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](#_Toc73788874)

[1. Osnove tenisa 2](#_Toc73788875)

[2. Statistička analiza podataka 3](#_Toc73788876)

[2.1. Eksploratorna statistička analiza 3](#_Toc73788877)

[2.2. Deskriptivna statistička analiza 3](#_Toc73788878)

[2.2.1. Mjere centralne tendencije 3](#_Toc73788879)

[2.2.2. Mjere rasipanja 4](#_Toc73788880)

[2.3. Matrica zabune 4](#_Toc73788881)

[3. Statistički modeli 7](#_Toc73788882)

[3.1. Markovljev model 7](#_Toc73788883)

[4. Strojno učenje 9](#_Toc73788884)

[4.1. Logistička regresija 10](#_Toc73788885)

[4.2. K-najbližih susjeda 11](#_Toc73788886)

[4.3. Stroj potpornih vektora 11](#_Toc73788887)

[4.4. Naivni Bayesov klasifikator 13](#_Toc73788888)

[4.5. Umjetna neuronska mreža 14](#_Toc73788889)

[5. Obrada podatkovnog skupa 16](#_Toc73788890)

[5.1. Podatkovni skup 16](#_Toc73788891)

[5.2. Predobrada podataka 16](#_Toc73788892)

[5.2.1. Psihološki moment 17](#_Toc73788893)

[5.2.2. Model zajedničkih protivnika 17](#_Toc73788894)

[5.3. Oznake značajki 19](#_Toc73788895)

[6. Rezultati 20](#_Toc73788896)

[6.1. Model Markovljevih lanaca 20](#_Toc73788897)

[6.1.1. Rezultati 21](#_Toc73788898)

[6.2. Modeli strojnog učenja 22](#_Toc73788899)

[6.2.1. Logistička regresija 22](#_Toc73788900)

[6.2.2. K-najbližih susjeda 23](#_Toc73788901)

[6.2.3. Stroj potpornih vektora 24](#_Toc73788902)

[6.2.4. Naivni Bayesov klasifikator 25](#_Toc73788903)

[6.2.5. Umjetna neuronska mreža 26](#_Toc73788904)

[Zaključak 28](#_Toc73788905)

[Literatura 29](#_Toc73788906)

[Sažetak 30](#_Toc73788907)

[Summary 31](#_Toc73788908)

[Skraćenice 32](#_Toc73788909)

[Privitak 33](#_Toc73788910)

# 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. Osim predviđanja konačnog pobjednika meča, obratit će se pažnja i na predviđanje duljine meča, odnosno ukupnog broja bodova koji će se odigrati u meču. Cilj rada je provesti eksploratornu analizu dostupnih podataka uz naglasak na primjenu algoritama strojnog učenja na predikciju osnovnih karakteristika 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 rad.

# 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. 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. 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. 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).

# Statistička analiza podataka

Statistička analiza podataka je

## 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

## 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. Najčešće korištene mjere 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.

### Mjere centralne tendencije

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

( 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:

( 2.2 )

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

### Mjere rasipanja

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

( 2.3 )

N – veličina populacije

µ – aritmetička sredina populacije

xi – 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:

( 2.4 )

gdje je aritmetička sredina uzorka.

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

## 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 retci 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 sa ukupnim brojem klasificiranih podataka:

Osjetljivost (engl. sensitivity) računa se kao broj pogođenih pozitiva (klasa 1) podijeljen sa ukupnim brojem pozitiva:

Specifičnost (engl. specificity) računa se kao broj pogođenih negativa (klasa 0) podijeljen sa ukupnim brojem negativa:

**Tablica 2.1:** Primjer matrice konfuzije

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

Stopa neinformiranosti (engl. No Information Rate) najbolji je mogući rezultat predviđanja uz nedostatak podataka osim razdiobe klasa koje pokušavamo predvidjeti[[1]](#footnote-1). Iz primjera matrice dane u tablici (Tablica 2.1) vidimo da većina (60,5%) jedinki iz populacije poprima klasu 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 uznačava udio većinske klase (u ovom slučaju 60,5%).

Uzimajući u obzir stopu neinformiranosti, moguće je sa 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 jednostrani test da bi provjerili je li preciznost modela značajno bolja od stope neinformiranosti.

# Statistički modeli

## 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: {\displaystyle \Pr(X\_{n+1}=x|X\_{n}=x\_{n},\ldots ,X\_{1}=x\_{1})=\Pr(X\_{n+1}=x|X\_{n}=x\_{n}).\,}

Markovljev lanac je slijed slučajnih varijabli X1, X2, X3, ... s Markovljevim svojstvom i to zato što su trenutno, buduće i prošlo stanje nezavisni.[[2]](#footnote-2)

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 3.1 prikazan je Markovljev lanac za jedan teniskim 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.

Diagram

Description automatically generated

**Slika 3.1** Markovljev lanac za teniski gem[[3]](#footnote-3)

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.

# 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.[[4]](#footnote-4)

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 ogromnim 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 predikciju ishoda teniskih mečeva. Također, za svaki algoritam su isprobane razne kombinacije ulaznih varijabli o kojima će biti više riječi u poglavlju 5 kada ćemo se baviti predobradom podataka i dobivanjem ulaznih varijabli.

## 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 dobivena 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:

( 4.1 )

Z dobivamo na sljedeći način:

( 4.2 )

xi – vrijednosti pojedinih varijabli

βi – vrijednosti koeficijenata koje logistička regresija određuje varijablama

A picture containing diagram

Description automatically generated

**Slika 4.1** Graf logističke funkcije[[5]](#footnote-5)

## 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:

( 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.

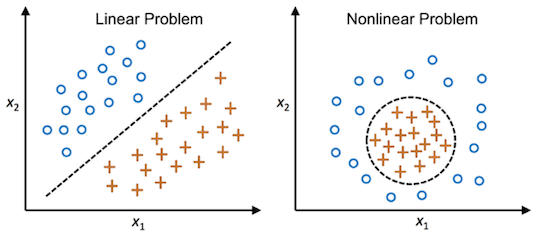
## Stroj potpornih vektora

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[[6]](#footnote-6)

U mnogim problemima linearna klasifikacija primjenom stroja potpornih vektora nije primjenjiva bez mapiranja 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[[7]](#footnote-7)

## Naivni Bayesov klasifikator

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

( 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:

( 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 (hMAP – maksimum aposteriori hipoteza)[[8]](#footnote-8).

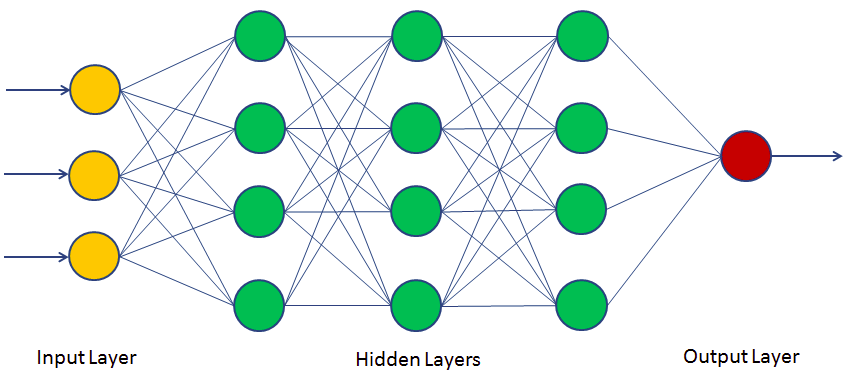
( 4.6 )

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.

## 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[[9]](#footnote-9). Na slici **Slika 4.4** prikazana je jednostavna potpuno povezana neuronska mreža s tri skrivena sloja neurona.



**Slika 4.4** Prikaz neuronske mreže[[10]](#footnote-10)

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

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

) ( 4.7 )

wi – težina i-te veze

xi – 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[[11]](#footnote-11). 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.

# Obrada podatkovnog skupa

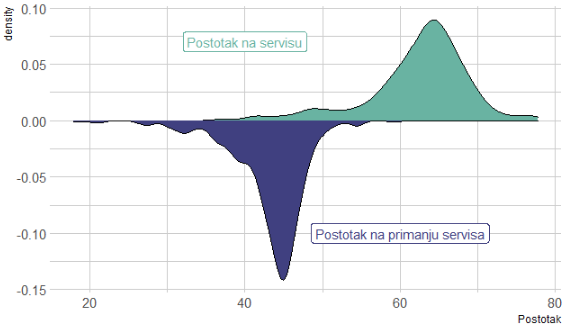
U praktičnom dijelu ovog rada obrađen je podatkovni skup u programskom jeziku R. U ovom poglavlju bavit ćemo se opisom podatkovnog skupa, predobradom podataka i intuicijom koja stoji iza korištenih tehnika u razvoju modela

## Podatkovni skup

Podatkovni skup preuzet je iz Hrvatske Lutrije 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). 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 i WTA razini.

## Predobrada podataka

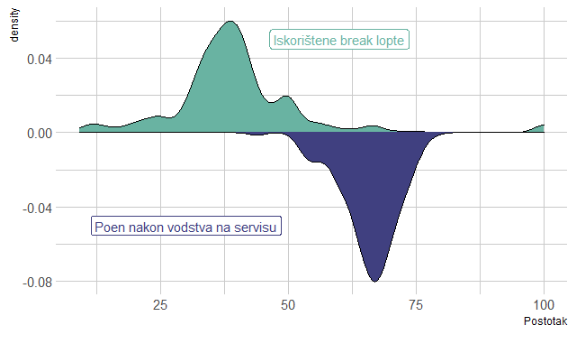
Predobrada podataka provodila se u više koraka i bila je velik dio praktičnog rada. Prvo je trebalo u radni okvir učitati sve dostupne tablice. Najosnovnije tablice, a ujedno i tablice s najviše podataka su sportevent tablice koje sadrže tijek svakog pojedinog teniskog meča po poenima. Sportevent tablice 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

### 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 u dobivenim rezultatima pa ti rezultati nisu korišteni u daljnjem radu.



**Slika 5.2:** Graf gustoće razdiobe psihološke značajke

### 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 predikciju. 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.

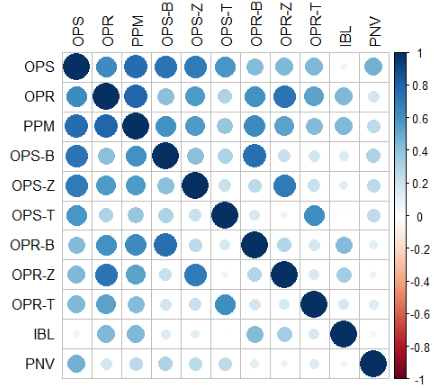
Ideja modela zajedničkih protivnika je da se kao značajke predikcije uzimaju vrijednosti izračunate na skupu mečeva koji su odigrani sa zajedničkim protivnicima dva igrača. Na primjer, najrigorozniji oblik ove strategije za izračun postotka osvojenog poena na servisu Novaka Đokovića kada igra s Rafaelom 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 podtreniranosti. 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.

## Oznake značajki

**Tablica 5.1:** Značajke s njihovim kraticama

|  |  |  |
| --- | --- | --- |
| Naziv značajke | Kratica značajke |  |
| Postotak osvojenih poena na servisu | **OPS** | **Osnovne značajke** |
| Postotak osvojenih poena na reternu | **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 reternu na tvrdoj podlozi | **OPR-B** (beton) |
| Postotak osvojenih poena na reternu na zemlji | **OPR-Z** |
| Postotak osvojenih poena na reternu 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) |

Oznaku igrača na kojeg se varijabla odnosi zapisivat ćemo brojem nakon kratice varijable (npr. OPS1 – postotak osvojenih poena na servisu prvog igrača).



**Slika 5.3:** Matrica korelacija

# Rezultati

U ovom poglavlju razmatrat ćemo rezultate izgrađenih 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 biti će korištena funkcija ConfusionMatrix iz paketa caret*[[12]](#footnote-12)*.

Stopa neinformiranosti za naš skup za ispitivanje je 51,85%.

Za kombinacije parametara smo uglavnom kombinirali osnovne značajke sa neosnovnim, isprobavali samo osnovne pa uspoređivali sa njima. Isprobavali samo psih moment, te sve skupa.

## 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 i WTA turnira koji su odigrani u 2016. i 2015. godini zajedno s postotkom osvojenih poena na servisu za oba suparnika u meču. Postoji nekoliko dostupnih radova koji koriste slične značajke za modeliranje teniskim 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 raspodjele 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 servira prvi u meču ili setu nema statističku važnost, tj. ne daje niti jednom igraču statistički značajnu prednost. Iz tog razloga, pri izračunu predikcija ovim modelom, računali smo na to da prvi igrač uvijek prvi servira u svakom setu.

### Rezultati

Prvo smo testirali ispravnost modela na osnovu podataka iznesenih 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 6630 mečeva iz ATP i WTA kategorije postigli smo rezultat od 67.5% 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 oko 65,5% pogodaka, što je lošiji rezultat nego u našem slučaju. Pokretanjem simulacije samo na mečeve odigrane u muškoj konkurenciji 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.

U tablicama 0 označava poraz prvog igrača, odnosno pobjedu drugog, a 1 označava pobjedu prvog igrača, odnosno poraz drugog.

**Tablica 6.1:** Matrica zabune simulacije Wimbledona

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 81 | 23 |
| 1 | 21 | 68 |

**Tablica 6.2:** Matrica zabune simulacije cijelog skupa

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 2142 | 1109 |
| 1 | 1044 | 2335 |

## Modeli strojnog učenja

Za implementaciju modela predikcije 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 80-20 raspodjela. Točnost modela računali smo kao udio ispravnih klasifikacija u ukupnom broju podataka. Modeli su zasad učeni i testirani samo na skupu muške konkurencije iz razloga što kombinacija muške i ženske konkurencije nije prihvatljiva zbog razlike u muškom i ženskom tenisu, a analizom ženskog tenisa u predobradi podataka nisu pronađene neke statistički značajne zanimljivosti koje bi uzrokovale da modeli strojnog učenja daju značajno drugačije rezultate za mušku i žensku konkurenciju.

### Logistička regresija

Za realizaciju i predviđanje koristeći logističku regresiju korištene su funkcije glm (Generalized Linear Models) i predict iz paketa stats*[[13]](#footnote-13)*. Za predikciju pobjednika meča korištene su razne kombinacije ulaznih varijabli. Zanimljivo je da je najbolji rezultat od 74,5% pogođenih ishoda postignut korištenjem samo četiri ulazne varijable – PPM1, PPM2, IBL1, i IBL2. Matrica konfuzije ove simulacije prikazana je u tablici (Tablica 6.3). Iz prikazane matrice konfuzije možemo iščitati osjetljivost od 77,5% i specifičnost od 71,3%. Ovaj rezultat daje naznake da bi osnovne značajke kao što je PPM u kombinaciji sa značajkom koja karakterizira psihološku spremnost kao što je IBL mogle biti dobra kombinacija u predviđanjima. Korištenjem svih dostupnih ulaznih varijabli model postiže rezultat od 73,5% pogodaka, a dobar rezultat postiže još i za ulazni skup koji sadrži postotke osvajanja poena na servisu i primanju servisa oba igrača (oko 73,6%).

**Tablica 6.3:** Matrica zabune za logističku regresiju s ulaznim varijablama PPM1, PPM2, IBL1 i IBL2

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 241 | 97 |
| 1 | 82 | 282 |

**Tablica 6.4:** Prikaz rezultata modela logističke regresije s obzirom na ulazne varijable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OPS | OPR | PPM | Podloge | IBL | PNV | Preciznost | 95% i.p. | p-vrijednost |
|  |  |  |  |  |  | 74,5% | 71,11%-77,69% |  |
|  |  |  |  |  |  | 73,65% | 70,22%-76,87% |  |
|  |  |  |  |  |  | 72,3% | 69,48%-76,19% |  |
|  |  |  |  |  |  | 68,95% | 65,38%-72,35% |  |
|  |  |  |  |  |  | 58,55% | 54,8%-62,22% | 0,0002 |

### 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 broja k (broja najbližih susjeda). Za treniranje i predviđanje korištena je funkcija knn iz paketa class koja u jednom koraku 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 ≈ 43. Model k-najbližih susjeda je za sve kombinacije ulaznih varijabli davao lošije rezultate od logističke regresije, a najbolji postignuti rezultat bio je 73,2% pogođenih ishoda za ulazne varijable PPM1 i PPM2 uz osjetljivost jednaku specifičnosti od 73,4%.

**Tablica 6.5:** Matrica zabune za K-najbližih susjeda s ulaznim varijablama PPM1 i PPM2

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 248 | 90 |
| 1 | 97 | 267 |

**Tablica 6.6:** Prikaz rezultata modela K-najbližih susjeda s obzirom na ulazne varijable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OPS | OPR | PPM | Podloge | IBL | PNV | Preciznost | 95% i.p. | p-vrijednost |
|  |  |  |  |  |  | 73,22% | 69,78%-76,46% |  |
|  |  |  |  |  |  | 71,08% | 67,57%-74,41% |  |
|  |  |  |  |  |  | 68,66% | 65,09%-72,08% |  |
|  |  |  |  |  |  | 66,38% | 62,75%-69,87% |  |
|  |  |  |  |  |  | 61,11% | 57,39%-64,74% |  |

### Stroj potpornih vektora

Za treniranje modela stroja potpornih vektora korištena je funkcija svm iz paketa e1071*[[14]](#footnote-14)*. Ponavljamo sličan postopak odabira kombinacija ulaznih značajki. U ovom modelu ponavlja se sličan uzorak kao kod logističke regresije. Ponovo smo najbolje rezultate dobili za kombinaciju PPM i značajki psihološkog stanja igrača (u ovom slučaju IBL i PNV). Dobiveni rezultat uz navedene ulazne značajke je oko 74,4% pogodaka uz osjetljivost od 78,0% i specifičnost od 70,4%. Ovaj rezultat je ipak za nijansu lošiji od rezultata kojeg smo dobili pri predviđanju logističkom regresijom. Matrica konfuzije najboljeg izgrađenog modela sa strojem potpornih vektora prikazana je u nastavku (Tablica6.7).

**Tablica 6.7:** Matrica zabune za stroj potpornih vektora s ulaznim varijablama PPM1, PPM2, IBL1, IBL2, PNV1, PNV2

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 238 | 100 |
| 1 | 80 | 284 |

**Tablica 6.8:** Prikaz rezultata modela stroja potpornih vektora s obzirom na ulazne varijable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OPS | OPR | PPM | Podloge | IBL | PNV | Preciznost | 95% i.p. | p-vrijednost |
|  |  |  |  |  |  | 74,36% | 70,96%-77,55% |  |
|  |  |  |  |  |  | 73,5% | 70,07%-76,74% |  |
|  |  |  |  |  |  | 68,95% | 65,38%-72,35% |  |
|  |  |  |  |  |  | 68,23% | 64,65%-71,67% |  |
|  |  |  |  |  |  | 57,83% | 54,08%-61,52% | 0,0008 |

### Naivni Bayesov klasifikator

Za izradu Bayesovog klasifikatora korištena je funkcija naiveBayes iz paketa e1071. Zanimljivo je da je ovaj klasifikator davao najlošije rezultate za većinu korištenih ulaznih varijabli, ali pri korištenju samo PPM1 i PPM2 Bayesov klasifikator postiže preciznost od 74,4% što je jednako kao kod modela stroja potpornih vektora. Korištenjem svih ulaznih varijabli model postiže preciznost od 69,1%. Matrica konfuzije najboljeg izgrađenog modela dana je u nastavku (Tablica 6.9).

**Tablica 6.9:** Matrica zabune za naivni Bayesov klasifikator s ulaznim varijablama PPM1 i PPM2

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 241 | 97 |
| 1 | 83 | 281 |

**Tablica 6.10:** Prikaz rezultata modela naivnog Bayesovog klasifikatora s obzirom na ulazne varijable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OPS | OPR | PPM | Podloge | IBL | PNV | Preciznost | 95% i.p. | p-vrijednost |
|  |  |  |  |  |  | 74,36% | 70,96%-77,55% |  |
|  |  |  |  |  |  | 69,09% | 65,52%-72,49% |  |
|  |  |  |  |  |  | 65,95% | 62,32%-69,46% |  |
|  |  |  |  |  |  | 64,39% | 60,72%-67,93% |  |
|  |  |  |  |  |  | 57,12% | 53,37%-60,82% | 0,0028 |

### Umjetna neuronska mreža

Za razvoj neuronskih mreža u R-u postoji nekoliko javno dostupnih paketa. Mi ćemo za potrebe ovog rada koristiti paket h2o*[[15]](#footnote-15)* 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[[16]](#footnote-16). Neki od najzanimljivijih hiperparametara koji su korišteni u 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 (Tablica 6.11 i Tablica 6.12) prikazani su najbolji dobiveni modeli za svaku prikazanu kombinaciju ulaznih varijabli te matrica zabune najboljeg izgrađenog modela.

Za većinu prikazanih modela nisu korištene pretjerano komplicirane neuronske mreže koje dugo treba trenirati iz jednostavnog razloga što je vrlo lako dolazilo do pretreniranosti. Najbolji rezultat postignut je korištenjem aktivacijske funkcije tangens hiperbolni, dimenzija skrivenog sloja (200, 200) sa stopom učenja od 0.005 te 10 epoha. Osim odlične preciznosti od 75,07%, model je postigao osjetljivost od 74,73% te specifičnost od 75,44%.

**Tablica 6.11:** Matrica zabune za naivni Bayesov klasifikator s ulaznim varijablama PPM1, PPM2, IBL1, IBL2, PNV1, PNV2

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | 255 | 83 |
| 1 | 92 | 272 |

**Tablica 6.12:** Prikaz rezultata modela neuronske mreže s obzirom na ulazne varijable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OPS | OPR | PPM | Podloge | IBL | PNV | Preciznost | 95% i.p. | p-vrijednost |
|  |  |  |  |  |  | 75,07% | 71,7%-78,23% |  |
|  |  |  |  |  |  | 72,22% | 68,75%-75,51% |  |
|  |  |  |  |  |  | 71,65% | 68,16%-74,96% |  |
|  |  |  |  |  |  | 68,52% | 64,94%-71,94% |  |
|  |  |  |  |  |  | 67,95% | 64,36%-71,39% |  |

# Zaključak

# Literatura

1. Barnett, T., & Clarke, S. R. (2005). Combining player statistics to predict outcomes of tennis matches. IMA Journal of Management Mathematics, 16 (2), 113-120.
2. Jackson, David, and Krzysztof Mosurski. "Heavy defeats in tennis: Psychological momentum or random effect?." Chance 10.2 (1997): 27-34.
3. Klaassen, Franc JGM, and Jan R. Magnus. "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): 500-509.
4. Knottenbelt, William J., Demetris Spanias, and Agnieszka M. Madurska. "A common-opponent stochastic model for predicting the outcome of professional tennis matches." Computers & Mathematics with Applications 64.12 (2012): 3820-3827.
5. Sipko, Michal, and William Knottenbelt. "Machine learning for the prediction of professional tennis matches." MEng computing-final year project, Imperial College London  (2015).
6. MacPhee, I. M., Jonathan Rougier, and G. H. Pollard. "Server advantage in tennis matches." Journal of applied probability  (2004): 1182-1186.
7. Barnett, T., A. Brown, and S. Clarke. "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., A. Brown. “The mathematics of tennis”, Strategic Games (2012)

# Sažetak

Todo

# Summary

Todo

# Skraćenice

# Privitak

1. Preuzeto s https://www.hranalytics101.com/how-to-assess-model-accuracy-the-basics [↑](#footnote-ref-1)
2. Preuzeto s https://en.wikipedia.org [↑](#footnote-ref-2)
3. Preuzeto sa http://deeconometrist.nl/predicting-outcomes-of-tennis-matches [↑](#footnote-ref-3)
4. Preuzeto s https://www.fer.unizg.hr/predmet/su [↑](#footnote-ref-4)
5. Preuzeto sa https://en.wikipedia.org/wiki/Logistic\_function [↑](#footnote-ref-5)
6. Preuzeto s https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm [↑](#footnote-ref-6)
7. Preuzeto s https://www.analyticsvidhya.com/blog/2021/05/support-vector-machines/ [↑](#footnote-ref-7)
8. Preuzeto s https://www.fer.unizg.hr/predmet/su [↑](#footnote-ref-8)
9. Preuzeto s https://www.fer.unizg.hr [↑](#footnote-ref-9)
10. Preuzeto s https://www.pinterest.com [↑](#footnote-ref-10)
11. Preuzeto s https://deepai.org/machine-learning-glossary-and-terms/activation-function [↑](#footnote-ref-11)
12. https://cran.r-project.org/web/packages/caret/ [↑](#footnote-ref-12)
13. https://www.rdocumentation.org/packages/stats [↑](#footnote-ref-13)
14. https://cran.r-project.org/web/packages/e1071 [↑](#footnote-ref-14)
15. https://cran.r-project.org/web/packages/h2o [↑](#footnote-ref-15)
16. Preuzeto s https://towardsdatascience.com/neural-networks-parameters-hyperparameters-and-optimization-strategies [↑](#footnote-ref-16)