriste isti princip rekurzivnog formiranja stabla pod nazivom Hantov algoritam (eng. Hunt's algorithm). Opšta struktura Hantovog algoritma se sastoji iz sledeća dva koraka:

1. Izlazak iz rekurzije: Ako su svi trening podaci pridruženi posmatranom čvoru drveta u istoj klasi, pravi se list drveta i označava tom klasom;

2. Rekurzivni korak: Ako podaci pridruženi posmatranom čvoru drveta nemaju jedinstvenu klasu, bira se atribut koji najefektivnije razdvaja trening skup podataka na podskupove. Kriterijumi efektivnosti razdvajanja mogu biti različite metrike: entropija, informacioni dobitak, Gini koe cijent, tačnost klasi kacije i dr. Nakon što se izvrši odabir najboljeg atributa, vrši se podela podataka prema mogućim vrednostima atributa, ili prema diskretizovanim intervalima u slučaju atributa sa realnim vrednostima ili velikim brojem mogućih diskretnih vrednosti. Konačno, na dobijenim podskupovima se primenjuje celokupni postupak rekurzivno.

.

Klasi katori iz ove grupe metoda su relativno jednostavni za implementaciju i vrlo e kasni po pitanju klasi kovanja novih vektora atributa. Njihov osnovni nedostatak leži u činjenici da je reč o pohlepnim algoritmima pretrage, što može dovesti do nezadovoljavajuće tačnosti klasi kacije kod odre enih klasa problema.

Primena klasi katora na predviđanje ishoda mečeva EPL

Ovo poglavlje bavi se predviđanjem krajnjih ishoda fudbalskih utakmica Premijer lige Engleske (eng. English Premier League - EPL) korišćenjem poznatih metoda za klasi kaciju. Svakoj utakmici predviđa se jedan od tri ishoda:

1. Pobeda domaćeg tima (H),
2. Nerešen ishod (D),
3. Pobeda gostujućeg tima (A).

Na osnovu istorijskih podataka, svakom meču dodeljen je skup atributa koji uzima u obzir informacije o konkretnom susretu koji se predviđa, kao i pokazatelje o trenutnoj formi svakog od timova. Navedeni atributi upotrebljeni su za treniranje nekoliko različitih klasi katora zasnovanih na metodama logističke regresije (LR) i podržavajućih vektora (SVM). Analiziran je kvalitet dobijenih klasi kacionih modela, problem podešavanja parametara klasi katora, kao i optimalnog izbora atributa za svaki od predloženih pristupa.

4.1 Metodologija

Obučavanje svih predloženih klasi katora obavljeno je korišćenjem trening seta koji se sastoji od rezultata utakmica iz tri sezone (od sezone 2010/11 do 2012/13). Ocena kvaliteta modela izvršena je na test setu koje čine rezultati susreta iz sezone 2013/14. Kako se timovi me usobno susreću dva puta tokom sezone, u trening i test setu nalazi se 1140, odnosno 380 mečeva. Za svaki od njih dostupne su informacije o imenima timova koji igraju, krajnjem rezultatu, broju šuteva na gol, posedu lopte i broju izvedenih kornera svakog od timova. Ovi podaci preuzeti su sa javno dostupnog registra istorijskih podataka o fudbalskim utakmicama Football-Data [14] i obra eni u programskom jeziku R. Programski jezik R izabran je kao pogodan za ovu analizu jer je namenjen za statističku obradu i prikazivanje podataka. Dostupan je i veliki broj biblioteka sa e kasnim implementacijama standardnih statističkih metoda.

4.1.1 Preliminarna razmatranja

Glavni faktori koji ovaj zadatak čine teškim su mala količina dostupnih trening podataka u relevantnim vremenskim periodima i visok nivo entropije tih podataka. Značajan broj nerešinih ishoda, koje drugi popularni sportovi nemaju ili se dešavaju retko, problem dodatno otežava. Na primer, od 380 utakmica odigranih u sezoni 2013/14, 179 se završilo pobedom domaćina, 123 pobedom gostujućeg tima, dok je 78 imalo nerešen ishod. Ovo daje izraz za entropiju sezone:

$$H = -\left(\frac{179}{380} log_3\left(\frac{179}{380}\right) + \frac{123}{380} log_3\left(\frac{123}{380}\right) + \frac{78}{380} log_3\left(\frac{78}{380}\right)\right) = 0.95. \quad (4.1)$$

Vrednost entropije 1 odgovara raspodeli potpuno slučajne promenljive, pa imajući u vidu da je dobijena vrednost veoma bliska, može se zaključiti da bi slučajno predvidjanje dalo tačnost od 33.33%. Još jedan trivijalan, ali značajno bolji pristup bio bi predviđanje pobede domaćina u svim slulajevima, što bi postiglo tačnost od oko 47%.

Pored toga što je raspodela ishoda skoro potpuno slučajna, u fudbalu nije potpuno neuobičajeno da tim kao što je Wigan trijumfuje protiv Arsenala, tima koji se smatra daleko boljim. Ovakvi događaji čine predviđanje rezultata još težim.

4.1.2 Izbor atributa

Za relevantne pokazatelje učinka tima na utakmici izabrani su:

- Broj postignutih golova
- Broj šuteva u okvir gola
- Broj izvedenih udaraca iz ugla

Golovi predstavljaju očigledan izbor, jer je tim sa više postignutih golova pobednik susreta. Udarci u okvir gola i iznu eni udarci iz ugla su, tako e, indikatori dobre igre tima. Posed lopte nije izabran kao faktor zato što je smatran posledicom načina igre, a ne nužno boljeg ostavljenog utiska.

Odabir atributa trebalo bi izvršiti uzimajući u obzir formu tima u vremenskom periodu koji neposredno prethodi datoj utakmici. Tako e, treba imati u vidu prednost domaćeg terena, kao i skorašnje uspone i padove u igri. Na osnovu datih poželjnih karakteristika predlažu se sledeći načini odabira vektora atributa:

Poslednjih k učinaka (PKU)

U ovom pristupu za odabir atributa koji predviđaju susret timova A i B koristi se poslednjih k učinaka svakog od timova. Ako je $\mathbf{g}^{(A)} = (g_1^{(A)}, g_2^{(A)}, ..., g_k^{(A)})$ vektor koji sadrži broj postignutih golova u prethodnih k susreta tima A, računa se

$$g^{(A)} = g_1^{(A)} + g_2^{(A)}\lambda + g_3^{(A)}\lambda^2 + \cdots + g_k^{(A)}\lambda^{k-1}, \qquad (4.2)$$

gde je $\lambda \in [0,1]$ realan parametar uveden kako bi učinci na utakmicama odigranim u skorije vreme bili relevantniji od onih odigranih ranije. Analogno se izračunavaju i parametri $st^{(A)}$ i $c^{(A)}$ za šuteve u okvir gola i kornere. Na ovaj način dobija se vektor $\mathbf{f}^{(A)} = (g^{(A)}, st^{(A)}, c^{(A)})$ koji predstavlja indikator forme tima A u prethodnih k utakmica. Na sličan način izračunava se i $\mathbf{f}^{(B)}$ koji dalje daje predloženi vektor atributa

$$\mathbf{f} = \mathbf{f}_{(A)} - \mathbf{f}_{(B)}. \qquad (4.3)$$

Operator oduzimanja je ovde upotrebljen kako bi vektor $\mathbf{f}$ u sebi sadržao informaciju o tome koji tim je domaćin, a koji gost na utakmici.

Gradijent poslednjih k učinaka (GPKU)

Slično prethodnom pristupu, prvo se dolazi do vektora $\mathbf{g}^{(A)}$, a osim vrednosti $g^{(A)}$ izračunava se i

$$g_{grad}^{(A)} = \left(g_1^{(A)} - g_2^{(A)}\right) + \left(g_2^{(A)} - g_3^{(A)}\right)\lambda + \cdots + \left(g_{k-1}^{(A)} - g_k^{(A)}\right)\lambda^{k-2}. \quad (4.4)$$

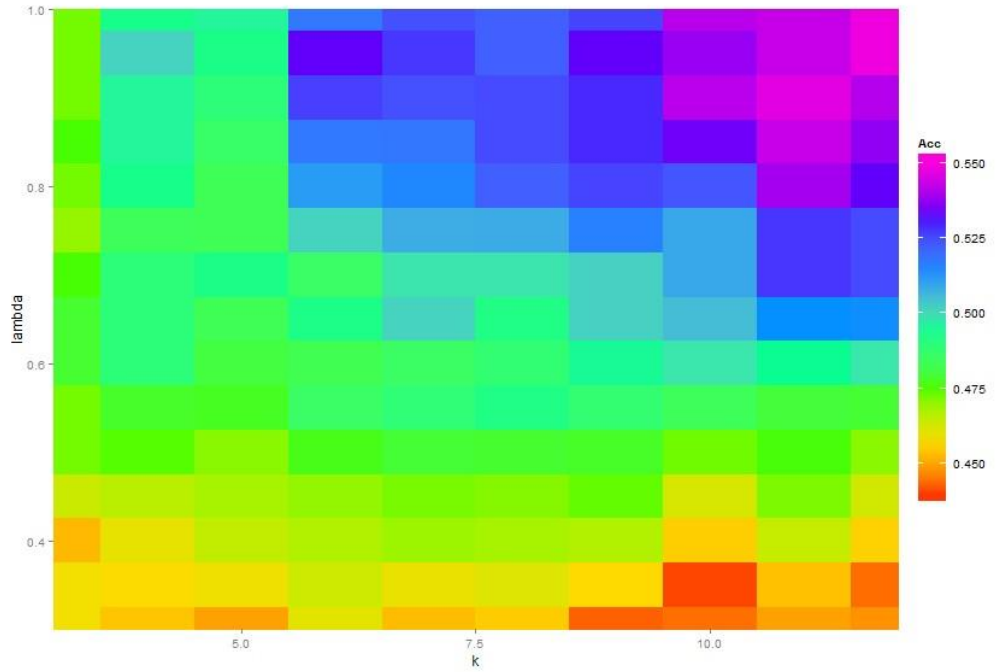
Analogno se dolazi do $st_{grad}^{(A)}$ i $c_{grad}^{(A)}$, a predloženi vektor atributa onda iznosi

$$\mathbf{f} = \left(g^{(A)} - g^{(B)}, st^{(A)} - st^{(B)}, c^{(A)} - c^{(B)}, g_{grad}^{(A)} - g_{grad}^{(B)}, st_{grad}^{(A)} - st_{grad}^{(B)}, c_{grad}^{(A)} - c_{grad}^{(B)}\right). \quad (4.5)$$

Gradijenti su u ovom pristupu upotrebljeni kako bi se modelovali usponi i padovi forme timova u vremenu koje neposredno prethodi utakmici koja se predviđa.

Nameće se pitanje pristupa odabiru atributa u prvih k utakmica, za koje ne postoji potrebna količina podataka da bi se upotrebili prethodno opisani metodi. Razmatrane su mogućnosti skaliranja parametra k u formulama u zavisnosti od broja dostupnih pre ašnjih utakmica, korišćenje susreta iz prethodne sezone, kao i ignorisanje takvih utakmica pri treniranju i oceni kvaliteta. Svi rezultati dati u okviru ovog rada dobijeni su korišćenjem trećeg pristupa. Odabir parametara k i λ izvršen je upotrebom metoda pretrage mreže uz ocenu kvaliteta unakrsnom proverom sa 5 podskupova ($k_{up} = 5$). Rezultati ove pretrage, uz primenu LR klasi katora i odabira atributa metodom KPU (LR_{KPU}), dati su na Slici 4.1. Za parametar k razmatrane su vrednosti $3 - 12$, a za λ pretražen je interval $[0.2,1]$ sa korakom 0.05. Odabrane su vrednosti:

$$k = 11, \lambda = 0.9. \quad (4.6)$$



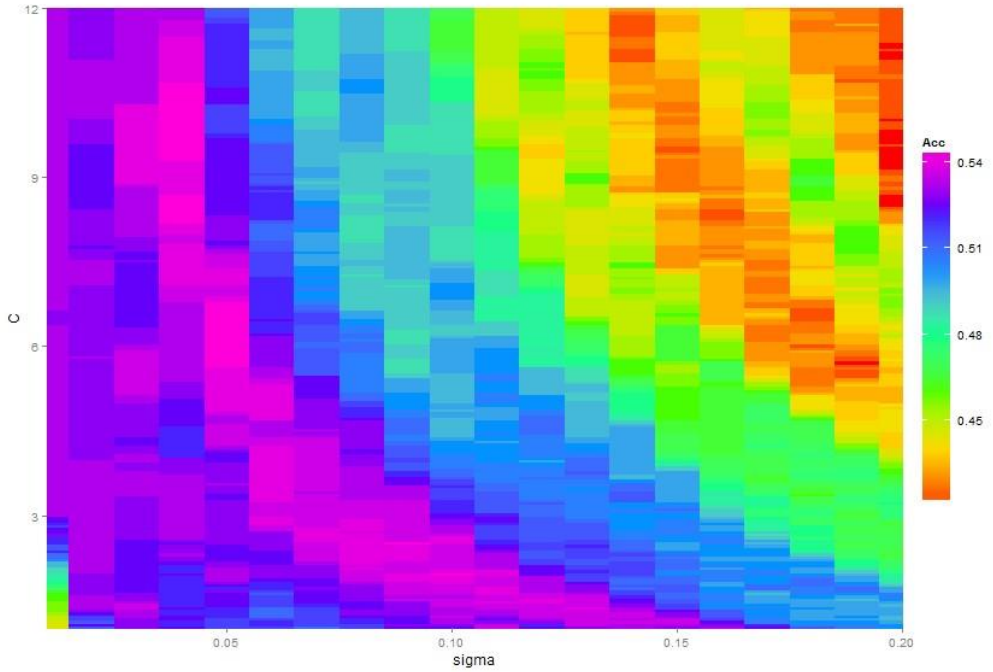
Slika 4.1: Zavisnost tačnosti klasi katora LR_{KPU} od parametara k i λ

4.1.3 Klasi kacioni modeli

Klasi katori koji su se u literaturi ([15], [16], [17]) pokazali kao relevantni za ovaj problem su uglavnom zasnovani na LR i SVM pristupima, pa su i u ovom radu razmatrane njihove varijacije. Prvi razmatrani pristup, a koji se pokazao najbolje u literaturi, je zasnovan na metodi podržavajućih vektora sa kernel funkcijom radijalne baze (eng. Radial Basis Function - RBF)

$$K(\mathbf{u}, \mathbf{v}) = e^{\frac{\|u-v\|^2}{2\sigma^2}}. \tag{4.7}$$

Parametri ovog klasi katora C i σ podešeni su metodom pretrage mreže uz ocenu kvaliteta unakrsnom proverom (k_{up} = 5). Rezultati ove pretrage dati su na Slici 4.2. Za parametar C razmatran je interval $[1,12]$ sa korakom 0.05, a za σ interval $(0,0.2]$ sa korakom 0.01. Odabrane su vrednosti C = 2 i σ = 0.08.

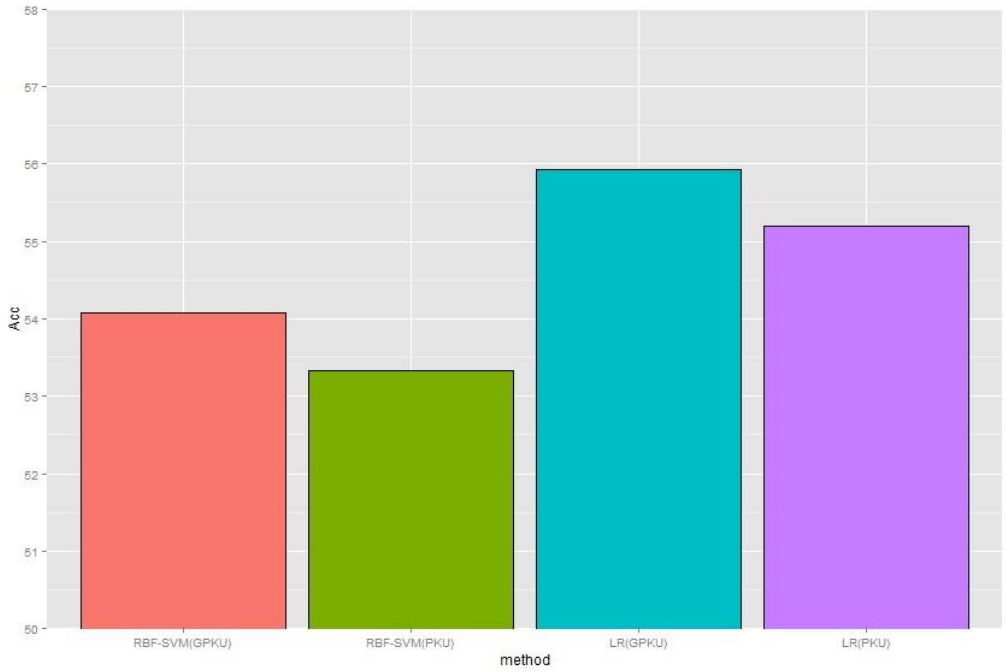


Slika 4.2: Zavisnost tačnosti klasi katora RBF-SVM od parametara σ i C

Višeklasna logistička regresija zasnovana na metodi gradijentnog spusta pokazala se dobro sa maksimalnom tačnošću od oko 56% na test setu, uz upotrebu GPKU za odabir atributa.

4.2 Rezultati i diskusija

Implementirana su i optimizovana četiri različita modela, dobijena odabirom jednog od opisanih pristupa odabiru atributa i jednog od dva navedena klasi katora. Tačnosti dobijene primenom svakog od njih na test setu uporedo su prikazane na Slici 4.3



Slika 4.3: Tačnosti dobijene primenom predloženih pristupa na test setu.

Svi prikazani modeli pokazali su tačnost izme u 53 i 56 procenata što predstavlja rezultat sličan modelima predstavljenim u literaturi. Pore enja radi, u eksperimentu koji je izveo sportski portal i sajt za kla enje Pinnaclesports [18] u sezoni 2012/13, fudbalski stručnjak na britanskoj televiziskoj mreži BBC, Mark Lorson, postigao je tačnost od 52.6% na svim utakmicama, dok su komercijalno primenjeni modeli ove kuće pokazali tačnost od 55.3%.

Iako je predloženim modelima postignuta zadovoljavajuća tačnost, pokazalo se da nijedan od njih nije imao mnogo uspeha u predviđanju nerešenih ishoda. Matrice konfuzije za RBF-SVM$_{GPKU}$ i LR$_{GPKU}$ koje ovo ilustruju date su u Tabeli 4.1.

RBF-SVM GPKU		Predvi eni		
		A	D	H
Stvarni	A	30	1	59
	D	12	0	41
	H	10	1	116

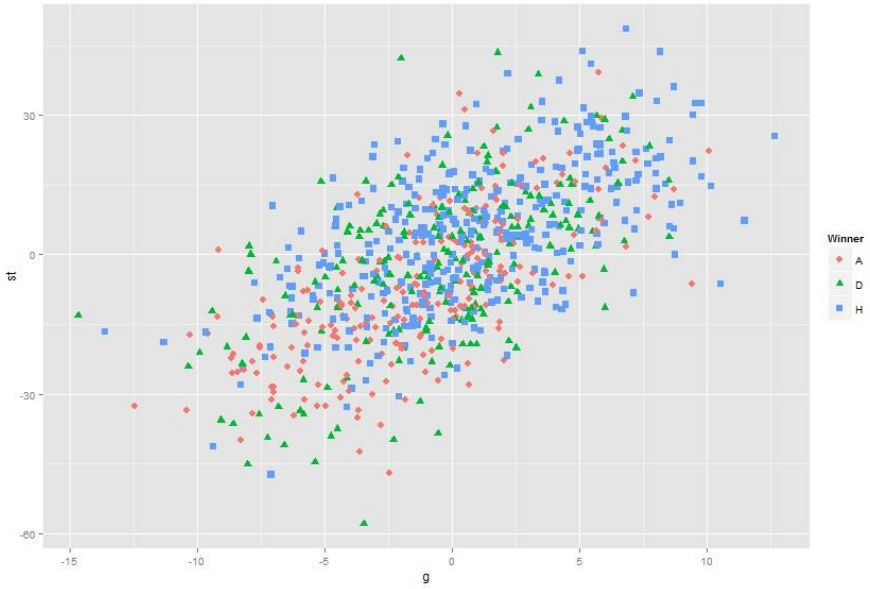
LR GPKU		Predvi eni		
		A	D	H
Stvarni	A	35	7	48
	D	15	2	36
	H	11	2	114

Tabela 4.1: Matrice konfuzije za RBF-SVM$_{GPKU}$ i LR$_{GPKU}$

Pokazalo se da, u slučaju klasi katora zasnovanih na metodi podržavajućih vektora, podešavanje parametara u cilju dobijanja maksimalne tačnosti dovodi do veoma retkog predviđanja nerešenog ishoda, dok se kod onih zasnovanih na logističkoj regresiji to uvek dešava. Ovakvo ponašanje modela je posledica relativno re eg pojavljivanja ovog ishoda, kao i lošije korelisanosti sa vektorima atributa. Na Slici 4.4 predstavljen je raspored ishoda u zavisnosti od atributa g i st. Ishodi u kojima pobe uje domaćin grupisani su u prvom kvadrantu i njegovoj neposrednoj okolini, dok se pobede gosta pretežno nalaze u trećem.

Me utim, nerešeni ishodi skoro su potpuno ravnomerno raspore eni po celom prostoru atributa.

Verovatnoće dodeljene nerešenom ishodu su često velike, ali nedovoljno da on bude izabran za konačno predviđanje. Podešavanje parametara u cilju poboljšanja predviđanja nerešenog ishoda drastično umanjuje tačnost pri predviđanju pobeda i poraza.



Slika 4.4: Zavisnost ishoda utakmice od atributa g i st u trening setu.

Zaključak

U ovom radu analizirana je primena poznatih metoda klasi kacije u predviđanju konačnog ishoda utakmica Premijer lige Engleske. Dati su pregledno osnovni pojmovi i koncepti vezani za klasi kacioni problem. Predstavljeni su poznati klasi kacioni algoritmi koji se smatraju teorijski značajnim, ili su ustaljeni u praktičnoj primeni.

Predložena su četiri klasi kaciona modela, koji rešavaju problem predviđanja ishoda utakmica, a zasnovani su na metodima logističke regresije i podržavajućih vektora. Svi predloženi modeli pokazali su tačnost u opesgu 53-56%, što predstavlja nešto bolji rezultat u odnosu na modele predstavljene u literaturi. Ovo poboljšanje najverovatnije je posledica boljeg metoda za formiranje vektora atributa.

Najveći izazov u dizajniranju predloženih modela predstavljala je skoro uniformna raspodela ishoda utakmica u korišćenim podacima, značajan broj nerešenih ishoda koje je teško predvideti, kao i česta pojava iznena enja u rezultatima. Nijedan od modela nije se dobro pokazao u predviđanju nerešenog ishoda, što je pripisano prirodi problema i korelisanosti korišćenih podataka.

Literatura

[1] Season review 2013-14, http://www.premierleague.com/, 2014

[2] K. Bache and M. Lichman, UCI Machine Learning Repository, 2013

[3] D. Michie, D. J. Spiegelhalter and C. Taylor, Machine Learning, Neural and Statistical Classi cation, 1994

[4] C. M. Bishop and N. M. Nasrabadi, Pattern recognition and machine learning, 2006

[5] R. O. Duda, P. E. Hart and D. G. Stork, Pattern classi cation, 2012 [6] D. Wettschereck, D. w. Aha and T. Mohri, A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. Arti cal Intelligence Review 11(1-5), 1997

[7] H. Brighton and C. Mellish, Advances in instance selection for instancebased learning algorithms. Data mining and knowledge discovery 6(2), 2002

[8] V. G. J. S. R. Mollineda and R. A. J. Sotonca, The class imbalance problem in pattern classi cation and learning, 2007

[9] V. Vapnik,The Nature of Statistical Learning Theory, Springer-Verlag, 1995

[10] V. Vapnik, An overview of statistical learning theory, Neural Networks, IEEE Transactions on 10 (5), 1999

[11] C. Cortes and V. Vapnik, Support-vector networks, Machine Learning 20 (3), 1995

[12] B.E Boser, I. M. Guyon and V. Vapnik, A training algorithm for optimal margin classi ers, 1992

[13] T. Mitchell, Machine learning, 1997

[14] England Football Results Betting Odds, http://footballdata.co.uk/englandm.php, 2015

[15] A. S. Timmaraju, A. Palnitkar, V. Khanna, Game ON! Predicting English Premier League Match Outcomes, 2013

[16] A. Joseph, A. E. Fenton and M. Neil, Predicting football results using Bayesian nets and other machine learning techniques, Knowledge-Based Systems 19(7), 2006

[17] B. Ulmer and M. Fernandez, Predicting Soccer Match Results in the English Premier League, 2014

[18] Mark Lawrenson vs. Pinnacle Sports,http://www.pinnaclesports.com/en/bettingarticles/sport/mark-lawrenson-vs-pinnacle-sports,2015