SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 44

ANALIZA I PREDIKTIVNO MODELIRANJE TENISKIH MEČEVA

Zvonimir Petar Rezo

Zagreb, lipanj 2021.

Sadržaj

Uvod 1

1. Osnove tenisa 2

2. Statistička analiza podataka 3

2.1. Eksploratorna statistička analiza 3

2.2. Deskriptivna statistička analiza 3

2.2.1. Mjere centralne tendencije 4

2.2.2. Mjere rasipanja 4

2.3. Matrica zabune 5

3. Statistički modeli 7

3.1. Markovljev model 7

4. Strojno učenje 9

4.1. Logistička regresija 10

4.2. K-najbližih susjeda 11

4.3. Stroj potpornih vektora 11

4.4. Naivni Bayesov klasifikator 13

4.5. Umjetna neuronska mreža 14

5. Obrada podatkovnog skupa 16

5.1. Podatkovni skup 16

5.2. Predobrada podataka 16

5.2.1. Psihološki moment 17

5.2.2. Model zajedničkih protivnika 18

5.3. Ulazne varijable 20

6. Rezultati 24

6.1. Model Markovljevih lanaca 24

6.1.1. Rezultati 25

6.2. Modeli strojnog učenja 26
6.2.1. Logistička regresija 27
6.2.2. K-najbližih susjeda 28
6.2.3. Stroj potpornih vektora 30
6.2.4. Naivni Bayesov klasifikator 31
6.2.5. Umjetna neuronska mreža 32
6.3. Usporedba rezultata 34
Zaključak 35
Literatura 36
Sažetak 37
Summary 38

Uvod

Predviđanje ishoda sportskih događaja oduvijek je privlačilo pažnju velikog broja ljudi. Tenis, jedan od popularnijih sportova kojeg uživaju milijuni gledatelja tijekom cijele godine osobito je atraktivan u znanstvenim istraživanjima. To je sport sa strogo definiranom strukturom i rigidnim sustavom bodovanja i kao takav se jednostavno može opisati skupom stanja i prijelaza između tih stanja. Za modeliranje takvog sustava najčešće se koriste Markovljevi lanci prvog reda, a vrlo važne parametre predstavljaju statistike ranijih susreta promatranih igrača. Zadatak rada je analiza teniskih mečeva i izrada prediktivnog modela koji na ulazu dobiva više parametara opisa postotka uspješnosti svakog od tenisača, a na izlazu daje vjerojatnosti različitih ishoda teniskog meča. Cilj rada je provesti eksploratornu analizu dostupnih podataka uz naglasak na primjenu algoritama strojnog učenja na predviđanje pobjednika teniskog meča.

U prvom poglavlju bit će objašnjena pravila i tijek teniske igre. Drugo poglavlje sadrži nužna znanja o statističkoj analizi podataka korištena u ostatku rada. U trećem i četvrtom poglavlju objašnjeni su statistički modeli i modeli strojnog učenja korišteni u samim predviđanjima. Peto poglavlje bavi se uvodom u korišteni podatkovni skup i kratkim objašnjenjem procesa predobrade podataka, dok su u šestom poglavlju izneseni rezultati rada.

1. Osnove tenisa

Tenis je sport u kojem sudjeluju dva (pojedinačno) ili četiri (parovi) igrača, a igra se na obilježenom igralištu koristeći reket i lopticu. Osnovni cilj igre je reketom plasirati lopticu preko mreže u protivnikovo polje tako da ju protivnik ne uspije na ispravan način vratiti.

Bodovanje se u tenisu dijeli na poene (engl. *point*), gemove (engl. game) i setove. Meč se dijeli na setove, setovi se dijele na gemove, a gemovi se dijele na poene. Svaki poen u tenisu započinje servisom, jedan igrač servira a drugi prima servis. Nakon što igrač koji servira na ispravan način ubaci lopticu u protivnikovo polje, igra se nastavlja tako što oba igrača prebacuju mrežu lopticom sve dok jedan od njih ne pogriješi, što rezultira osvajanjem poena drugog igrača. Prvi igrač koji osvoji barem četiri poena ili za dva više od protivnika osvaja gem. Nakon svakog odigranog gema kreće servirati onaj igrač koji je u prošlom gemu primao servis. Prvi igrač koji osvoji barem šest gemova ili za dva više od protivnika osvaja set. Poseban slučaj je ako rezultat dođe do 6-6. Tada se igra tzv. tiebreak u kojem prvi igrač koji osvoji barem sedam poena ili za dva više od protivnika osvaja gem, a time i set. Na većini turnira pobjednik je onaj igrač koji prvi osvoji dva seta (engl. *best of 3*), ali na najvećim turnirima u muškoj se konkurenciji igra na tri dobivena seta (engl. *best of 5*).

2. Statistička analiza podataka

Statistika je znanstvena disciplina koja se bavi prikupljanjem, opisivanjem, analizom i interpretacijom podataka[1]. Statistička analiza podataka je proces odrađivanja statističkih operacija u svrhu dobivanja znanja iz podataka.

2.1. Eksploratorna statistička analiza

Eksploratorna statistička analiza bavi se istraživanjem skupa podataka u svrhu izlučivanja karakteristika te pronalaženja pravila i anomalija. Često se u tu svrhu koriste razni načini vizualizacije podataka. Ova statistička analiza nije skup tehnika, nego pristup analizi podataka. Korištenjem eksploratorne statističke analize dobiva se bolji uvid u podatke i odnose među podacima kako bi se daljnjim analizama lakše došlo do statistički bitnih rezultata

2.2. Deskriptivna statistička analiza

Deskriptivna statistička analiza uglavnom se bavi mjerama centralne tendencije i mjerama rasipanja. Neke od mjera centralne tendencije su aritmetička sredina, medijan, mod, geometrijska sredina i harmonijska sredina. Neke od najčešće korištenih mjera centralne tendencije su aritmetička sredina, medijan i mod. Položajne mjere ili mjere lokacije širi su pojam od mjera centralne tendencije. Jedna od najvažnijih položajnih mjera je percentil. Neke od bitnih mjera rasipanja su rang, varijanca, standardna devijacija, koeficijent varijacije i interkvartilni rang. U sljedećim potpoglavljima opisane su mjere centralne tendencije i mjere rasipanja korištene u ovom radu.

[1] Preuzeto s https://www.fer.unizg.hr/predmet/sap

2.2.1. Mjere centralne tendencije

Aritmetička sredina je jedna od najintuitivnijih i najosnovnijih mjera centralne tendencije. Računa se po formuli:

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n} \quad (2.1)$$

Aritmetička sredina provodi se na uzorku populacije jer su slučajevi u kojima imamo sve podatke populacije jako rijetki. Problem kod aritmetičke sredine je osjetljivost na ekstreme pri korištenju malih uzoraka.

Medijan je mjera centralne tendencije koja poprima vrijednost srednjeg podatka u skupu poredanom po veličinama. Računa se po formuli:

$$M = \begin{cases} x_{\frac{(n+1)}{2}} & za\ neparan\ n \\ \frac{x_{n/2} + x_{n/2+1}}{2} & za\ paran\ n \end{cases} \quad (2.2)$$

Prednost medijana u odnosu na aritmetičku sredinu je neosjetljivost na ekstreme.

2.2.2. Mjere rasipanja

Prva mjera rasipanja koju ćemo spomenuti bit će varijanca. Varijanca predstavlja srednje kvadratno odstupanje od aritmetičke sredine i se računa prema formuli:

$$\sigma^2 = \frac{1}{N} \sum_{i=0}^{N} (\mu - x_i)^2 \quad (2.3)$$

N – veličina populacije

μ – aritmetička sredina populacije

x_i – i-ti podatak iz populacije

Kao što je ranije navedeno, aritmetička sredina rijetko je poznata te se u većini slučajeva računa iz uzorka populacije. Iz tako dobivene aritmetičke sredine varijanca se dobiva formulom:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=0}^{N} (\bar{x} - x_i)^2 \quad (2.4)$$

gdje je $\bar{x}$ aritmetička sredina uzorka.

Standardna devijacija je mjera centralne tendencije koja se dobiva izvođenjem operacije drugog korijena na varijanci.

2.3. Matrica zabune

Za prikaz rezultata kod problema klasifikacije u strojnom učenju često se koristi matrica zabune. To je matrica koja na poprilično jednostavan način omogućuje vizualizaciju rezultata modela strojnog učenja. U tablici (**Tablica 2.1**) dan je primjer matrice zabune za binarni klasifikator (samo dvije klase, 1 i 0). Matrice zabune koriste se i kod problema sa više klasa, ali mi ćemo se baviti matricom s dvije klase s obzirom da nam je takva potrebna za naš problem klasifikacije. Iz prikazane tablice vidimo da redci predstavljaju stvarne vrijednosti, a stupci predviđanja modela za iste te podatke. Iz prikazane matrice čitamo da je klasifikator ukupno napravio 190 predviđanja. Klasa 1 kao stvarna vrijednost pojavila se 115 puta, a klasa 0 kao stvarna vrijednost pojavila se 75 puta. Vrijednost 1 bila je rezultat klasifikatora 125 puta, a vrijednost 0 bila je rezultat 65 puta. Osnovni izrazi korišteni kod analize matrica zabune su: istinski pozitiv (IP) - i stvarna i predviđena vrijednost su 1, istinski negativ (IN) - i stvarna i predviđena vrijednost su 0, lažni pozitiv (LP) - predviđena vrijednost je 1, ali stvarna vrijednost je 0, lažni negativ (LN) - predviđena vrijednost je 0, ali stvarna vrijednost je 1. Sada kada smo upoznati sa osnovnim izrazima, možemo se upoznati sa svojstvima klasifikatora koja računamo na osnovu tih izraza.

Preciznost (engl. *accuracy*) je svojstvo klasifikatora koje se, kao i ostala svojstva kojima ćemo se baviti, vrlo lako iščitava iz matrice konfuzije. Preciznost se računa kao broj pogođenih klasifikacija podijeljen s ukupnim brojem klasificiranih podataka: $\frac{IP+IN}{Ukupno}$

Osjetljivost (engl. *sensitivity*) računa se kao broj pogođenih pozitiva (klasa 1) podijeljen s ukupnim brojem pozitiva: $\frac{IP}{Ukupno\ pozitiva}$

Specifičnost (engl. *specificity*) računa se kao broj pogođenih negativa (klasa 0) podijeljen s ukupnim brojem negativa: $\frac{IN}{Ukupno\ negativa}$

Osjetljivosti i specifičnosti doticat ćemo se u manjoj mjeri pri prikazivanju rezultata. Razlog tome je što u našem slučaju ne možemo puno doznati iz tih svojstava s obzirom da kod klasifikacije imamo dvije klase, pobjedu jednog i pobjedu drugog igrača, a jedini razlog

zašto je pobjeda jednog igrača označena klasom 1, a pobjeda drugog klasom 0 je raspored igrača u tablici iz koje čitamo podatke koji se bez gubitka informacije može i obrnuti (1 u 0 i 0 u 1).

Stopa neinformiranosti (engl. *No Information Rate*) *najbolji je mogući rezultat predviđanja uz nedostatak podataka osim razdiobe klasa koje pokušavamo predvidjeti*[2]. Iz primjera matrice dane u tablici (**Tablica 2.1**) vidimo da većina (60,5%) jedinki iz populacije dolazi iz klase 1, što znači da, kad bismo u svakom predviđanju odabrali većinsku klasu (klasu 1), bili bismo u pravu 60,5% vremena. Stopa neinformiranosti je upravo ta vrijednost koja označava udio većinske klase (u ovom slučaju 60,5%).

Tablica 2.1: Primjer matrice konfuzije

	Predviđeno 1	Predviđeno 0
Stvarna vrijednost 1	100	15
Stvarna vrijednost 0	25	50

Uzimajući u obzir stopu neinformiranosti, moguće je s mnogo većom sigurnošću odrediti je li preciznost modela iskazana u matrici zabune zadovoljavajuća. U ovu svrhu, pri analizi rezultata modela strojnog učenja, promatrat ćemo stopu neinformiranosti te provoditi statističke testove da bismo provjerili je li preciznost modela značajno bolja od stope neinformiranosti.

[2] Preuzeto s https://www.hranalytics101.com/how-to-assess-model-accuracy-the-basics

3. Statistički modeli

3.1. Markovljev model

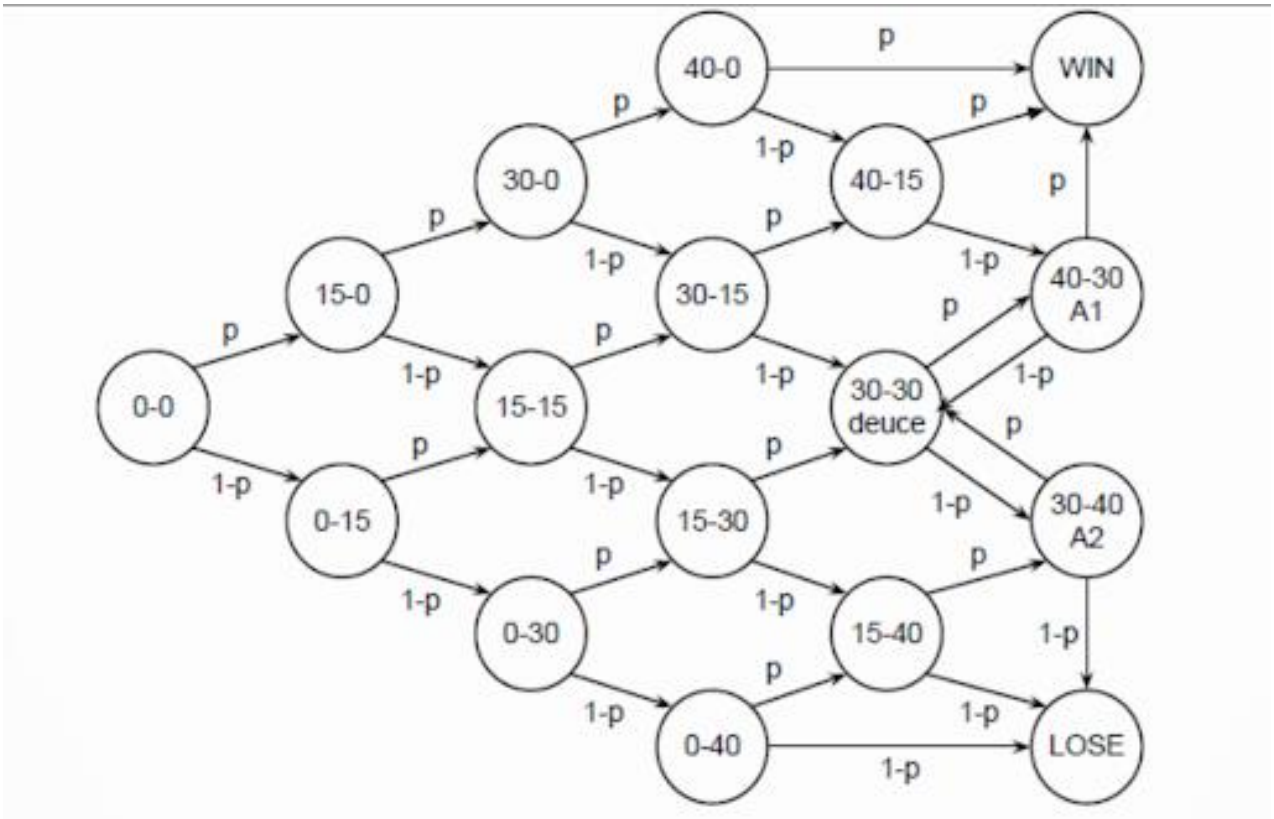
Markovljev model je model zasnovan na Markovljevim lancima. Markovljev lanac predstavlja niz stanja sustava i prijelaza između tih stanja. Za slijed stanja kaže se da ima Markovljevo svojstvo ako je svako buduće stanje vremenski neovisno u svakom prijašnjem stanju. Formalna definicija Markovljevog lanca kaže:

Markovljev lanac je slijed slučajnih varijabli X_1, X_2, X_3, ... s Markovljevim svojstvom i to zato što su trenutno, buduće i prošlo stanje nezavisni.[3]

Često je predmet analize i rasprave kod modeliranja teniskih mečeva Markovljevim lancima upravo tzv. *nedostatak pamćenja*. Nedostatak pamćenja u Markovljevim lancima znači da sljedeće stanje ovisi isključivo o trenutnom stanju, time zanemarujući način na koji se došlo do trenutnog stanja. Samim time, u većini modela koji koriste Markovljeve lance pri modeliranju teniskog meča koristi se isključivo vjerojatnost osvajanja poena na servisu igrača koji trenutno servira. U nekim modelima kao što je onaj kojega su opisali Barnett i Clarke[1] koristi se više varijabli od same vjerojatnosti osvajanja poena na servisu. Konkretno u navedenom radu dane su formule kojima se na temelju parametara kao što su vjerojatnost osvajanja poena na prvom i drugom servisu zasebno, vjerojatnost osvajanja poena na primanju prvog i drugog servisa protivnika, prosječnog postotka osvajanja poena na servisu svih igrača itd. računa vjerojatnost osvajanja poena igrača te se ta vjerojatnost dalje koristi u Markovljevom modelu.

Markovljevi lanci često se prikazuju direktnim grafom gdje su bridovi označeni vjerojatnošću koja predstavlja prelazak iz jednog stanja u drugo. Na slici (**Slika 3.1**) prikazan je Markovljev lanac za jedan teniski gem. Vjerojatnost p označava vjerojatnost osvajanja poena za igrača koji servira, a 1-p vjerojatnost da igrač koji servira izgubi poen, odnosno da ga njegov protivnik dobije.

[3] Preuzeto s https://en.wikipedia.org/wiki/Markov_chain



Slika 3.1 Markovljev lanac za teniski gem[4]

U ovom radu korišteni su Markovljevi lanci koji na ulazu dobivaju postotak osvojenih poena na servisu dva igrača te na osnovu toga računaju vjerojatnost pobjede za oba igrača.

[4] Preuzeto sa http://deeconometrist.nl/predicting-outcomes-of-tennis-matches

4. Strojno učenje

Strojno učenje je grana umjetne inteligencije koja se bavi izradom algoritama koji uče kroz iskustvo iz nekog podatkovnog skupa. Definicija strojnog učenja koju je 2010. dao Alpaydin glasi: *Strojno učenje jest programiranje računala na način da optimiziraju neki kriterij uspješnosti temeljem podatkovnih primjera ili prethodnog iskustva. Raspolažemo modelom koji je definiran do na neke parametre, a učenje se svodi na izvođenje algoritma koji optimizira parametre modela na temelju podataka ili prethodnog iskustva*. Najjednostavnije rečeno, algoritmi strojnog učenja predviđaju neke nepoznate podatke na osnovu viđenih podataka. Cilj strojnog učenja je izgraditi modele koji na osnovu dobivenih podataka dobro generaliziraju problem.[5]

Strojno učenje koristi se na mnogim poljima i, s obzirom na razvoj tehnologije, postalo je neophodno u današnjem svijetu. Neke od najčešćih općenitih primjena strojnog učenja su problemi koji su presloženi da bi ih se riješilo algoritamski, sustavi koji se dinamički mijenjaju te sustavi sa velikim količinama podataka iz kojih je teško izvući korisna znanja.

Vrste strojnog učenja su: nadzirano učenje, nenadzirano učenje i podržano učenje. Nadzirano učenje svodi se na traženje funkcije koja preslikava ulazne vrijednosti u izlaznu vrijednost. Ako je izlazna vrijednost diskretna radi se o klasifikaciji, a ako je kontinuirana radi se o regresiji. Kod nenadziranog učenja na ulazu su dani podaci bez ciljne vrijednosti te je potrebno pronaći pravilnosti i nepravilnosti u podatcima. Podržano učenje bavi se pitanjem kako sustav naučiti optimalnoj strategiji kako bi maksimizirao kumulativnu nagradu u okruženju u koje je postavljen.

U strojnom učenju postoje dva glavna problema na koje treba paziti pri izradi modela: prenaučenost i podnaučenost. Prenaučenost se dešava onda kada se model previše prilagodi podacima koje je dobio u skupu za učenje, a onda jako slabo ili gotovo nikako ne funkcionira na novim podacima. Podnaučenost je posljedica prevelike jednostavnosti modela koji nije u stanju shvatiti i prihvatiti osnovne karakteristike i odnose među podacima.

U sklopu ovog rada korišteni su mnogi algoritmi strojnog učenja za predviđanje ishoda teniskih mečeva. Također, za svaki algoritam su isprobane razne kombinacije ulaznih

[5] Preuzeto s https://www.fer.unizg.hr/predmet/su

varijabli o kojima će biti više riječi u poglavlju 5 kada ćemo se baviti predobradom podataka i dobivanjem ulaznih varijabli.

4.1. Logistička regresija

Logistička regresija je nadzirani klasifikacijski algoritam. Iako se većinom koristi samo za klasifikaciju, logistička regresija zapravo radi tako da regresijskim modelom predviđa vjerojatnost da dani podatak pripada kategoriji s brojem 1. Logistička regresija postaje klasifikacijska metoda onda kada se na rezultate doda prag odluke. Ako je dobivena vrijednost veća od praga odluke, ona spada u kategoriju s brojem 1, a ako je manja, spada u kategoriju 0. Odabir praga odluke je jedan od ključnih koraka pri korištenju ove metode. Logistička regresija modelira podatke na osnovu funkcije:

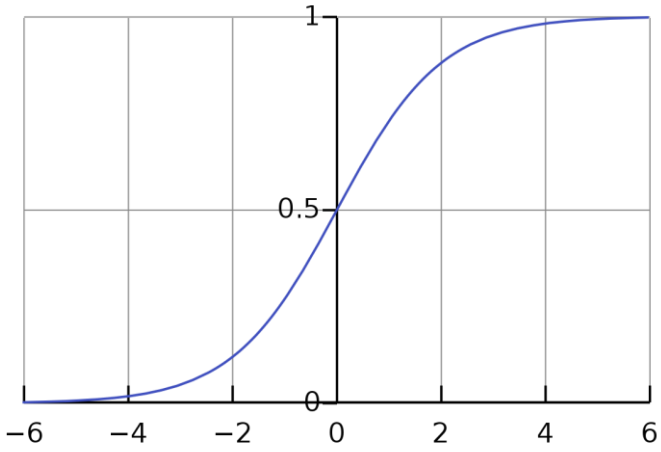
$$f(z) = \frac{1}{1+e^{-z}} \tag{4.1}$$

Z dobivamo na sljedeći način:

$$z = \beta_0 + \sum_{n=1}^{N}(\beta_i x_i) \tag{4.2}$$

x_i – vrijednosti pojedinih varijabli

β_i – vrijednosti koeficijenata koje logistička regresija određuje varijablama



Slika 4.1 Graf logističke funkcije[6]

[6] Preuzeto sa https://en.wikipedia.org/wiki/Logistic_function

4.2. K-najbližih susjeda

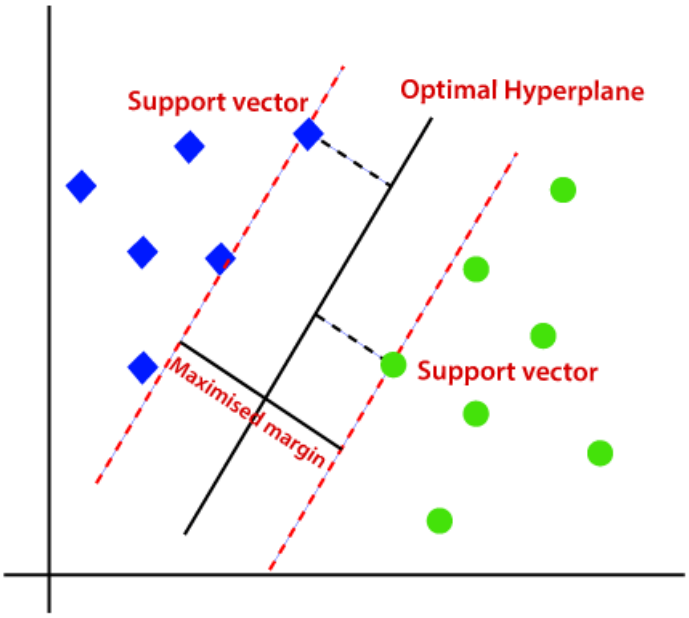
K-najbližih susjeda je nadzirani algoritam strojnog učenja koji se koristi i za klasifikacijske i za regresijske probleme. Kod klasifikacije, ideja ove metode je da se novi podatak klasificira tako da se promatraju njemu najbliži podaci iz skupa za učenje. Broj k, odnosno broj najbližih susjeda koje metoda uzima u obzir vrlo je bitan za učinkovitost modela, ali nema egzaktnog načina za odabir optimalne vrijednosti. Za izračun udaljenosti najčešće se koristi Euklidova udaljenost koja je za n-dimenzionalan sustav dana sa:

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2} \qquad (4.3)$$

Zanimljivo je kod ove metode strojnog učenja što se za vrijeme faze učenja zapravo ništa ne „uči“, već se samo podaci spremaju (mapiraju) te se onda novi podaci klasificiraju na osnovu tih podataka iz skupa za učenje. Problem kod obične klasifikacije na osnovu metode K-najbližih susjeda se događa kada je distribucija asimetrična. U tom slučaju će klasa koja prevladava (kojoj pripada više podataka iz skupa) zadobiti nove podatke na osnovu toga što jednostavno podataka iz te klase ima puno više pa će među k najbližih susjeda gotovo uvijek biti više pripadnika te klase. Postoji nekoliko načina za rješavanje ovoga problema. Najčešće rješenje je dodavanje težine (važnosti) podacima na osnovu udaljenosti od novog podatka. Na taj način se daje šansa klasi koja je u manjini da prevlada iako nije najbrojnija u skupu najbližih susjeda.

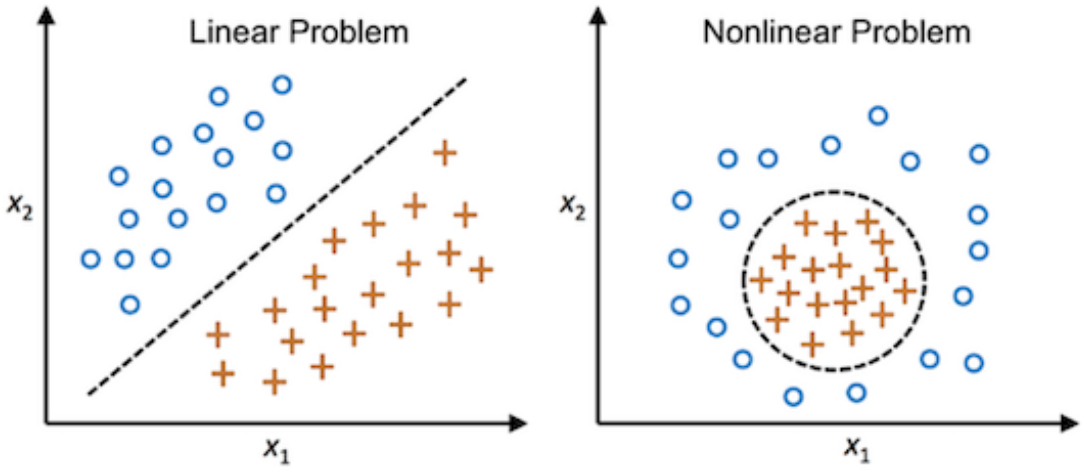
4.3. Stroj potpornih vektora

Stroj potpornih vektora je nadzirani algoritam strojnog učenja koji se koristi u svrhu regresije i klasifikacije. Stroj potpornih vektora konstruira hiperravninu ili skup hiperravnina u visokodimenzionalnom prostoru. Metoda radi na način da traži idealnu hiperravninu za podjelu podataka. Hiperravnina se određuje tako da se maksimizira margina oko hiperravnine odnosno udaljenost od najbližeg podatka svake klase. Vektori koji se koriste za određivanje hiperravnine nazivaju se potporni vektori.



Slika 4.2 Linearni model stroja potpornih vektora[7]

U mnogim problemima linearna klasifikacija primjenom stroja potpornih vektora nije primjenjiva bez preslikavanja podataka u prostor viših dimenzija. Na desnoj strani slike (**Slika 4.3**) prikazan je najosnovniji primjer problema koji ne možemo učinkovito riješiti linearnim modelom. Ovakvi problemi rješavaju se mapiranjem podataka na višu dimenziju. Ipak, izračun koordinata podataka u prostoru visokih dimenzija može biti iznimno računski zahtjevan. Da bi se to izbjeglo, koriste se takozvane kernel funkcije koje omogućuju izračun potpornih vektora bez eksplicitnog izračuna koordinata u visokim dimenzijama.



Slika 4.3 Nelinearni model stroja potpornih vektora[8]

[7] Preuzeto s https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm

[8] Preuzeto s https://www.analyticsvidhya.com/blog/2021/05/support-vector-machines/

4.4. Naivni Bayesov klasifikator

Naivni Bayesov klasifikator je model strojnog učenja koji je baziran na Bayesovom teoremu koji glasi:

$$P(B|A) = \frac{P(A|B)P\,(B)}{P\,(A)} \quad (4.4)$$

P(A) – vjerojatnost događaja A

P(B) – vjerojatnost događaja B

P(A|B) – vjerojatnost događaja A uz uvjet da se dogodio događaj B

P(B|A) – vjerojatnost događaja B uz uvjet da se dogodio događaj A

Glavna karakteristika i važnost Bayesovog teorema (4.4) je mogućnost određivanja vjerojatnosti jednog događaja na osnovu drugih događaja. Riječ „naivni“ u imenu ovog modela je tu iz razloga što model koristi snažnu (naivnu) pretpostavku o nezavisnosti značajki skupa podataka, a ta pretpostavka općenito ne vrijedi. Bayesov klasifikator tako izravno koristi Bayesov teorem kako bi izračunao vjerojatnost da ulazni podatak *x* pripada klasi *y*:

$$P(y|x) = \frac{\mathrm{P}\,(x,y)}{\mathrm{P}\,(x)} = \frac{\mathrm{P}(x|y)P\,(y)}{\mathrm{P}\,(x)} \quad (4.5)$$

Vrijednost P(y|x) je upravo ono što se traži, vjerojatnost pripadnosti klasi *y* podatka *x*. Ova vjerojatnost se još naziva i aposteriorna vjerojatnost oznake**.** Sada se formulom 4.5 mogu računati aposteriorne vjerojatnosti pripadnosti svakoj klasi koju problem sadrži, a onda se od tih vjerojatnost izabire ona najveća i podatak se klasificira u klasu čija je aposteriorna vjerojatnost najveća (h_{MAP} – maksimum aposteriori hipoteza)[9].

$$\mathrm{h_{MAP}} = \underset{\mathrm{y}}{\mathrm{argmax}}\,\mathrm{P(x|y)P\,(y)} \quad (4.6)$$

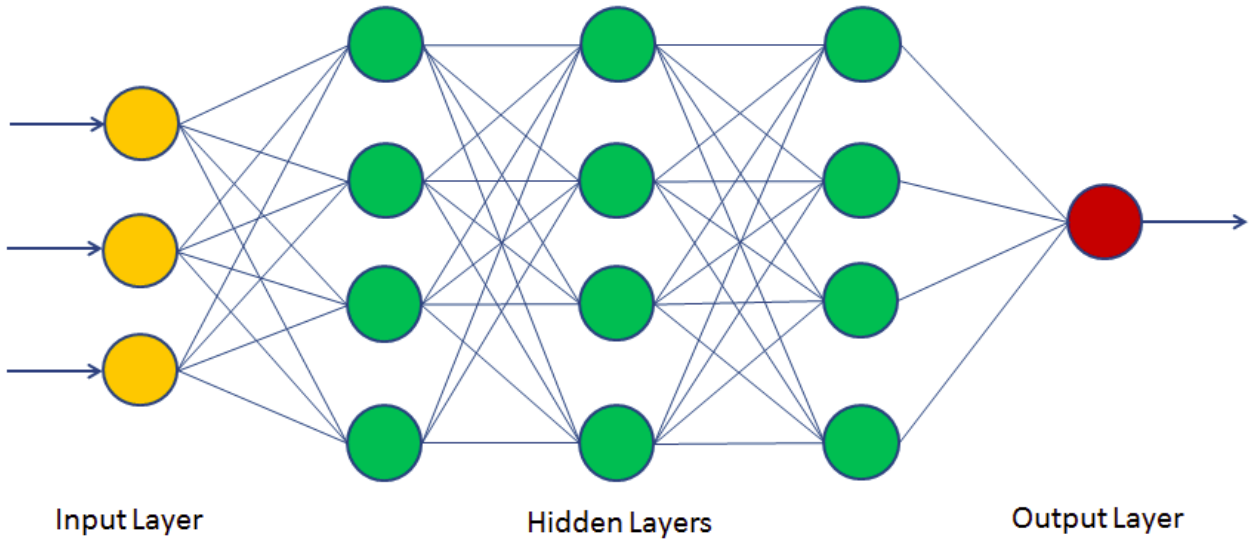
U većini slučajeva nije potrebno računati same vjerojatnosti, već je dovoljno dobiti samo pripadnost klasi, zbog toga je moguće ukloniti P(x) iz računa (4.5) jer je ta vrijednost uvijek ista.

[9] Preuzeto s https://www.fer.unizg.hr/predmet/su

4.5. Umjetna neuronska mreža

Umjetna neuronska mreža je model zasnovan na biološkim neuronima. To je sustav međusobno povezanih računalnih „neurona“ koji simulira način na koji ljudski mozak procesuira informacije. Višeslojne neuronske mreže sastoje se od ulaznog sloja, izlaznog sloja te jednog ili više skrivenih slojeva neurona koji se nalaze između ulaznog i izlaznog sloja. Svaki neuron ima svoje ulaze na osnovu kojih računa izlaz. Izlazi neurona iz *n*-tog sloja predstavljaju ulaze u neurone (*n+1*)-og sloja.

S obzirom na povezanost neuronske mreže dijele se na potpuno povezane i djelomično povezane. Potpuno povezane neuronske mreže su one u kojima je svaki neurom u svakom sloju povezan na sve neurone u sljedećem sloju. Ako neke od tih veza nisu pristupne onda se govori o djelomično povezanoj neuronskoj mreži[10]. Na slici (**Slika 4.4**) prikazana je jednostavna potpuno povezana neuronska mreža s tri skrivena sloja neurona.



Slika 4.4 Prikaz neuronske mreže[11]

Broj značajki ulaznog sloja jednak je broju odabranih značajki podatkovnog skupa za koji se gradi neuronska mreža.

[10] Preuzeto s https://www.fer.unizg.hr/predmet/neumre_b
[11] Preuzeto s https://www.pinterest.com

Svakoj vezi između dva neurona pridijeljena je težina. Neuron tako koristi dobivenu vrijednost i težinu veze za izračun izlazne vrijednosti po formuli:

$$f(x) = k\left(\sum_i w_i x_i\right) \quad (4.7)$$

w_i – težina *i*-te veze

x_i – vrijednost *i*-tog ulaza

k – aktivacijska funkcija

Aktivacijska funkcija je funkcija korištena u umjetnim neuronskim mrežama koja kao izlaz daje malu vrijednost ako je na ulazu mala vrijednost, a daje veću vrijednost ako je ulaz veći od granice (engl. *threshold*). Drugim riječima, *aktivacijska funkcija je kao senzor koji provjerava je li ulazna vrijednost veća od nekog kritičnog broja*[12]. Najčešće korištene aktivacijske funkcije su ReLU (engl. *rectified linear unit*) te sigmoidne funkcije kao što su logistička sigmoidna funkcija (4.1), tangens hiperbolni te arkus tangens.

Umjetne neuronske mreže mogu naučiti neke kompleksne veze među podatcima, ali su podložne prenaučenosti (engl. *overfitting*) i zbog toga kod većine problema ne daju pretjerano dobre rezultate ako im nije pružen velik skup podataka za učenje.

[12] Preuzeto s https://deepai.org/machine-learning-glossary-and-terms/activation-function

5. Obrada podatkovnog skupa

U praktičnom dijelu ovog rada obrađen je podatkovni skup u programskom jeziku R. U ovom poglavlju bavit ćemo se opisom podatkovnog skupa, predobradom podataka i intuicijom koja stoji iza korištenih tehnika u razvoju modela.

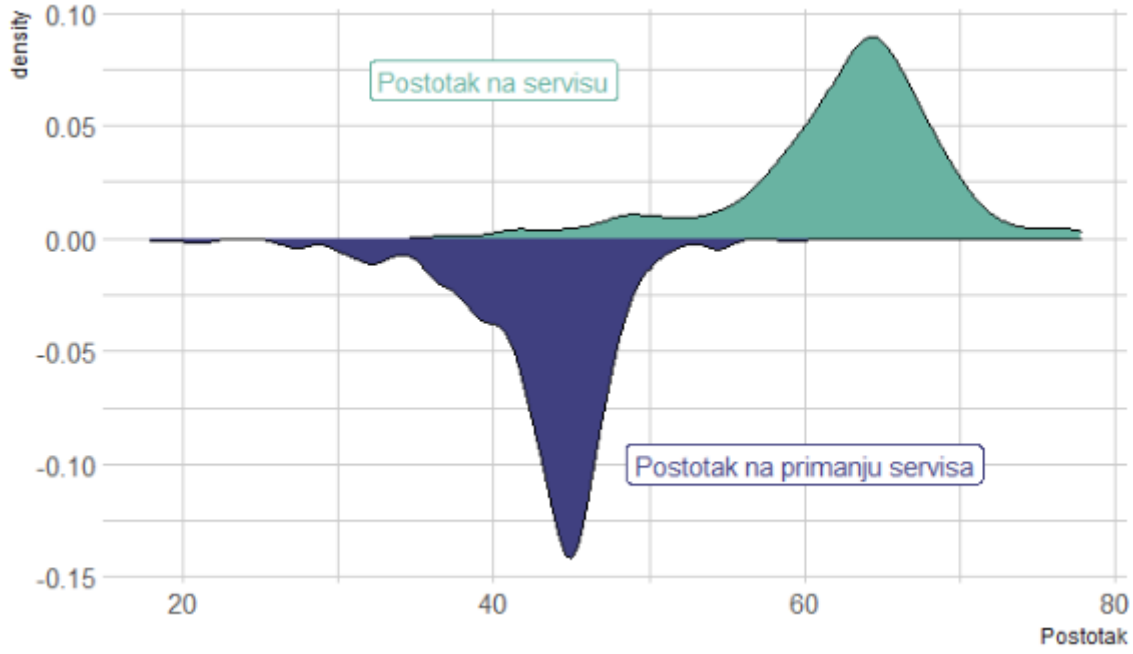
5.1. Podatkovni skup

Podatkovni skup preuzet je iz [9] i opisuje poen po poen tijek svakog teniskog meča odigranog na profesionalnoj razini u 2015. i 2016. godini. Skup je u *.csv* formatu (engl. *comma separated values*). U ovom radu promatrat ćemo samo mušku konkurenciju kako bi izbjegli nepotrebno ponavljanje sličnih rezultata za mušku i za žensku konkurenciju te kako bi se ipak više koncentracije usmjerilo na jedan podskup umjesto na dva. Skupovi dostupni na internetu uglavnom sadrže općenitije podatke kao što su ATP i WTA rang liste, broj bodova na tim rang listama, te broj osvojenih gemova i poena svakog igrača u tom meču. Prednost ovog podatkovnog skupa u odnosu na mnoge skupove dostupne na internetu je upravo mogućnost dublje analize po poenima te usporedbe poena kao takvih, odnosno razmatranje važnosti poena kao i utjecaja poena na ishod meča. Iako skup sadrži svaki teniski meč odigran na profesionalnoj razini, mi ćemo radi smanjenja raspršenosti podataka i povećanja zanimljivosti rada koristiti samo one odigrane na ATP razini.

5.2. Predobrada podataka

Predobrada podataka provodila se u više koraka i bila je velik dio praktičnog rada. Prvo je trebalo u radni okvir učitati sve dostupne tablice. Najosnovnije tablice, a ujedno i tablice s najviše podataka su *sportevent* tablice koje sadrže tijek svakog pojedinog teniskog meča po poenima. *Sportevent* tablice bilo je potrebno spojiti sa *competitions* tablicom kako bi se meč povezao s turnirom na kojem je odigran i samim tim omogućio određivanje podloge meča. Tablice je naravno trebalo očistiti jer su mnoge sadržavale duplicirane vrijednosti, a više takvih pogrešaka bi moglo uvelike utjecati na učinkovitost izgrađenih modela. Nakon ovih početnih koraka bilo je moguće započeti s eksploratornom analizom podatkovnog skupa. Iz očišćenog skupa sada je bilo moguće izvući mnoge zanimljive značajke koje su korištene kasnije pri izradi modela strojnog učenja. S obzirom na oblik podatkovnog skupa, dobivanje

većine značajki svodilo se na brojanje poena s obzirom na neki kriterij te usporedbu s ostalim podacima (igračima). Osnovne značajke izračunate na početku analize su postotci osvajanja poena na servisu i na primanju servisa ukupno te isti ti postotci grupirani s obzirom na podlogu na kojoj su mečevi odigrani. Na slici (**Slika 5.1**) prikazan je graf gustoće razdiobe za dvije osnovne značajke, postotak osvojenih poena igrača na servisu i postotak osvojenih poena igrača na primanju servisa.

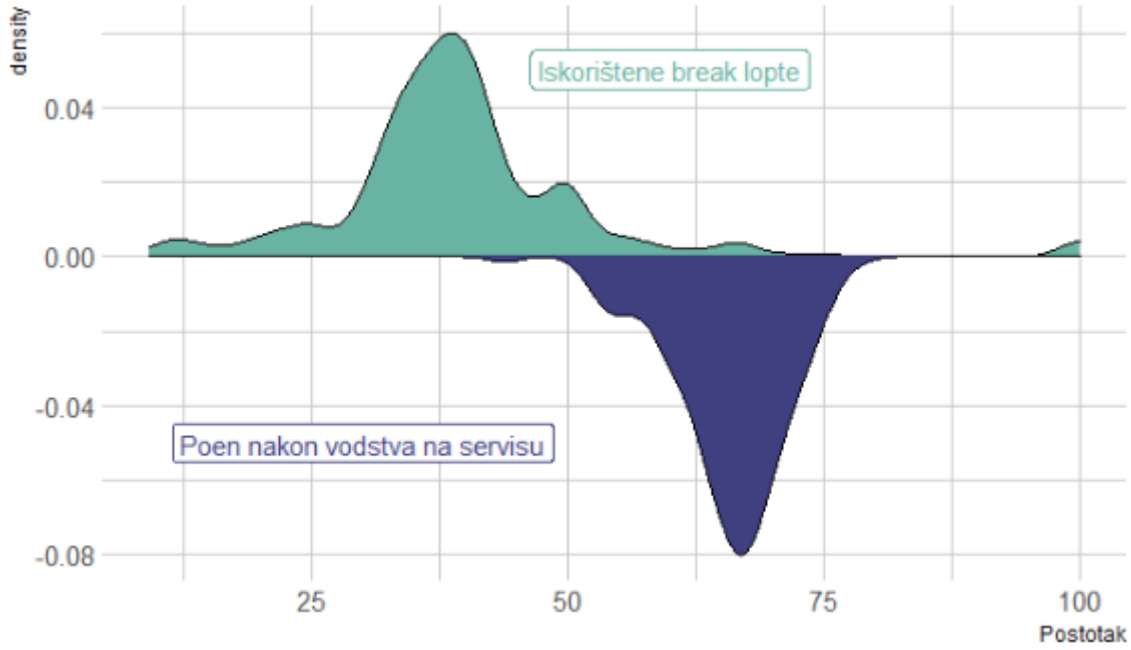


Slika 5.1: Graf gustoće razdiobe – servis i return

5.2.1. Psihološki moment

Neke od zanimljivijih značajki su one koje će biti ispitane u svrhu analize psihološkog momenta u tenisu. Psihološki moment ili psihološki zamah je događaj u sportu kada sportaš/i igra/ju iznad svojih mogućnosti. Većina ljudi koji prati sport vjeruje u postojanje psihološkog momenta i smatraju da je bitan faktor u većini sportova, ali u ozbiljnim znanstvenim analizama pokazuje se da je psihološki moment nije tako očit kao što se na prvu loptu čini. Znanstveni radovi na ovu temu su podijeljeni. Konkretno za tenis, Jackson i Mosurski[2] nisu pronašli konkretne dokaze da teški porazi u tenisu imaju veze sa psihološkim momentom, dok u [3] F. J. Klaasen i J. R. Magnus zaključuju da osvajanje prošlog poena ima pozitivan utjecaj na vjerojatnost osvajanja trenutnog poena te da je na bitnim poenima igrač koji servira u nepovoljnom položaju. Psihološki moment nije centralna

tema ovog rada ali spominjat će se kao zanimljivost i mogući presudni faktor u nekim rezultatima. Značajke promatrane i korištene za izradu modela koje se baziraju na pretpostavci psihološkog momenta su postotak iskorištenih break prilika te postotak osvajanja poena nakon vodstva na servisu. Analizirana je još i vjerojatnost osvajanja seta nakon osvajanja ili gubitka jednog seta, ali ove analize su rezultirale velikom varijancom pa ti rezultati nisu korišteni u daljnjem radu.



Slika 5.2: Graf gustoće razdiobe - psihološke značajke

5.2.2. Model zajedničkih protivnika

Model zajedničkih protivnika je strategija uvedena u mnogim radovima koji se bave tenisom kako bi se izbjegla pristranost podataka uzrokovana razlikom u kvaliteti protivnika s kojima su igrala dva igrača nad čijim mečem želimo izvršiti predviđanje. Nakon objave rada [4] koji je jedan od prvih značajnijih radova koji su uveli ovaj model, mnogi drugi istraživači ovog područja počeli su koristiti slične modele. Konkretno, navedeni rad pokazuje kako stohastički model baziran na Markovljevim lancima daje točnije rezultate kada se primjeni metoda zajedničkih protivnika. Također, [5] pokazuje da je ovaj model jednako primjenjiv i na slučajeve predikcije na bazi strojnog učenja.

Ideja modela zajedničkih protivnika je da se kao značajke za predviđanje uzimaju vrijednosti izračunate na skupu mečeva koji su odigrani sa zajedničkim protivnicima dva igrača. Na

primjer, najrigorozniji oblik ove strategije za izračun postotka osvojenog poena na servisu Novaka Đokovića kada igra s Rafaelom Nadalom neće uzimati u obzir Đokovićev meč s igračem s kojim Nadal nije igrao. Problem s ovako rigoroznim modelom je što se uvelike smanjuje skup podataka za učenje pa može doći do podnaučenosti. Jedna od ideja za manje rigorozan model je dodavanje težina na mečeve, odnosno da se mečevi sa zajedničkim protivnicima gledaju kao bitniji u odnosu na one koji nisu sa zajedničkim protivnicima. Također, u [4] je iznesena zanimljiva ideja o rekurzivnom pristupu ovoj strategiji, tj. da se u obzir uzmu i protivnici protivnika (ili protivnici protivnika protivnika, ovisno o dubini rekurzije). To bi doprinijelo veličini skupa podataka i time bi se izbjegla podtreniranost, ali treba paziti s dubinom rekurzije jer korištenje ove strategije s predubokom rekurzijom bi moglo postati ekvivalentno zanemarivanju modela zajedničkih protivnika tako što bi se skoro svi dostupni podaci uzeli u obzir.

Ovaj model nije korišten u ovom radu, ali u nekom od budućih radova na ovu temu će biti implementiran.

5.3. Ulazne varijable

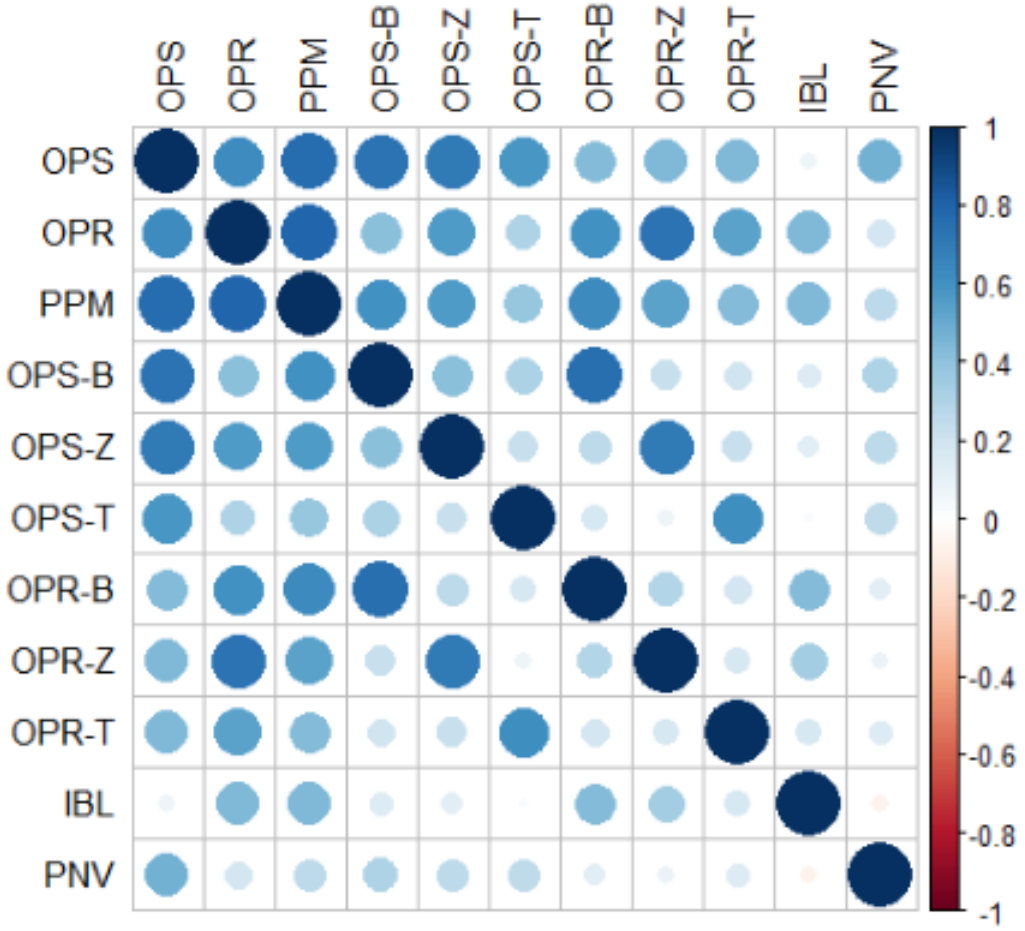
Tablica 5.1: Ulazne varijable s njihovim kraticama

Naziv značajke	Kratica značajke	
Postotak osvojenih poena na servisu	**OPS**	**Osnovne značajke**
Postotak osvojenih poena na returnu	**OPR**	
Postotak pobijeđenih mečeva	**PPM**	
Postotak osvojenih poena na servisu na tvrdoj podlozi	**OPS-B** (beton)	**Podloge**
Postotak osvojenih poena na servisu na zemlji	**OPS-Z**	
Postotak osvojenih poena na servisu na travi	**OPS-T**	
Postotak osvojenih poena na returnu na tvrdoj podlozi	**OPR-B** (beton)	
Postotak osvojenih poena na returnu na zemlji	**OPR-Z**	
Postotak osvojenih poena na returnu na travi	**OPR-T**	
Postotak iskorištenih break lopti	**IBL**	**Psihološka sprema**
Postotak osvajanja poena na servisu nakon osvojena prva dva poena	**PNV** (poen nakon vodstva)	
Podloga na kojoj se igra meč	**POD**	**Kategorička varijabla**

Oznaku igrača na kojeg se varijabla odnosi zapisivat ćemo brojem nakon kratice varijable (npr. OPS_1 – postotak osvojenih poena na servisu prvog igrača).

Na slici (**Slika 5.3**) prikazana je matrica korelacija između ulaznih varijabli. Dobivene korelacije su u većoj mjeri očekivane. Vidimo da kada igrač dobro servira na određenoj

podlozi da će onda uglavnom dobro i primati servis na toj podlozi. Iskorištenost break lopti ima visoku korelaciju s učinkovitošću igrača na primanju servisa, dok osvajanje poena nakon vodstva na servisu najviše ovisi o samom postotku osvajanja poena na servisu.



Slika 5.3: Matrica korelacija

Sve ove značajke za svakog igrača posebno spremljene su u tablicu *statistikaIgraca* (svaki redak je jedan igrač). S obzirom da nisu svi igrači ispunili preduvjete za izračun značajki (igrač nema odigran meč na travi, nije imao break priliku itd.), tablica sadrži dosta NA vrijednosti (NA je način na koji R označava da je podatak nepoznat). U nastavku je opisan način na koji smo se nosili s tim vrijednostima.

Za značajke OPS, OPR i PPM nije bilo NA vrijednosti, pa ćemo te značajke koristiti za izračun očekivanih vrijednosti značajki igrača koje sadrže NA vrijednost. Tako smo za popunjavanje vrijednosti povezanih sa servisom igrača (OPS-B, OPS-Z, OPS-T, PNV) koristili varijablu OPS, a za popunjavanje vrijednosti povezanih sa primanjem servisa igrača (OPR-B, OPR-Z, OPR-T, IBL) varijablu OPR. Kako bismo postigli što veću točnost u izračunu očekivanja nepoznatih podataka, uspoređivali smo aritmetičke sredine značajki

koje mijenjamo sa aritmetičkim sredinama značajki kojima ih mijenjamo. U tu svrhu provodili smo t-test na dva uzorka sa početnom hipotezom o jednakosti aritmetičkih sredina ($\mu_1 = \mu_2$) uz razinu značajnosti od 5%.

Započnimo s podacima po podlogama. Za svaku podlogu proveden je t-test skupa OPS sa skupovima OPS po podlogama. U tablici (**Tablica 5.2**) prikazani su rezultati testova. Iz dobivenih rezultata zaključujemo da za postotak osvojenih poena na servisu kada se igra na travnatoj podlozi možemo na razini značajnosti od 5% odbaciti hipotezu o jednakosti aritmetičkih sredina. S ovim saznanjima, popunjavamo NA vrijednosti na sljedeći način: nepoznate vrijednosti OPS-B i OPS-Z zamijenit ćemo s vrijednosti OPS tog igrača, a nepoznate vrijednosti OPS-T zamijenit ćemo s vrijednosti OPS tog igrača pomnoženom s kvocijentom aritmetičke sredine OPS-T i aritmetičke sredine OPS.

Tablica 5.2: Rezultati t-testa - servis po podlogama

Kratica značajke	$\bar{X}$(značajka)	$\bar{X}(OPS)$	p-vrijednost	$p < \alpha$
OPS-B	62,99	62,35	0,103	NE
OPS-Z	62,19	62,35	0,375	NE
OPS-T	65,28	62,35	$1{,}6 \times 10^{-7}$	DA

Sličan postupak provodimo na OPR varijablama. Rezultati testova prikazani su u tablici (). U ovom slučaju ne možemo na razini značajnosti od 5% odbaciti hipotezu o jednakosti aritmetičkih sredina ni za jednu promatranu varijablu pa ćemo sve nepoznate vrijednosti iz ovih skupova zamijeniti sa OPR igrača.

Tablica 5.3: Rezultati t-testa - return po podlogama

Kratica značajke	$\bar{X}$(značajka)	$\bar{X}(OPR)$	p-vrijednost	$p < \alpha$
OPR-B	43,31	43,12	0,321	NE
OPR-Z	43,44	43,12	0,195	NE
OPR-T	43,21	43,12	0,413	NE

Isti test sada provodimo na varijablama IBL i PNV. IBL ćemo uspoređivati sa OPR budući da se poen na kojem igrač može iskoristiti break priliku igra na primanju servisa, a PNV sa OPS jer se promatrani poen igra pri vodstvu na servisu. Rezultati testova prikazani su u tablicama (**Tablica 5.4** i **Tablica 5.5**). U oba slučaja odbacujemo nultu (početnu) hipotezu te ćemo za popunjavanje nepoznatih vrijednosti množiti OPR i OPS s kvocijentom aritmetičkih sredina prikazanih u tablicama.

Tablica 5.4: Rezultat t-testa - break prilike

Kratica značajke	$\bar{X}(IBL)$	$\bar{X}(OPR)$	p-vrijednost	$p < \alpha$
IBL	39,73	43,12	$3{,}2 \times 10^{-6}$	DA

Tablica 5.5: Rezultat t-testa - vodstvo na servisu

Kratica značajke	$\bar{X}(PNV)$	$\bar{X}(OPS)$	p-vrijednost	$p < \alpha$
PNV	65,71	62,35	$8{,}2 \times 10^{-10}$	DA

6. Rezultati

U ovom poglavlju razmatrat ćemo rezultate izrađenih modela kroz svojstva navedena u ranijim poglavljima. Za samu izradu modela često će biti korišteni paketi programskog jezika R o kojima će se posebno govoriti u poglavljima u kojima budu korišteni. Dodatno, za izradu matrica zabune i analizu podataka iste korištena je funkcija *ConfusionMatrix* iz paketa *caret*. U prikazima matrice zabune broj 1 označava prvog igrača, a broj 2 drugog igrača (Pobjeda 1 znači pobjedu prvog igrača, a Pobjeda 2 pobjedu drugog igrača).

6.1. Model Markovljevih lanaca

U poglavlju 3.1 ukratko je opisan način rada Markovljevih lanaca te njihova primjena u tenisu. U svrhu izrade ovog modela formiramo podatkovni skup koji sadrži sve mečeve s ATP turnira koji su odigrani u 2015. i 2016. godini zajedno s postotkom osvojenih poena na servisu za oba suparnika u meču. Postoji nekoliko dostupnih radova koji koriste slične značajke za modeliranje teniskih mečeva Markovljevim lancima kao što su [1] i [4]. Ovi radovi svojim rezultatima pokazuju da je tenis gotovo idealan sport za ovakvo modeliranje zbog toga što se poeni, gemovi i setovi lako prikazuju kao stanja te su prijelazi među tim stanjima relativno lako formulirani matematičkim formulama uz pretpostavku nezavisnosti i jednolike razdiobe značajki.

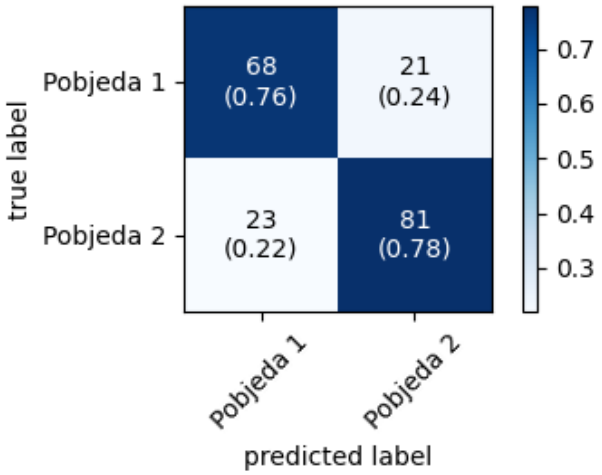
Markovljeve lance za tenis programski smo izveli u programskom jeziku R korištenjem tablica prijelaza između stanja. Tablicom prijelaza za gem računa se vjerojatnost osvajanja gema za igrača koji servira, ta vjerojatnost se propagira dalje u tablicu prijelaza za set, a vjerojatnosti dobivene tablicom prijelaza za set se dalje propagiraju u tablice prijelaza za meč na 2 dobivena seta (engl. *best of 3*) i na 3 dobivena seta (engl. *best of 5*). U radu [6] pokazano je da pitanje tko prvi servira u meču ili setu nema statističku važnost, tj. ne daje niti jednom igraču statistički značajnu prednost. Iz tog razloga, pri izračunu predviđanih vrijednosti ovim modelom, računali smo na to da prvi igrač uvijek prvi servira u svakom setu.

6.1.1. Rezultati

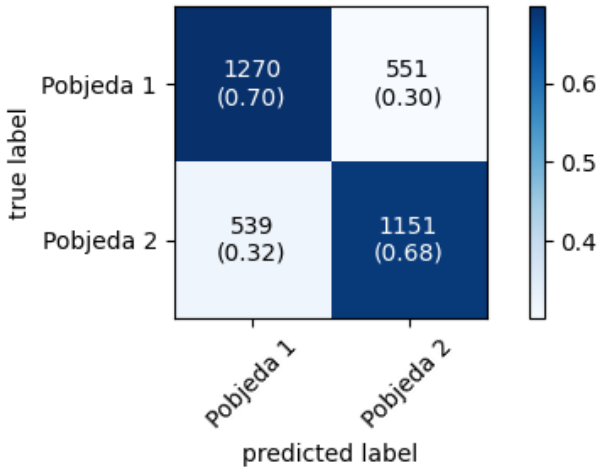
Prvo smo testirali ispravnost modela usporedbom s podacima iznesenim u [8]. Ovaj rad se bavi kompletnom matematikom zasnovanom na Markovljevim lancima u tenisu i naš model je za jednake ulaze davao jednake rezultate onima opisanima u tome radu te zaključujemo da je model ispravno implementiran.

Primjenom modela Markovljevih lanaca na cijeli promatrani podatkovni skup koji sadrži 3511 mečeva iz ATP kategorije postigli smo rezultat od 68.95% pogođenih ishoda meča. Rezultat je sličan rezultatima iz ostalih radova koji se bave ovim područjem, [7] za Australian Open 2003. godine postiže 72,4% pogodaka, dok u [4] model za sve ATP mečeve odigrane u 2011. godini postiže preciznost od oko 65,5%, što je lošiji rezultat nego u našem slučaju. Osim preciznosti od 68,95%, model je postigao osjetljivost od 69,7% te specifičnost od 68,1%.

Pokretanjem simulacije samo na mečeve odigrane na Wimbledonu dobivamo rezultat od čak 77,2% pogodaka što je najbolji rezultat koji smo dobili simulacijom turnira. Rast u uspješnosti modela za Wimbledon je očekivan (iako ne u ovakvoj mjeri) zbog toga što je trava najbrža podloga u tenisu i time najviše pogoduje igraču koji servira, što znači da će igrači češće osvajati gemove na svom servisu, a to pogoduje našem modelu s obzirom na to da su jedini ulazni podaci učinkovitosti igrača na servisu.



Slika 6.1: Matrica zabune simulacije Wimbledona



Slika 6.2: Matrica zabune predviđanja cijelog skupa

6.2. Modeli strojnog učenja

Za implementaciju modela predviđanja ishoda teniskih mečeva koristeći algoritme strojnog učenja potrebno je razdvojiti podatkovni skup na skup za učenje i skup za ispitivanje. Korišten je paket *caTools* kako bi se nasumično odvojila dva navedena skupa. Za većinu modela korištenih u ovom dijelu najbolje rezultate dobili smo kada smo 80% podataka dodijelili skupu za učenje, a preostalih 20% skupu za ispitivanje. Takvu raspodjelu postavit ćemo kao pretpostavljenu, odnosno ako se ne spomene da je drugačije, podrazumijeva se ovakva raspodjela. Preciznost modela računali smo kao udio ispravnih klasifikacija u ukupnom broju podataka, a pri izračunu osjetljivosti i specifičnosti kao pozitivnu vrijednost odabrali smo pobjedu prvog igrača.

Stopa neinformiranosti skupa za ispitivanje je 51,85%. Ova vrijednost nam je bitna zbog provođenja testa koji se odvija u pozadini već navedene funkcije *ConfusionMatrix*. Test koji ova funkcija koristi i koji će nam dati odgovor na pitanje je li preciznost našeg modela statistički značajno veća od stope neinformiranosti naziva se binomni test. *Binomni test je egzaktan test statističke značajnosti odstupanja od očekivane raspodjele opažanja u dvije kategorije*[13]. U provođenju ovog testa izračunat ćemo 95%-tne intervale pouzdanosti za preciznost modela te na osnovu dobivenih p-vrijednosti zaključiti je li preciznost modela značajno veća od stope neinformiranosti. Nulta hipoteza testa biti će da je preciznost modela jednaka stopi neinformiranosti. U tablicama koje su prikazane u sljedećim potpoglavljima

[13] Preuzeto s https://en.wikipedia.org/wiki/Binomial_test

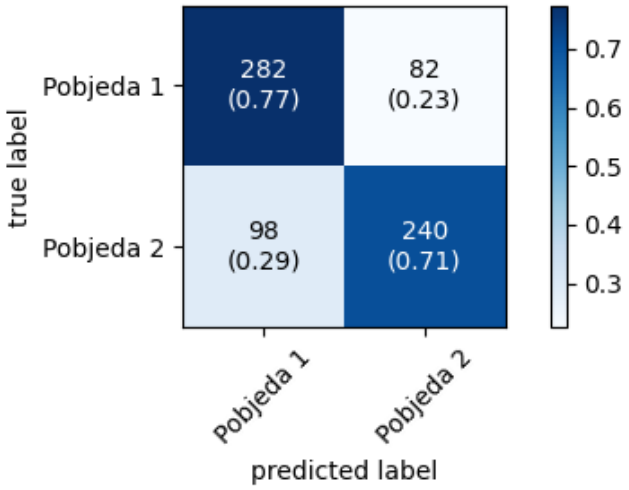
prikazani interval povjerenja i p-vrijednost su upravo vrijednosti dobivene binomnim testom. Također, bitno je napomenuti da, kod prikaza odabira ulaznih varijabli u tablicama, stupac podloga označava sve značajke koje imaju vezu s podlogom (OPS-B, OPS-Z, OPS-T, OPR-B, OPR-Z, OPR-T te kategorička varijabla POD) te ako je označen, znači da su modelu kao ulazne varijable predane sve navedene značajke.

Isprobane su razne kombinacije ulaznih varijabli za svaki korišteni model, ali zbog sažetosti rada biti će prikazani rezultati objektivno najboljih modela te modela koji daju naznake nekih zanimljivosti.

6.2.1. Logistička regresija

Za izradu klasifikatora i predviđanje modelom logističke regresije korištene su funkcije *glm* (Generalized Linear Models) i *predict* iz paketa *stats*. Za predviđanje pobjednika meča korištene su razne kombinacije ulaznih varijabli. Najbolji rezultat od 74,4% pogođenih ishoda postignut je koristeći sve ulazne varijable. Matrica zabune ove simulacije prikazana je na slici (**Slika 6.3**). Iz prikazane matrice zabune možemo iščitati osjetljivost od 77,5% i specifičnost od 71,01%.

Za sve kombinacije varijabli prikazane u tablici možemo uz razinu značajnosti od 5% odbaciti nultu hipotezu te zaključujemo da je svaki prikazani model, s obzirom na izrazito niske p-vrijednosti, značajno bolje preciznosti od stope neinformiranosti.



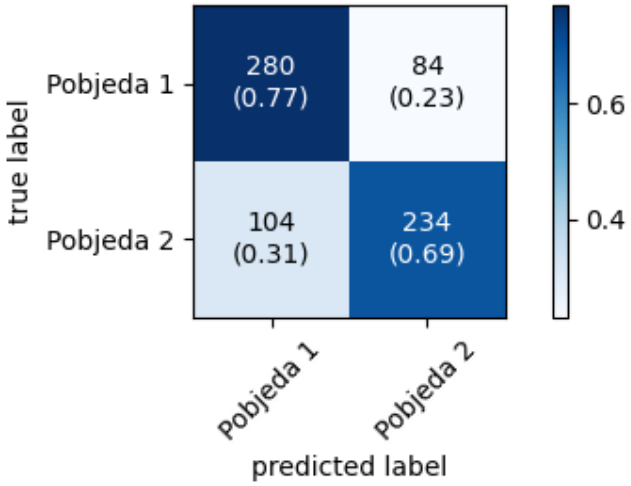
Slika 6.3: Matrica zabune za logističku regresiju koristeći sve ulazne varijable

Tablica 6.1: Prikaz rezultata modela logističke regresije s obzirom na ulazne varijable

OPS	OPR	PPM	Podloga	IBL	PNV	Preciznost	95%-tni interval povjerenja	p-vrijednost
■	■	■	■	■	■	74,36%	70,96%-77,55%	2×10^{-16}
■	■	■	■			74,36%	70,96%-77,55%	2×10^{-16}
		■		■		73,79%	70,37%-77,01%	2×10^{-16}
■	■					71,79%	68,31%-75,1%	2×10^{-16}
				■	■	57,98%	54,23%-61,66%	0,0006

6.2.2. K-najbližih susjeda

Slično kao i kod logističke regresije, za model k-najbližih susjeda isprobane su razne kombinacije ulaznih varijabli. U ovom modelu dodatno je trebalo eksperimentirati sa vrijednosti parametra k (broja najbližih susjeda). Za treniranje i predviđanje korištena je funkcija *knn* iz paketa *class* koja jednim pozivom radi i treniranje i predviđanje. I u ovom modelu su se kompleksnije ulazne varijable pokazale kao manje korisne te je model najbolje rezultate dao za ulazne varijable PPM i $k \approx 30$. Model k-najbližih susjeda je za većinu kombinacija ulaznih varijabli davao lošije rezultate od logističke regresije, a najbolji postignuti rezultat bio je 73,2% pogođenih ishoda za ulazne varijable PPM_1 i PPM_2 uz osjetljivost od 76,9% i specifičnost od 69,2%. Na slici (**Slika 6.4**) je prikazana matrica zabune najboljeg dobivenog modela k-najbližih susjeda, a u tablici (**Tablica 6.2**) prikazani su rezultati raznih kombinacija ulaznih varijabli. Treba primijetiti da su p-vrijednosti i kod ovog modela za sve kombinacije ulaznih varijabli jako niske te da vrlo lako odbacujemo nultu hipotezu.



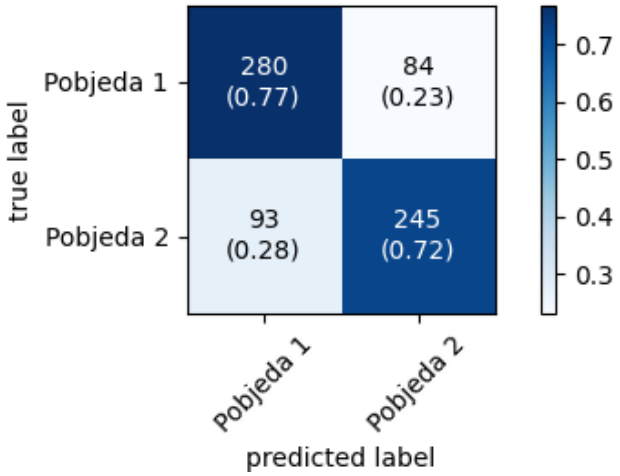
Slika 6.4: Matrica zabune za K-najbližih susjeda s ulaznim varijablama PPM_1 i PPM_2

Tablica 6.2: Prikaz rezultata modela K-najbližih susjeda s obzirom na ulazne varijable

OPS	OPR	PPM	Podloge	IBL	PNV	Preciznost	95%-tni interval povjerenja	p-vrijednost
						73,22%	69,78%-76,46%	2×10^{-16}
						70,09%	66,55%-73,45%	2×10^{-16}
						69,66%	66,11%-73,04%	2×10^{-16}
						66,38%	62,75%-69,87%	$4{,}6 \times 10^{-15}$
						59,83%	56,1%-63,48%	$1{,}3 \times 10^{-5}$

6.2.3. Stroj potpornih vektora

Za treniranje modela stroja potpornih vektora korištena je funkcija *svm* iz paketa *e1071*. Ponavljamo sličan postupak odabira kombinacija ulaznih značajki. Korištenjem ovog modela dobili smo iznimno dobre rezultate. Neki od boljih i zanimljivijih rezultata prikazani su u tablici (**Tablica 6.3**). Vrlo važan dio izrade modela stroja potpornih vektora bio je odabir kernel funkcije. Većinom je najbolje rezultate davala polinomna kernel funkcija, ali najbolju preciznost postigli smo linearnom kernel funkcijom i postavljanjem svih dostupnih varijabli na ulaz modela. Matrica zabune modela sa najvećom preciznošću prikazana je na slici (**Slika 6.5**). Model je postigao preciznost od 74,8%, osjetljivost od 76,9% i specifičnost od 72,5%.



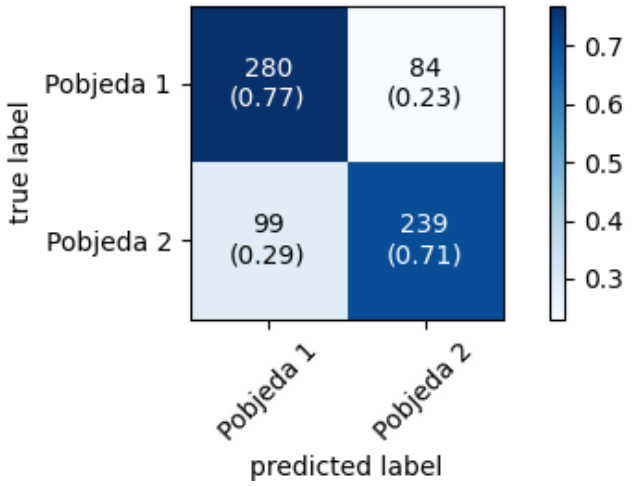
Slika 6.5: Matrica zabune stroja potpornih vektora sa svim ulaznim varijablama

Tablica 6.3: Prikaz rezultata modela stroja potpornih vektora s obzirom na ulazne varijable

OPS	OPR	PPM	Podloge	IBL	PNV	Preciznost	95%-tni interval povjerenja	p-vrijednost
						74,79%	71,4%-77,96%	2×10^{-16}
						74,5%	71,11%-77,69%	2×10^{-16}
						73,93%	70,52%-77,14%	2×10^{-16}
						71,23%	67,72%-74,55%	2×10^{-16}
						62,68%	58,98%-66,27%	$4{,}7 \times 10^{-9}$

6.2.4. Naivni Bayesov klasifikator

Za izradu Bayesovog klasifikatora korištena je funkcija *naiveBayes* iz paketa *e1071*. Ovaj klasifikator je u većini slučajeva postizao lošije rezultate od modela koji koriste stroj potpornih vektora ili logističku regresiju. Ipak, koristeći samo ulazne varijable PPM_1 i PPM_2 Bayesov klasifikator postiže preciznost od 73,93% što je sasvim solidan rezultat. Matrica zabune najboljeg izrađenog modela dana je u nastavku (**Slika 6.6**), dok je tablica s ostalim rezultatima prikazana u tablici (**Tablica 6.4**).



Slika 6.6: Matrica zabune za naivni Bayesov klasifikator s ulaznim varijablama PPM_1 i PPM_2

Tablica 6.4: Prikaz rezultata modela naivnog Bayesovog klasifikatora s obzirom na ulazne varijable

OPS	OPR	PPM	Podloge	IBL	PNV	Preciznost	95%-tni interval povjerenja	p-vrijednost
						73,93%	70,52%-77,14%	2×10^{-16}
						71,37%	67,87%-74,69%	2×10^{-16}
						69,66%	66,11%-73,04%	2×10^{-16}
						70,66%	67,13%-74,0%	2×10^{-16}
						59,4%	55,66%-63,06%	$3{,}5 \times 10^{-5}$

6.2.5. Umjetna neuronska mreža

Za razvoj neuronskih mreža u R-u postoji nekoliko javno dostupnih paketa. Mi ćemo za potrebe ovog rada koristiti paket *h2o* i njegove funkcije *deeplearning* za učenje klasifikatora te *predict* za predviđanje rezultata na skupu za ispitivanje. Za razliku od većine dosad prikazanih modela strojnog učenja, pri korištenju neuronskih mreža moguće je namještati mnoge hiperparametre za izradu klasifikatora koji uvelike utječu na učinkovitost modela. *Ideja odabira hiperparametara je da želimo da dobivena neuronska mreža bude što jednostavnija, ali da istovremeno što bolje klasificira podatke*[14]. Neki od najzanimljivijih hiperparametara koji su promatrani pri izradi što bolje neuronske mreže su: aktivacijska funkcija, broj skrivenih slojeva, broj neurona u skrivenom sloju, stopa učenja (engl. *learning rate*) i broj epoha.

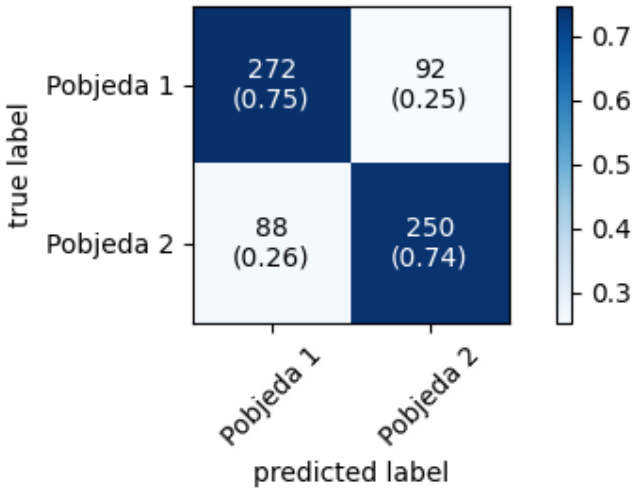
U funkciji *deeplearning* aktivacijska funkcija zadaje se promjenom parametra *activation*. Tangens hiperbolni i ispravljačka (engl. rectifier) funkcija uglavnom su davale najbolje rezultate te ćemo samo njih i koristiti u prikazivanju rezultata. Broj skrivenih slojeva, kao i broj neurona u skrivenom sloju zadaje se pridavanjem vektora parametru hidden tako što dimenzije vektora označavaju broj skrivenih slojeva, a vrijednosti po pozicijama broj neurona u svakom sloju (npr. vektor (3, 3) označava dva skrivena sloja sa po tri neurona u svakom).

U tablicama u nastavku (**Slika 6.7** i **Tablica 6.5**) prikazani su najbolji dobiveni modeli za svaku prikazanu kombinaciju ulaznih varijabli te matrica zabune modela s najvećom preciznošću.

Za većinu prikazanih modela nisu korištene pretjerano komplicirane neuronske mreže koje dugo treba trenirati iz razloga što je vrlo lako dolazilo do prenaučenosti, a povećanjem kompleksnosti nismo dobivali ništa bolje rezultate. Neuronske mreže dale su sasvim solidne rezultate s obzirom da su imale prilično mali skup za učenje. Ipak, niti jedan od modela nije uspio nadmašiti rezultate postignute modelom stroja potpornih vektora. Najbolju preciznost od 74,36% postigao je model sa ulaznim varijablama PPM_1, PPM_2, IBL_1, IBL_2, PNV_1 i PNV_2. Postignuta osjetljivost iznosi 74,73%, a postignuta specifičnost 73,96%. Za ovaj model korištena je aktivacijska funkcija tangens hiperbolni, 10 neurona u skrivenom sloju,

[14] Preuzeto s https://towardsdatascience.com/neural-networks-parameters-hyperparameters-and-optimization-strategies-3f0842fac0a5

prilagodljiva stopa učenja te broj iteracija (epoha) 100. Kod izrade ostalih neuronskih mreža koristili smo slične hiperparametre, osim što je u većini slučajeva korištena ispravljačka aktivacijska funkcija.



Slika 6.7: Matrica zabune za umjetnu neuronsku mrežu sa ulaznim varijablama PPM_1, PPM_2, IBL_1, IBL_2, PNV_1 i PNV_2

Tablica 6.5: Prikaz rezultata modela neuronske mreže s obzirom na ulazne varijable

OPS	OPR	PPM	Podloge	IBL	PNV	Preciznost	95%-tni interval povjerenja	p-vrijednost
						74,36%	70,96%-77,55%	2×10^{-16}
						72,36%	68,9%-75,64%	2×10^{-16}
						72,22%	68,75%-75,51%	2×10^{-16}
						70,94%	67,43%-74,28%	2×10^{-16}
						69,8%	66,25%-73,18%	2×10^{-16}

6.3. Usporedba rezultata

U ovom poglavlju dotaknut ćemo se cjelokupnih rezultata te ćemo usporediti rezultate dobivenih modela. Izrada klasifikatora za predviđanje ishoda teniskih mečeva bila je uspješna te smo kod svakog prikazanog modela odbacili hipotezu o tome da je preciznost modela jednaka stopi neinformiranosti. Najbolje rezultate dobili smo metodom stroja potpornih vektora, dok su se na cjelokupnom skupu podataka najlošijima pokazala predviđanja modela Markovljevih lanaca. U tablici (**Tablica 6.6**) su prikazani modeli sa najvećom postignutom preciznošću za svaku korištenu metodu. Važno je napomenuti da su kod većine korištenih metoda najbolji rezultati postignuti korištenjem svih ulaznih varijabli, što daje naznake da svaka korištena ulazna varijabla doprinosi povećanju količine informacije koju model može iskoristiti za predviđanje.

Tablica 6.6: Preciznost svih modela

Model	Preciznost
Stroj potpornih vektora	74,79%
Linearna regresija	74,36%
Umjetna neuronska mreža	74,36%
Naivni Bayesov klasifikator	73,93%
K-najbližih susjeda	73,22%
Markovljevi lanci	68,95%

Zaključak

Cilj ovog rada bio je provesti eksploratornu analizu skupa podataka teniskih mečeva te na osnovu metode Markovljevih lanaca i algoritama strojnog učenja izraditi i usporediti modele za predviđanje pobjednika u teniskom meču. Predviđanje pobjednika meča svodi se na klasifikacijski problem s dvije klase, pobjeda prvog i pobjeda drugog igrača. U sklopu ovog rada objašnjena su teniska pravila i zanimljivosti, provedeno je čišćenje i obrada podatkovnog skupa u svrhu dobivanja ulaznih parametara te su na osnovu tih ulaznih parametara izrađeni prediktivni modeli. U praktičnom dijelu rada opisan je model Markovljevih lanaca implementiran preko tablica prijelaza između stanja te su implementirani i opisani modeli zasnovani na algoritmima strojnog učenja.

Iz rezultata prikazanih u posljednjem poglavlju zaključujemo da je izrada modela bila uspješna te da je učinkovitost modela i više nego zadovoljavajuća. Najbolje rezultate dobili smo korištenjem algoritma stroja potpornih vektora. Rezultati bi se mogli unaprijediti proširivanjem podatkovnog skupa, dubljom analizom odabira parametara za algoritme strojnog učenja te primjenom modela zajedničkih protivnika. Budući rad na ovoj temi uključivao bi predviđanje drugih značajki teniskog meča kao što je broj odigranih poena u meču te korištenje već izrađenih modela u analizi strategija klađenja.

Literatura

[1] Barnett, T., Clarke, S.R. *Combining player statistics to predict outcomes of tennis matches.* IMA Journal of Management Mathematics, 16,2 (2005), str. 113-120.

[2] Jackson, D., Mosurski, K. Heavy defeats in tennis: *Psychological momentum or random effect?*, Chance 10,2 (1997), str. 27-34.

[3] Klaassen, F.J.G.M., Magnus, J.R. *Are points in tennis independent and identically distributed? Evidence from a dynamic binary panel data model*, Journal of the American Statistical Association, 96,454 (2001), str. 500-509.

[4] Knottenbelt, W.J., Spanias, D., Madurska, A.M. *A common-opponent stochastic model for predicting the outcome of professional tennis matches,* Computers & Mathematics with Applications, 64,12 (2012): str. 3820-3827.

[5] Sipko, M., Knottenbelt, W. *Machine learning for the prediction of professional tennis matches,* MEng computing-final year project. Imperial College London, 2015.

[6] MacPhee, I. M., Rougier J., Pollard, G.H. *Server advantage in tennis matches,* Journal of applied probability, (2004), str. 1182-1186.

[7] Barnett, T., Brown, A., Clarke, S. *Developing a model that reflects outcomes of tennis matches.* Proceedings of the 8th Australasian Conference on Mathematics and Computers in Sport, Coolangatta, Queensland, (2006)

[8] Barnett T., Brown A. *The Mathematics of Tennis*, Strategic Games, 2012.

[9] Šarčević, A. *Prediktivna analiza i modeliranje teniskih mečeva.* Diplomski rad. Sveučilište u Zagrebu Fakultet elektrotehnike i računarstva, 2017.

Sažetak

U ovom radu provedena je eksploratorna analiza podatkovnog skupa teniskih mečeva te su na osnovu značajki iz tog skupa izrađeni prediktivni modeli za predviđanje pobjednika u teniskom meču. U prvim poglavljima opisana su pravila tenisa te dijelovi statističke analize podataka, statistički modeli i modeli strojnog korišteni kasnije u radu. U drugom dijelu rada izrađeni su izloženi prediktivni modeli te su prikazani rezultati predviđanja na osnovu raznih kombinacija ulaznih varijabli.

Ključne riječi: tenis, strojno učenje, Markovljevi lanci, klasifikacija, statistička analiza podataka

Summary

In this paper, an exploratory analysis of the data set containing data about tennis matches was performed, and based on the features extracted from that data set, predictive models were developed to predict the winner of a tennis match. The first chapter describes rules of tennis and parts of statistical data analysis as well as statistical and machine learning models used later in the paper. In the second part of the paper, previously described models are developed and the results of predictions with various combinations of input variables are presented.

Keywords: tennis, machine learning, Markov chains, classification