SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Komić Daria

Rusovan Karlo

Inteligentni sustav za predviđanje rezultata u videoigri League of Legends

Istraživački projekt

Varaždin, 2023.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Daria Komić

Karlo Rusovan

Studij: Baze podataka i baze znanja

Inteligentni sustav za predviđanje rezultata u videoigri League of Legends

ISTRAŽIVAČKI PROJEKT

Mentori:

Prof. dr. sc. Božidar Kliček

Izv. prof. dr. sc. Dijana Oreški

Varaždin, listopad 2023.

Sadržaj

Sadržaj **2**
Uvod **3**
Opis metodologije (CRISP DM) **4**
Razumijevanje domene **5**
Pregled prethodnih istraživanja 8
Ciljevi istraživanja 12
Razumijevanje podataka **13**
Opis atributa 13
Distribucije vrijednosti atributa i kvaliteta podataka 14
Priprema podataka **27**
Selekcija varijabli 27
Uklanjanje outlier vrijednosti 31
Imputacija nedostajućih vrijednosti 32
Izrada validacijskog stupca 32
Modeliranje podataka **33**
k-najbližih susjeda 33
Optimizacija modela 35
Evaluacija najboljeg modela 37
Neuronska mreža 39
Evaluacija najboljeg modela 40
Naivni Bayesov klasifikator 42
Evaluacija najboljeg modela 42
Interpretacija rezultata i primjena najboljeg modela **45**
Literatura **49**
Popis slika **50**

Uvod

Videoigra League of Legends je jedan od najpopularnijih e-sportova danas. Ovaj istraživački rad fokusira se na prepoznavanje elemenata igre koji najviše utječu na pozitivan uspjeh, odnosno pobjedu. S obzirom da igra postoji već 13 godina, današnji igrači na najvišoj razini imaju dovoljne sposobnosti da igraju blizu savršenstva na mikro razini. Iz tog razloga potrebno je gledati igru na makro razini te gledati na elemente koji su izvan kontrole jednog igrača, a koji utječu na pobjedu. Koristiti će se podaci iz 10 000 mečeva odigranih na diamond razini, odnosno razini kojoj pripada 2.7% najboljih igrača. Podaci će se uzimati u trenutku 10. minute meča te je cilj na temelju tog stanja predvidjeti pobjednika. Time će se igračima dati smjernice o tome kojim elementima je potrebno dati najviše pažnje na početku meča kako bi se postiglo stanje koje ima najveću šansu da dovodi do pobjede.

Cilj ovog istraživanja je razviti inteligentni sustav koji će točno i pouzdano identificirati one karakteristike stanja meča koje imaju najveći utjecaj na krajnji ishod, odnosno sustav koji će predviđati krajnji ishod na temelju stanja meča u 10. minuti.

U sklopu rada najprije će se dati pregled domene gdje će se objasniti na koji način videoigra League of Legends funkcionira i gdje će se opisati pojmovi potrebni za daljnje razumijevanje rada. Zatim će se napraviti priprema podataka normalizacijom i transformacijom varijabli, a nakon toga, razviti će se prediktivni modeli primjenom algoritama strojnog učenja. Dobiveni rezultati zatim će se vrednovati kako bi se utvrdilo koji je model najkvalitetniji temeljem definiranih metrika kvalitete. Primjenom analize osjetljivosti će se interpretirati rezultati i dati će se smjernice za najbolji način postizanja stanja igre koje ima najveću vjerojatnost pozitivnog ishoda, odnosno pobjede.

Opis metodologije (CRISP DM)

U ovom projektu koristiti će se CRISP-DM metodologija. CRISP-DM (Cross Industry Standard Process for Data Mining) definira procesni model koji daje okvir za provođenje projekata rudarenja podataka. CRISP-DM je neovisan o domeni u kojoj se koristi i o tehnologiji koja se koristi te ima cilj ubrzati projekte rudarenja podataka, učiniti jeftinijima, pouzdanijima i lakšima za upravljanje [1]. Organiziran je kao hijerarhija četiri razine apstrakcije. Pri tome su prve dvije razine neovisne o primjeni i tehnologiji, a te prve dvije razine su faze i općeniti zadaci. Druge dvije razine su specijalizirani zadaci i instance procesa te su njihove primjene direktno vezane uz specifični problem i tehnologiju koja se koristi [1]. Sama struktura ovog projekta temelji se upravo na najvišoj razini apstrakcije CRISP-DM metodologije, odnosno fazama. U nastavku će se dati općeniti pregled svake faze koju će ovaj projekt pratiti:

- Business understanding - inicijalna faza CRISP-DM. Cilj je razumijevanje ciljeva i zahtjeva projekta te definiranje problema rudarenja podataka iz toga [1].
- Data understanding - počinje sa prikupljanjem podataka, nakon čega se provode koraci upoznavanja s podacima, identifikacije problema u kvaliteti podataka te dolaženja do nekih prvih zaključaka i hipoteza koji su vidljivi iz sirovih podataka [1].
- Data preparation - sastoji se od aktivnosti koje pretvaraju sirove podatke u skup podataka koji će se koristiti u modeliranju. Neke od tih aktivnosti su odabir relevantnih atributa i slogova, čišćenje podataka te transformacija podataka kako bi bili primjenjivi u alatima za modeliranje [1].
- Modeling - u ovoj fazi se odabiru i primjenjuju algoritmi strojnog učenja te se kalibriraju njihovi parametri. [1]
- Evaluation - evaluacija izgrađenih modela. Odabiru se oni modeli za koje se smatra da su visoko kvalitetni iz perspektive analize podataka te se detaljnije evaluiraju kako bi se moglo pouzdano zaključiti da ostvaruju poslovne ciljeve [1].
- Deployment - organiziranje i prezentiranje stečenog znanja na način da ga kupac može koristiti [1].

Razumijevanje domene

Razumijevanje domene je prva faza metodologije CRISP-DM gdje je potrebno razumjeti inicijalni cilj i poslovne potrebe projekta iz poslovne perspektive. Korištenjem tog znanja moguće je definirati problem i plan kako ostvariti pojedine ciljeve.

Prije nego što se konkretizira poslovni cilj problema kojeg postavljamo, dati će se opći pregled videoigre League of Legends te osnovnih koncepata i izraza koji će se koristiti u nastavku ovog rada. League of Legends je besplatna MOBA (eng. multiplayer online battle arena) videoigra. Stil igre je PvP (player versus player) gdje postoje dva tima po 5 igrača koji brane svoj dio mape i gdje svaki od 5 igrača igra na definiranoj poziciji. Cilj igre je jednostavan, doći do suprotnog dijela mape i uništiti bazu protivničkog tima, pri čemu se pobjeda označava uništavanjem protivničkog "Nexusa" [2]. Svaki od igrača bira jednog *championa* iz skupa od 165 gdje svaki od njih ima svoje originalne sposobnosti i stil igre. Srž videoigre leži u natjecanju igrača koje koristi sustav formiranja timova temeljem njihovog ranga na ljestvici. Porastom popularnosti videoigara i natjecateljskog karaktera League of Legends, ona je postala najveći svjetski esport sa natjecateljskom scenom koja se sastoji od više regionalnih liga i internacionalnih natjecanja od kojih je najveće godišnje natjecanje League of Legends World Championship [3]. 2022. godine finale natjecanja je istovremeno gledalo preko 5 milijuna gledatelja.

League of Legends je videoigra sa jednostavnim ciljem, lako ju je naučiti, ali teško savladati. Od poznavanja svakog *championa*, do strateških pozicioniranja na mapi i igranja oko svojeg tima, puno varijabli utječe na ishod meča. U nastavku biti će prikazani najbitniji dijelovi igre za bolje razumijevanje njihovih utjecaja kasnije.

Mapa, *championi* i uloge

Kako je već poznato, timovi od 5 igrača igraju na mapi koja je poznata pod nazivom "Summoner's Rift". Cilj igre je ući u neprijateljsku bazu i srušiti "Nexus" (Slika 1.). Na slici su vidljive i "staze" odnosno "*lanes*" gdje svaki od igrača igra prema zadanoj ulozi. Postoji 5 uloga u timu: *support, bottom, midlane, jungle i toplane.* Od tih, jedino *support* i *bottom* igrači igraju zajedno u *bottom laneu*, dok *jungler* ne igra u *laneu* već na dijelovima mape pod nazivom *jungle* uzima neutralne *minione* (objašnjeno u sljedećem odlomku) i pomaže ostatku tima. Određeni *championi* su bolji za određene uloge od ostalih. Igrači sa više znanja o igri i sinergiji *championa* mogu odabrati one koji prema njihovom mišljenju stvaraju bolji tim od neprijateljskog te time ostvariti prednost. Takva sinergija i pametno biranje *championa*

dosta je popularno u esports sceni, jer postoji tzv. *counterpick* odnosno biranje *championa* koji daje igraču prednost nad njegovim protivnikom.

Slika 1. Summoner's Rift mapa [10]

Objectives

Nakon formiranja oba tima, igrači zauzimaju svoje pozicije i igra kreće. Osim ključnog cilja, postoje manji ciljevi koje igrači pokušavaju ispuniti tokom meča. Od prvih ciljeva imamo *farming* odnosno ubijanje *miniona* u *laneu* koji se pojavljuju u 1. minuti meča i nastavljaju se pojavljivati svakih 30 sekundi do kraja meča. Ubijanjem tih *miniona* svaki igrač skuplja *gold* i *experience* i broj *miniona* koje je ubio u meču ima skraćenicu *CS* odnosno *Creep Score. Gold je* novac kojim igrač kupuje određene *iteme* koji poboljšavaju njegovog *championa* te on kroz vrijeme postaje jači. *Experience* služi za *level up championa* (maksimalni level championa je 18) te sa svakim *level upom* jedan od *abilityja* (njegova vještina, svaki champion ima najčešće 4 *abilityja*, rjeđe 3) postaje jači.

Još jedan način dobivanja *golda* i *experienca* je ubijanje protivnika. Time igrač može dobiti više *golda* i *experienca* nego ubijanjem *miniona*, ali postoji rizik da protivnik ubije igrača. U gornjem desnom kutu meča, moguće je vidjeti statistiku zvanu *KDA* koja predstavlja: *kill* (koliko je igrač ubio protivnika odnosno *enemy championa*), *deaths* (koliko je puta umro u meču) i *assists* (koliko je puta asistirao suigraču u ubijanju protivnika. Rušenjem obrambenih tornjeva (*turreta*) također se dobiva *gold* i naravno igrač je bliže bazi protivnika. Bitne zgrade prije *Nexusa* koje timovi žele uništiti su *inhibitori*, koji nakon uništenja omogućuju pojavu *super miniona* za tim koji je uništio protivnički *inhibitor*, koji se pojavljuju sve dok je neki od *inhibitora* uništen. Od bitnijih ključnih *objectiva* imamo čudovište *Herald*, čudovište *Baron* i *drakeove.* Od početka meča u *Baron pitu* (slika 2.) je dostupan *Herald* čijim ubijanjem ga je moguće "pokupiti" (zamjenjuje *ward slot* u sučelju) i moguće ga je pustiti u bilo kojem *laneu* gdje se on "zabija" u obrambene tornjeve i omogućava brže rušenje. U 20. minuti *Herald* prestaje biti dostupan i pojavljuje se *Baron*. Ukoliko tim ubije *Barona*, daje *buff* kojim njihovi *minioni* postaju jači i moguće je brže uništavati protivničke *turrete* i bazu. *Drakeovi* omogućuju dobivanje *Dragon soul buffa* ukoliko jedan od timova ubije 4 *drakea.* Od *soulova* postoje *infernal, ocean, hextech, cloud, mountain* i *chemtech.* Šest minuta nakon osvajanja *Dragon soula* na mjestu se pojavi specijalni *drake*, *Elder drake*.

Items

Itemi su poboljšanja koja svaki od igrača može kupiti za svojeg *championa* za *gold* koji dobivaju od ispunjavanja različitih *objectiva* (navedenih u prošlom odjeljku) na mapi. *Itemi* se kupuju iz *shopa* i svaki od *championa* može imati maksimalno 6 *itema* i jedan *trinket* odnosno *ward* koji kada se postavi na mapi daje vidljivost u tom dijelu mape. Postavljanjem *trinketa* ili *warda* na mapi, odnosno uništavanjem protivničkog *warda,* povećava igraču statistiku *vision score*. Sučelje *shopa* vidi se na slici 2. te vidimo da ispod trake za pretraživanje imamo ikone kategorija koje odgovaraju ulozi *championa* (vidi odjeljak Mapa, *championi* i uloge).

Slika 2. Sučelje shopa [11]

Pregled prethodnih istraživanja

Na temu League Of Legends već je napravljeno nekoliko projekata korištenja strojnog učenja za predikciju ishoda meča. Jedan od takvih projekata je opisan u znanstvenom članku “Using Machine Learning to Predict Game Outcomes Based on Player-Champion Experience in League of Legends” [3]. Cilj tog istraživanja je bio predikcija ishoda meča na temelju razine iskustva pojedinog igrača sa championom kojega igra. Glavna propozicija je to da igrači koji igraju champione koje su u potpunosti savladali i sa kojima imaju puno iskustva imaju mnogo veće šanse za pobjedu od igrača koji igraju champione sa kojima imaju manje iskustva. Podaci su se sastojali od 4000 mečeva, pri čemu se 80% koristilo za treniranje a 20% za testiranje. Atributi skupa podataka bili su:

Champion mastery points, odnosno bodovi koje igrač dobiva kada igra mečeve s određenim championom. Pri tome će igrači koji više igraju nekog championa imati više *mastery* bodova za tog championa.

Player-champion win rate, postotak pobjeda od odigranih mečeva tog igrača sa tim championom.

Algoritmi strojnog učenja koji su se koristili su *Support Vector Classifiers* (SVC), k-najbližih susjeda, *Random Forest trees*, *Gradient Boosting* i *Deep Neural Networks* (DNN). Najveću srednju točnost imali su modeli *DNN* (75.1% +/- 1.2%) *i Gradient Boosting* (75.4% +/- 1.2%), pri čemu su svi modeli imali srednju točnost veću od 72.7% [3]. Zaključak je kako se već prije početka meča ishod može predvidjeti sa točnosti od preko 75% samo na temelju odabira championa.

Idući članak, “Machine Learning Methods for Predicting League of Legends Game Outcome” je fokusiran na predikciju ishoda u profesionalnim mečevima [4]. S obzirom da profesionalni igrači znaju igrati sve relevantne champione, pretpostavka je kako atributi iz prethodnog članka ovdje neće biti prediktori ishoda meča. Korišten je skup podataka od 7583 profesionalna meča te su neki od atributa pobjednik meča, godina, sezona, regionalna liga, timovi, champion na svakoj od pet pozicija oba tima, te igrači na svakoj poziciji u oba tima. Algoritmi strojnog učenja koji su dali najbolju srednju točnost su Naivni Bayesov klasifikator (68,96%) te *Extreme Gradient Boosting* (68,82%). K-najbližih susjeda je isto tako dalo komparativno zadovoljavajuću točnost od 66,58%. Pri tome, atributi koji su bili najbolji prediktori ishoda su bili postotak pobjeda pojedinog igrača protiv igrača na istoj poziciji u

protivničkom timu, te postotak pobjeda pojedinog igrača protiv championa na istoj poziciji u protivničkom timu [4].

Prethodna dva članka su imala postavljeni cilj predikcije ishoda meča prije nego što meč uopće počne, što je i razlog relativno niske točnosti predikcije (75% za obične igrače, 68% za profesionalne igrače). Druga kategorija istraživanja na ovu temu su istraživanja koja žele predvidjeti ishod meča nakon što je meč već počeo. Ovdje je važno napomenuti kako prosječni meč traje od 30 do 45 minuta. Jedno takvo istraživanje, "How We Trained a Machine to Predict the Winning Team in League of Legends" za cilj ima predvidjeti ishod profesionalnog meča u različitim trenucima tog meča [5]. Korišten je skup podataka od 3802 profesionalnih mečeva te su neki od atributa: timovi, sezone, pobjednik, *gold difference* (razlika u *goldu* koje je pojedini tim prikupio, a prikuplja se pomoću ubijanja protivnika i rušenja dijelova baze), *kills* (koliko puta je tim ubio protivnike), *towers* (koliko je tim srušio tornjeva baze protivničkog tima) itd. Algoritmi strojnog učenja koji su se koristili su Stablo odlučivanja, *Logistic Regression* i različita stabla (*Gradient Boosted, Random forests…*). Postignuta je srednja točnost od 75% za slučaj gdje se predikcija izvršava sa informacijama o timovima i odabranim championima, prije početka meča, te točnost od isto tako 75% nakon odigranih 15 minuta meča [5]. Zaključak je kako prvih 15 minuta meča nema gotovo nikakav utjecaj na točnost predikcije, te kako su najbitniji prediktori vezani uz timove koji igraju te uz champione koje su odabrali.

Osim predikcije ishoda meča na temelju odabranih championa, efikasnosti igrača u igranju pojedinog championa te prvih 15 minuta meča, neka istraživanja fokusirana su na ocjenjivanje igrača temeljem raznih karakteristika njihovog stila igranja. Jedno takvo istraživanje je "Machine Learning based Analysis of e-Sports Player Performances in LoL for Winning Prediction based on Player Roles and Performances" [6]. Cilj ovog istraživanja je dodavanje ocjene igraču temeljem atributa poput prosječnog broja ubijenih *miniona*, protivničkih *championa*, pozicije koju igra i sličnih u posljednjih X mečeva. Cilj toga je izvršenje predikcije pobjednika meča na način da se gleda aritmetička sredina dodijeljenih ocjena igračima za svaki od timova. Model strojnog učenja koji je odabran kao najbolji je *Gradient Boost*, te se postigla srednja točnost od 86% [6]. S obzirom da je to najveća postignuta srednja točnost od obrađenih istraživanja, može se zaključiti kako je relativna snaga igrača bitnija od championa koje igraju te od njihove vještine na pojedinom championu.

Naposljetku, neka istraživanja idu i korak dalje. Osim predikcije pobjednika, analiziraju se razne varijable pobjedničkog tima kako bi se kreirao model koji preporučuje akcije koje igrač treba poduzeti kako bi mu se šansa za pobjedu povećala. Jedno takvo istraživanje je “A machine learning approach for recommending items in League of Legends” [7]. Ovo istraživanje definira vrijednost pojedinog *itema* (objašnjeno u razumijevanju domene) na temelju analiza mečeva i *itema* koje su kupovali igrači pobjedničkih timova. Na temelju toga, korištenjem Neuralnih mreža kreiran je model koji igraču preporučuje koje *iteme* kupiti ovisno o trenutnom stanju meča. Isto tako, time je moguće predvidjeti pobjednika meča na temelju *itema* koje su igrači kupili do nekog trenutka meča. Postignuta točnost predikcije na temelju kupljenih *itema* u trenutku *mid game* (15. minuta meča) je 63.35% te točnost od 70% na kraju meča.

Na temelju analiziranih istraživanja zaključuje se kako se kreirani inteligentni sustavi mogu podijeliti u tri kategorije. To su predikcija pobjednika na temelju parametara prije početka meča (relativna snaga igrača, odabrani *championi*), predikcija pobjednika na temelju stanja meča u određenoj minuti (predikcija u 10. minuti meča), te predikcija pobjednika na temelju specifičnih varijabli (kupljenih *itema)* kako bi se mogao generirati model za preporuku upravljanja tim varijablama u svrhu povećanja šanse za pobjedu. Korišteni skupovi podataka dijele se na SoloQueue (mečevi koji igraju obični igrači kako bi se penjali na rang ljestvici) te na profesionalne mečeve. Zaključeno je kako je točnost predikcije pobjednika prije početka meča za profesionalne mečeve manja nego za SoloQueue mečeve, s obzirom da profesionalni mečevi uključuju mnogo više varijabli poput snage tima, forme igrača, sinergije itd. Isto tako, zaključeno je kako prvih 15. minuta meča gotovo uopće ne doprinosi točnosti predikcije pobjednika za profesionalne mečeve, te kako je model sa najvećom prosječnom točnosti onaj koji koristi razne karakteristike igrača kako bi kreirao ocjenu na temelju koje radi postotak šanse pobjede tima u kojem je taj igrač. Nijedan osim tog posljednjeg modela nije uspio postići predikciju pobjednika sa srednjom točnosti većom od 75% . Algoritmi strojnog učenja koji su se najviše koristili bili su Gradient Boost, Stablo odlučivanja, Neuronske mreže, k-najbližih susjeda te naivni Bayesov klasifikator kao i *Random Forest*.

Ciljevi istraživanja

Osim navedenih članaka, jedna od inspiracija za ovo istraživanje je i predikcijski model koji je izradila kompanija Demacia Data u suradnji sa Amazon Web Services (AWS) [8]. Taj predikcijski model koristi se na svjetskom prvenstvu 2023. koje se odvija tijekom listopada i studenog. Cilj ovog istraživanja biti će predikcija pobjednika na temelju stanja meča u 10. minuti. Iako je već napravljeno istraživanje na sličnu temu (predikcija pobjednika profesionalnog meča u 15. minuti), smatramo kako je to istraživanje napravilo ključnu pogrešku. Naime, u profesionalnim mečevima rane faze meča imaju mnogo manju varijancu s obzirom da su igrači mnogo bolje organizirani te su glavni prediktori faktori prije početka meča (snaga tima, forma igrača, odabrani championi), a ne faktori unutar meča. Iz tog razloga uzimamo SoloQueue mečeve (mečeve običnih igrača). Time je cilj napraviti popis smjernica koje su relevantne za obične igrače, a koje ukazuju na one elemente stanja meča na koje treba staviti najviše pozornosti u prvih 10 minuta kako bi se osigurala što veća šansa za pobjedu. Odlučili smo raditi predikciju na temelju stanja u 10. minuti umjesto predikcije na temelju stanja prije meča zato što već postoji mnogo izvora koji ocjenjuju igrače i koji govore koje je *championе* najbolje odabrati u kojim situacijama. S druge strane, za ono što mi pokušavamo napraviti, a to je dati upute kako najbolje odigrati rane faze meča, nema mnogo lako dostupnih izvora osim plaćanja treninga s profesionalnim trenerima.

Razumijevanje podataka

Skup podataka za analizu preuzet je sa Kaggle.com [9]. Sastoji se od približno 10 tisuća mečeva (redova) i 40 atributa (stupaca). Podaci opisuju stanje u 10. minuti *soloqueue* meča (mečeva običnih igrača) u *Diamond* i *low Master ranku* (oko 3% najboljih igrača). Podaci koji su učitani na Kaggle.com povučeni su direktno sa Riot API (Riot Games je organizacija koja drži prava nad videoigrom) i ne sadrže prazne vrijednosti.

Od spomenutih 40 atributa, 20 njih se odnosi na crveni tim i 20 na plavi tim. Sukladno tome biti će objašnjeno 20 njih zbog ponavljanja za svaki tim. Kako je za svaki stupac tip atributa cijeli ili decimalni broj (diskretni ili kontinuirani numerički podatak), neće se posebno navoditi. Jedini atributi koji se mogu klasificirati kao nominalni su BlueWins te FirstBlood, s obzirom da samo označavaju koji je tim pobijedio, odnosno koji je dobio prvi *kill*.

Opis atributa

- BlueWins - označava koji tim je pobijedio (0 znači da je pobijedio crveni, a 1 da je pobijedio plavi tim), ovaj stupac je time i ciljni stupac predikcije
- WardsPlaced - ukupan broj postavljenih *wardova* od strane svih članova tima
- WardsDestroyed - ukupan broj uništenih protivničkih *wardova* od strane svih članova tima
- FirstBlood - tko je dobio prvi *kill* u meču
- Kills - broj koliko su puta članovi tima ubili članove protivničkog tima
- Deaths - broj koliko su puta članovi tima umrli
- Assists - broj koliko su puta članovi tima pomogli u ubijanju članova protivničkog tima
- EliteMonsters - broj ubijenih *elite* čudovišta koji daju veći bonus u *goldu* i *experience* (*drakes* i *Heralds*)
- Dragons - ukupan broj *drakeova* koje je jedan od timova ubio (maksimalna vrijednost 1 zato što se do 10. minute može pojaviti samo jedan *dragon*)
- Heralds - broj *Heralda* koje je jedan od timova ubio (maksimalna vrijednost 1 zato što se do 10. minute može pojaviti samo jedan *herald)*.
- TowersDestroyed - broj obrambenih tornjeva (*turreta)* koje je jedan od timova srušio
- AvgLevel - prosječni *level* championa svih članova tima (ovisi o količini *experience)*
- TotalExperience - sveukupni broj *experience* bodova prikupljenih od strane svih članova tima

- TotalMinionsKilled - sveukupni broj ubijenih *miniona* svih članova tima
- TotalJungleMinionsKilled - sveukupni broj ubijenih *jungle miniona* svih članova tima
- GoldDiff - razlika u *goldu* s obzirom na protivnički tim
- ExperienceDiff - razlika u *experienceu* s obzirom na protivnički tim
- CSPerMin - sveukupni *Creep Score* (vidi odjeljak *Objectives*) po minuti svih članova tima
- GoldPerMin - sveukupni prikupljeni *gold* po minuti svih članova tima

Distribucije vrijednosti atributa i kvaliteta podataka

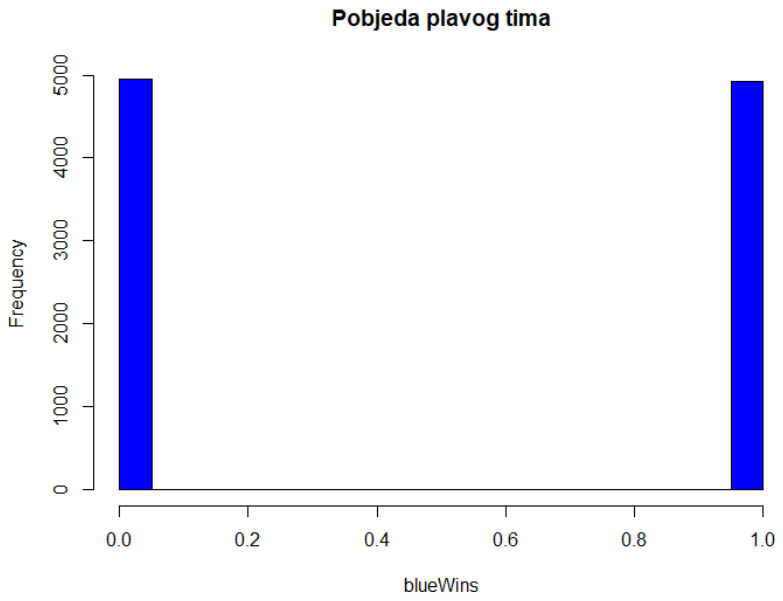
Za prikaz distribucija vrijednosti atributa koristiti će se programski jezik R i programski paket RStudio. Za provjeru outliera i nedostajućih vrijednosti korišten je alat JMP. Za traženje outliera korištena je opcija Robust Fit Outliers sa Huber funkcijom i K sigma vrijednosti jednakom 4.

Column	Number Missing
gameId	0
blueWins	0
blueWardsPlaced	0
blueWardsDestroyed	0
blueFirstBlood	0
blueKills	0
blueDeaths	0
blueAssists	0
blueEliteMonsters	0
blueDragons	0
blueHeralds	0
blueTowersDestroyed	0
blueTotalGold	0
blueAvgLevel	0
blueTotalExperience	0
blueTotalMinionsKilled	0
blueTotalJungleMinionsKilled	0
blueGoldDiff	0
blueExperienceDiff	0
blueCSPerMin	0
blueGoldPerMin	0
redWardsPlaced	0
redWardsDestroyed	0
redFirstBlood	0
redKills	0
redDeaths	0
redAssists	0
redEliteMonsters	0
redDragons	0
redHeralds	0
redTowersDestroyed	0
redTotalGold	0
redAvgLevel	0
redTotalExperience	0
redTotalMinionsKilled	0
redTotalJungleMinionsKilled	0
redGoldDiff	0
redExperienceDiff	0
redCSPerMin	0
redGoldPerMin	0

Column	Huber Center	Huber Spread	Huber N Outliers
gameId	4,5011e+9	26037791	36
blueWins	0,4990384	0,527078	0
blueWardsPlaced	19,167649	7,8775862	556
blueWardsDestroyed	2,6879827	1,7220598	114
blueFirstBlood	0,5048082	0,5270546	0
blueKills	6,1263821	3,0163854	3
blueDeaths	6,088336	2,9528488	5
blueAssists	6,5195887	3,9284854	27
blueEliteMonsters	0,5340279	0,6221597	0
blueDragons	0,36198	0,5065999	0
blueHeralds	0,1879745	0,4118508	0
blueTowersDestroyed	1,4926e-6	3,785e-6	464
blueTotalGold	16476,976	1527,8147	10
blueAvgLevel	6,9189792	0,3007047	11
blueTotalExperience	17937,533	1197,032	9
blueTotalMinionsKilled	216,91376	22,062201	3
blueTotalJungleMinionsKilled	50,472666	9,9018868	7
blueGoldDiff	9,8547426	2456,2814	3
blueExperienceDiff	-36,45552	1918,0421	7
blueCSPerMin	21,691376	2,2062201	3
blueGoldPerMin	1647,6976	152,78147	10
redWardsPlaced	19,349377	8,2615138	513
redWardsDestroyed	2,5912816	1,6882438	110
redFirstBlood	0,4951918	0,5270546	0
redKills	6,088336	2,9528488	5
redDeaths	6,1263821	3,0163854	3
redAssists	6,543138	3,9536736	18
redEliteMonsters	0,5597179	0,6293308	0
redDragons	0,4130985	0,519057	0
redHeralds	0,0084565	0,02066	1581
redTowersDestroyed	1,7303e-6	4,4148e-6	396
redTotalGold	16467,717	1495,9321	3
redAvgLevel	6,9290997	0,2993674	13
redTotalExperience	17974,009	1187,7551	9
redTotalMinionsKilled	217,58717	22,059036	5
redTotalJungleMinionsKilled	51,225294	9,9907314	9
redGoldDiff	-9,854743	2456,2814	3
redExperienceDiff	36,455516	1918,0421	7
redCSPerMin	21,758717	2,2059036	5
redGoldPerMin	1646,7717	149,59321	3

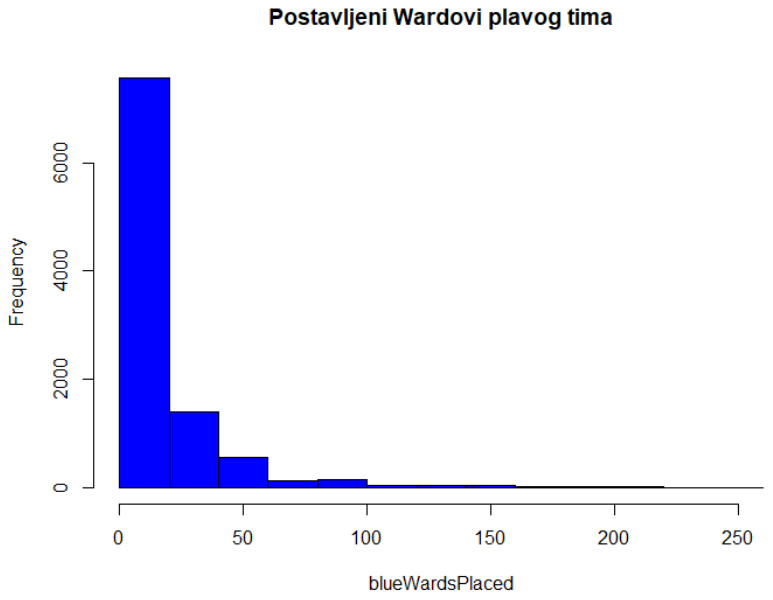
Slika 3. Nedostajuće i outlier vrijednosti

Vidljivo je kako skup podataka nema nedostajućih vrijednosti, te kako je broj outliera za većinu varijabli relativno mali: za 29 od 40 varijabli broj outliera je manji od 10, a skup podataka ima 10 000 zapisa. Ipak, neke varijable imaju veći broj outlier vrijednosti, a to će se pobliže istražiti gledanjem njihovih grafova distribucije.



Slika 4. Distribucija vrijednosti varijable blueWins

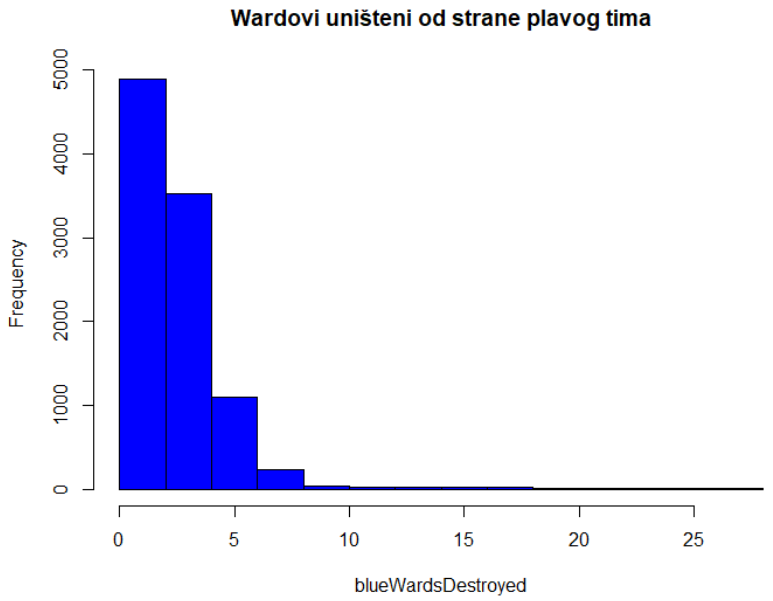
S obzirom da je ovo varijabla koja može poprimiti vrijednost 0 ukoliko je pobjedio crveni tim ili vrijednost 1 ukoliko je pobjedio plavi tim (u League of Legends ne postoji neriješeni ishod), zaključujemo kako boja tima, i time strana mape na kojoj se nalazi njihova baza, ne utječe na vjerojatnost pobjede.



Slika 5. Distribucija vrijednosti varijable blueWardsPlaced

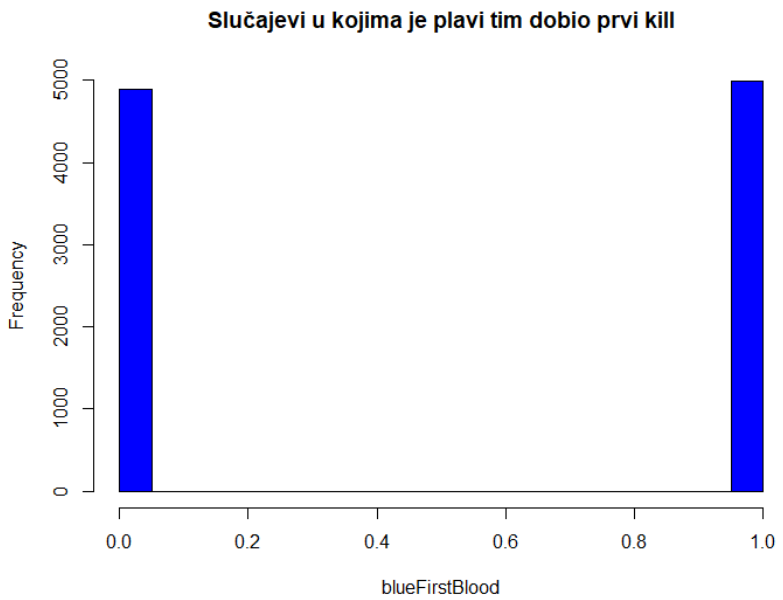
Sa histograma je vidljivo da ova varijabla ima relativno veliki broj outliera. Naime, očekivani broj postavljenih *wardova* od strane tima u prvih 10 minuta je 15-20 (medijan je 16). Prema tome, vrijednost aritmetičke sredine od 22.29 ukazuje na to kako je distribucija

podataka ove varijable pozitivno asimetrična. To je objašnjivo s obzirom da je česta taktika odustajanja ukoliko tim ima loš početak meča, kupnja velikog broja wardova i njihovo postavljanje.



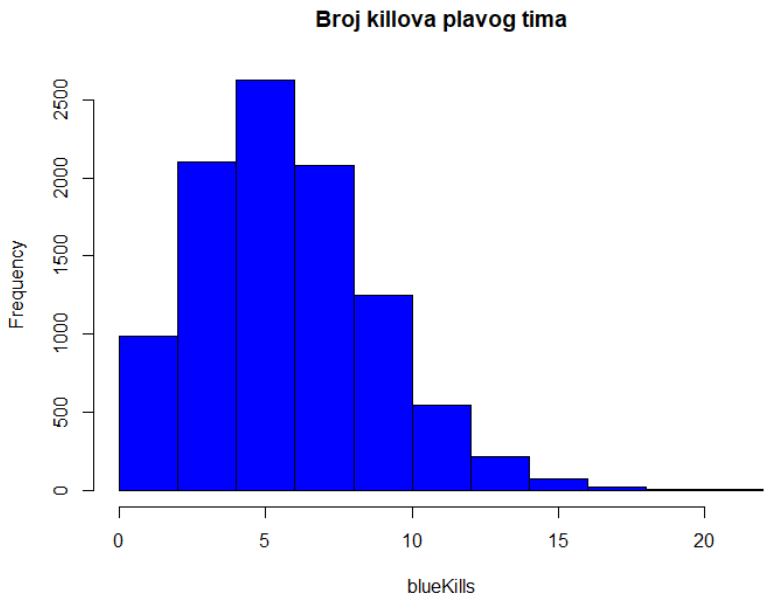
Slika 6. Distribucija vrijednosti varijable blueWardsDestroyed

Varijabla blueWardsDestoryed ima sličnu distribuciju kao i prethodna varijabla, sa nešto manje outliera s obzirom da je wardove teže uništiti nego postaviti.

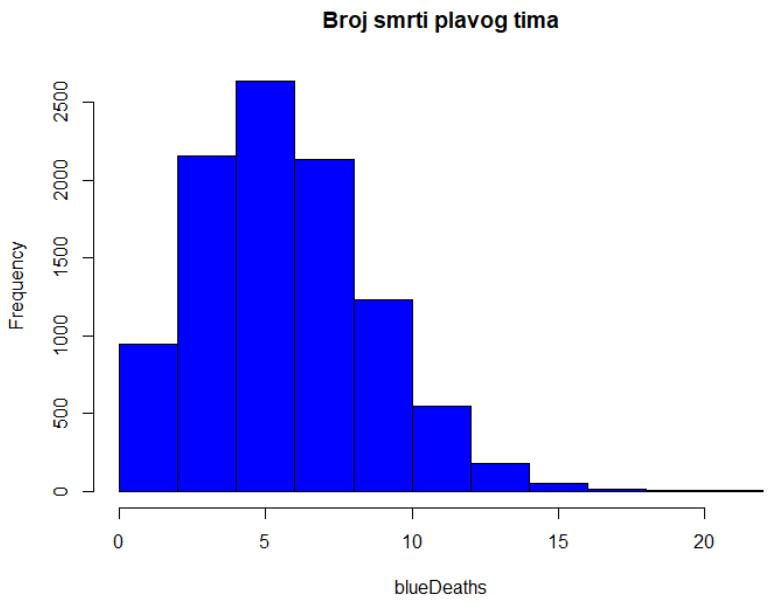


Slika 7. Distribucija vrijednosti varijable blueFristBlood

Slična distribucija kao i kod varijable koja označava pobjednički tim. Vrijednosti dobivanja prvog *killa* su podjednake za plavi i crveni tim.

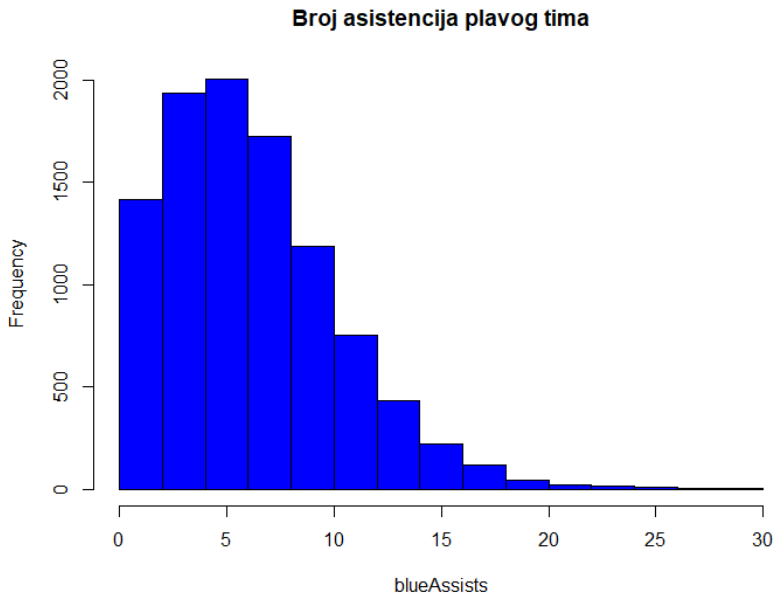


Slika 8. Distribucija vrijednosti varijable blueKills



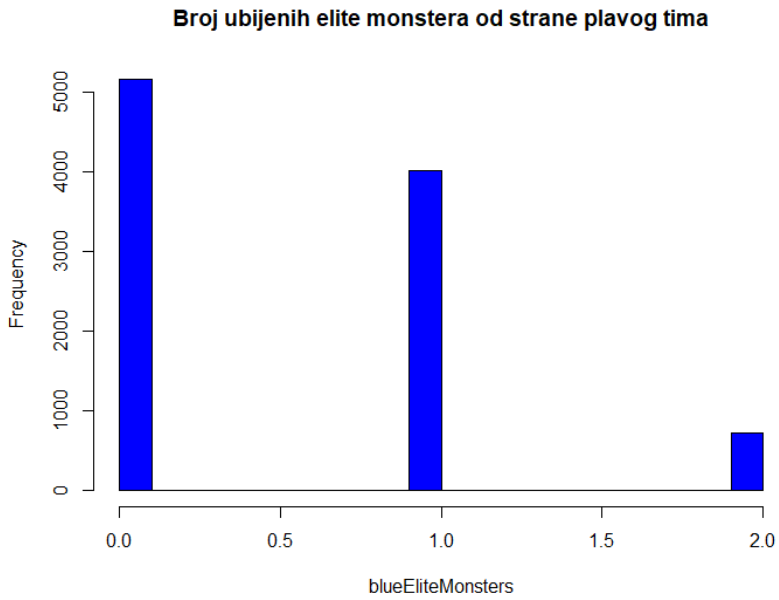
Slika 9. Distribucija vrijednosti varijable blueDeaths

Varijable koje označavaju broj *killova* i broj smrti plavog tima imaju približno normalnu distribuciju te gotovo nemaju outliera (3 outliera za broj *killova* i 5 outliera za broj smrti).



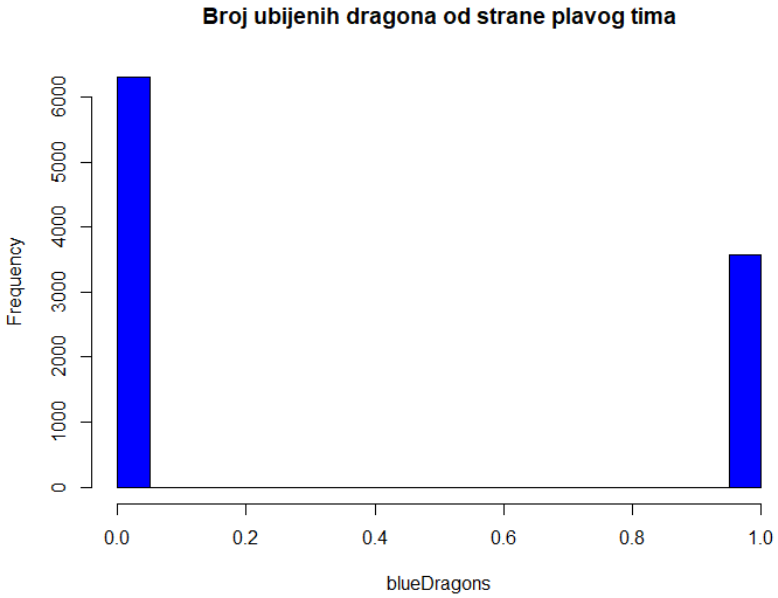
Slika 10. Distribucija vrijednosti varijable blueAssists

Broj asistencija ima nešto veći broj outliera, a razlog tome su opet mečevi u kojima protivnički tim odustane pa namjerno umire od strane plavog tima. Veći broj outliera u asistencijama je time objašnjen zato što za svaku namjernu smrt crvenog tima plavi tim dobije jedan *kill*, ali može dobiti više asistencija (maksimalno 4, ukoliko je cijeli tim sudjelovao u *killu*).

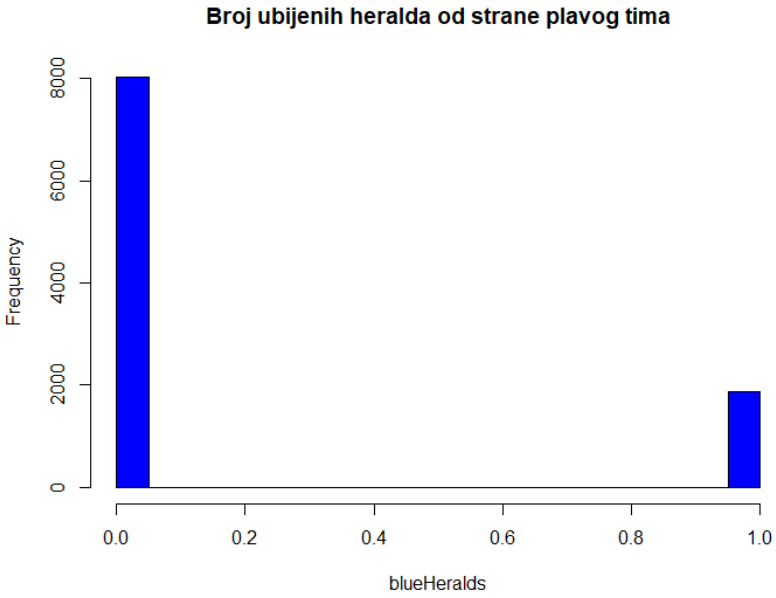


Slika 11. Distribucija vrijednosti varijable blueEliteMonsters

Broj ubijenih *elite monstera* ima očekivanu distribuciju. Naime, do 10. minute meča pojave se dva *elite monstera* (jedan *herald* i jedan *dragon*). Prema tome, tim može ubiti 0, jednog ili dva. Vrijednost 0 je najčešća zbog slučajeva u kojem nijedan tim ne ubije nijednog *elite monstera*, što je često zato što se *herald* pojavljuje tek nakon 8 minuta meča.

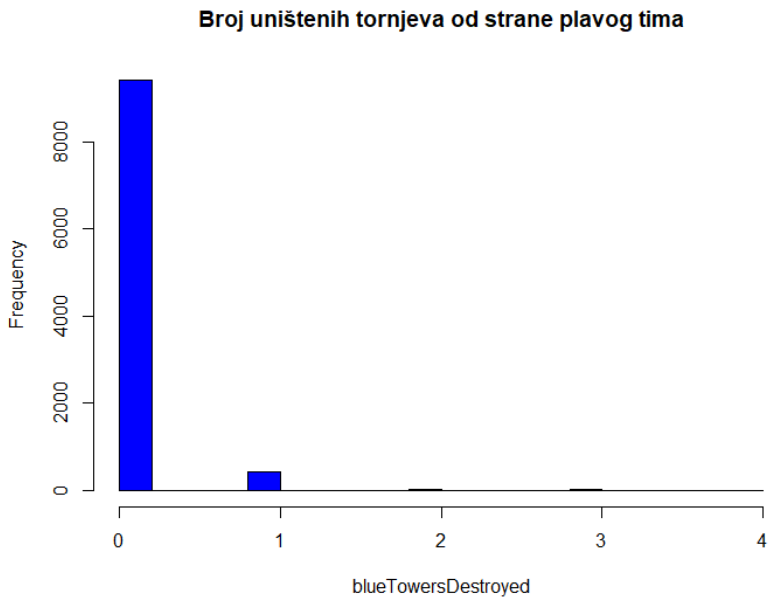


Slika 12. Distribucija vrijednosti varijable blueDragons



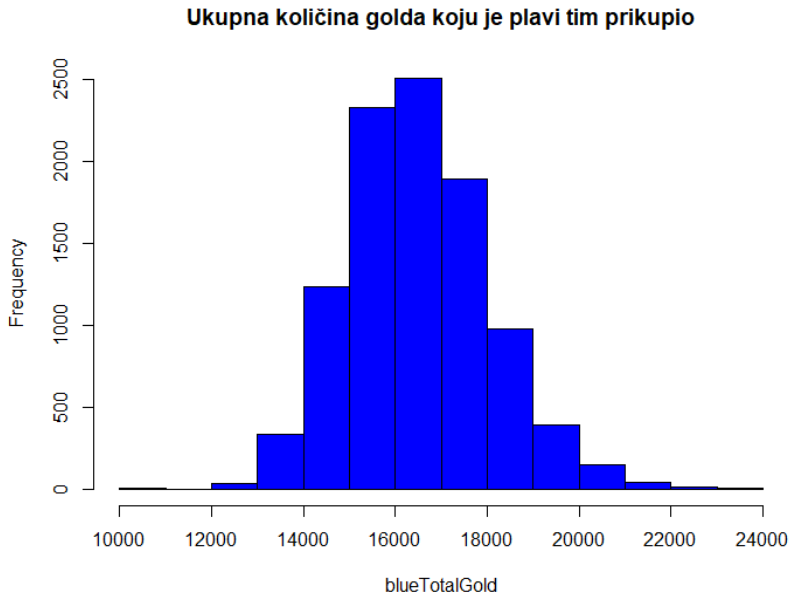
Slika 13. Distribucija vrijednosti varijable blueHeralds

Isto objašnjenje vrijedi i za broj ubijenih *dragona* i *heralda*. Prema tome, varijabla *blueEliteMonsters* je izvedena od varijabli *blueHeralds* i *blueDragons*.



Slika 14. Distribucija vrijednosti varijable blueTowersDestroyed

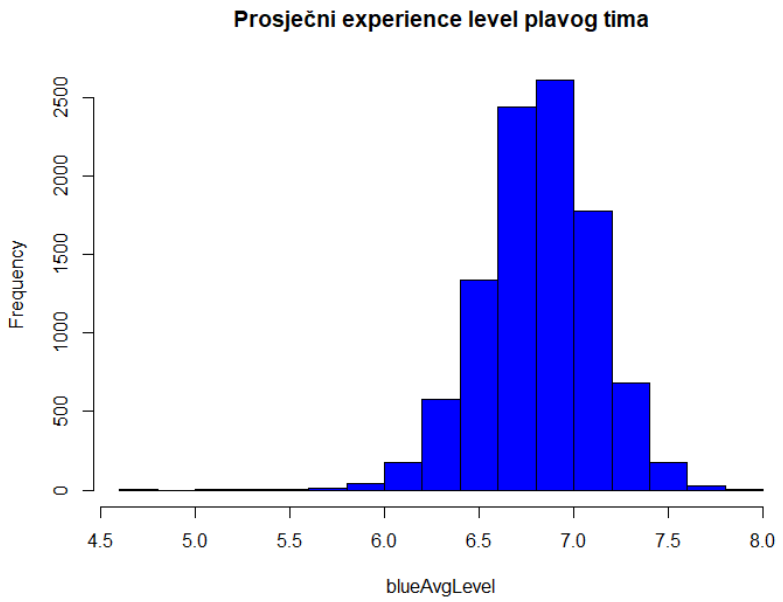
Broj uništenih tornjeva će najčešće biti 0, s obzirom da je najčešće vrijeme uzimanja prvog tornja između 14. i 17. minute. Ova varijabla ima relativno veliki broj outliera (464) opet zbog slučajeva odustajanja drugog tima.



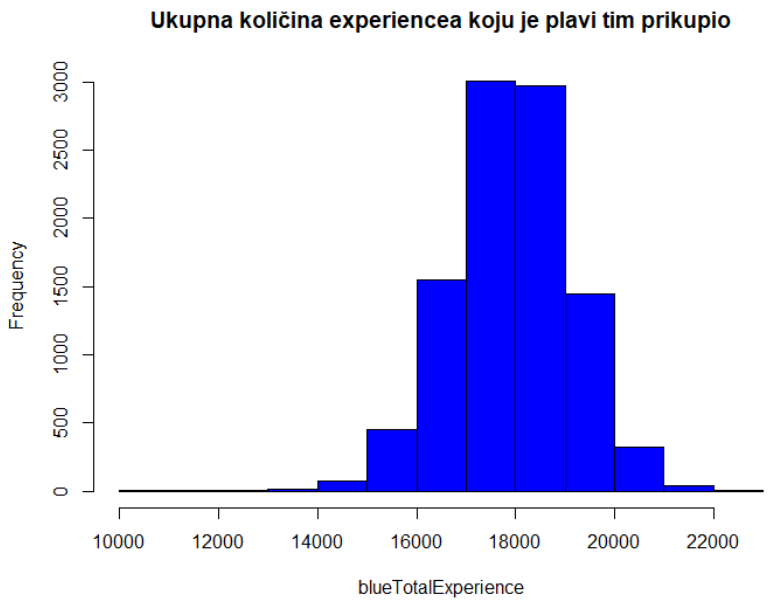
Slika 15. Distribucija vrijednosti varijable blueTotalGold

Ukupna količina *golda* koju je plavi tim prikupio ima približno normalnu distribuciju te relativno mali broj outliera (10 od 1000 mečeva). To je objašnjivo time da postoje sustavi koji

timu koji gubi omogućuje sve više načina da lakše prikupe *gold* i vrate se u meč (protivnici vrijede više *golda* ako je razlika u *goldu* između timova veća).

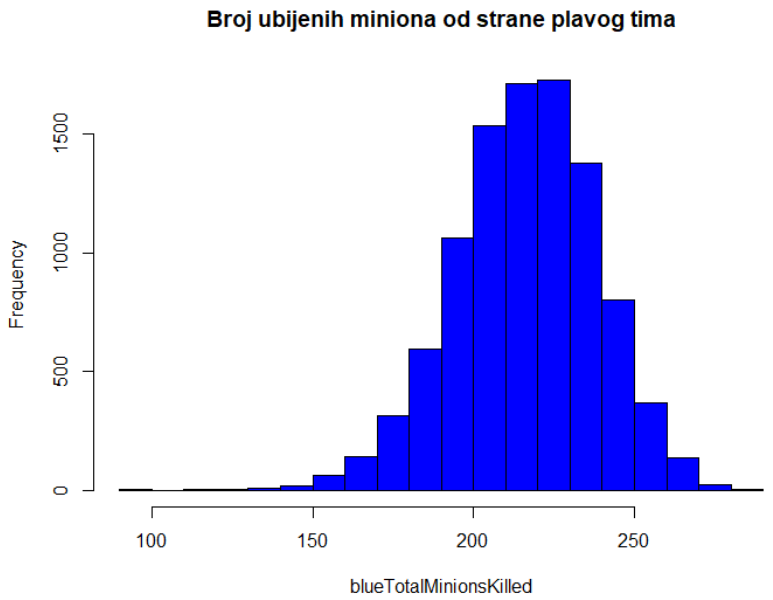


Slika 16. Distribucija vrijednosti varijable blueAvgLevel

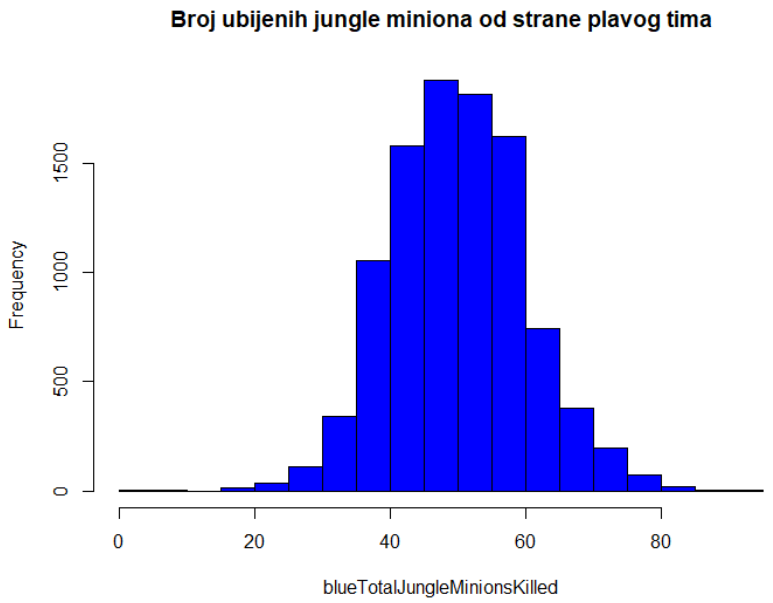


Slika 17. Distribucija vrijednosti varijable blueTotalExperience

Prosječni *experience level* i i ukupna količina *experiencea* imaju lagano negativnu asimetriju zbog slučajeva u kojima neki član tima prijevremeno napusti meč, pri čemu više ne skuplja *experience* (*gold* se u manjoj količini pasivno prikuplja čak i ako igrač odustane i ne igra aktivno).

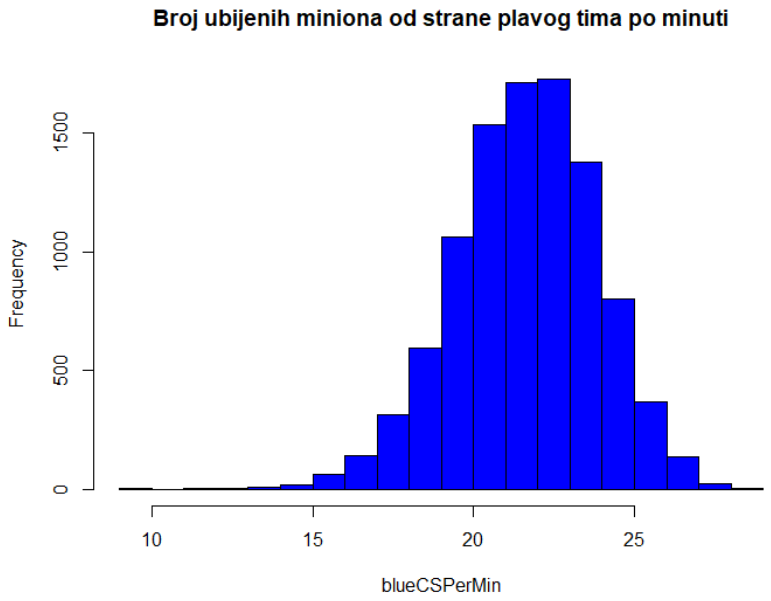


Slika 18. Distribucija vrijednosti varijable blueTotalMinionsKilled



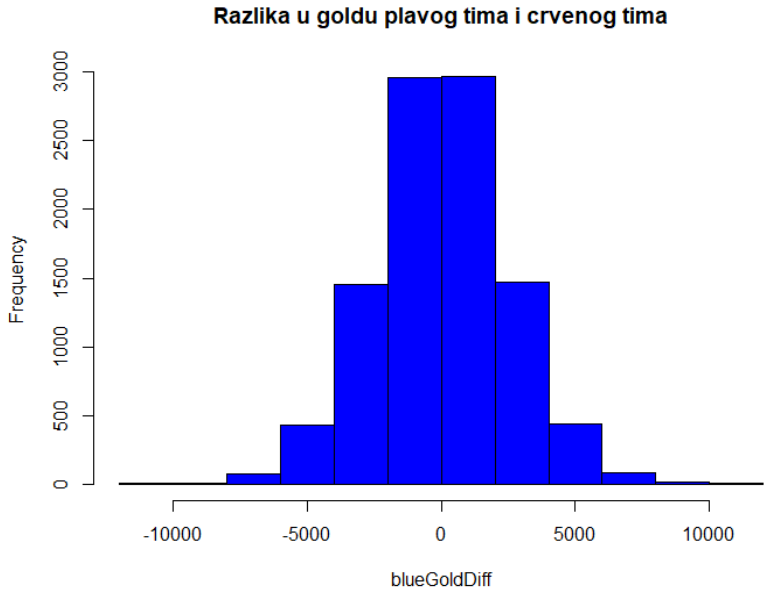
Slika 19. Distribucija vrijednosti varijable blueTotalJungleMinionsKilled

.

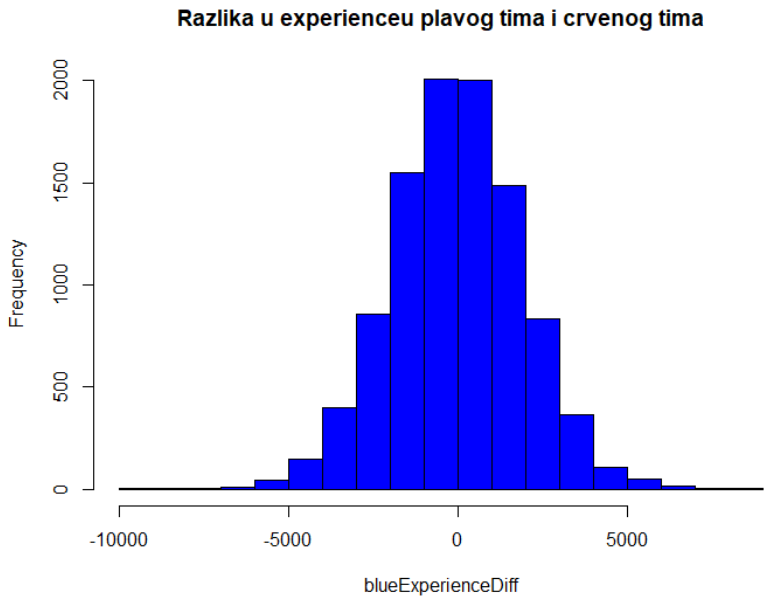


Slika 20. Distribucija vrijednosti varijable blueCSPerMin

Brojevi ubijenih *miniona* i *jungle miniona* te broj ubijenih *miniona* po minuti imaju približno normalnu distribuciju, opet sa nekoliko outliera u slučaju kada tim odustane i više ne ubija *minione*.

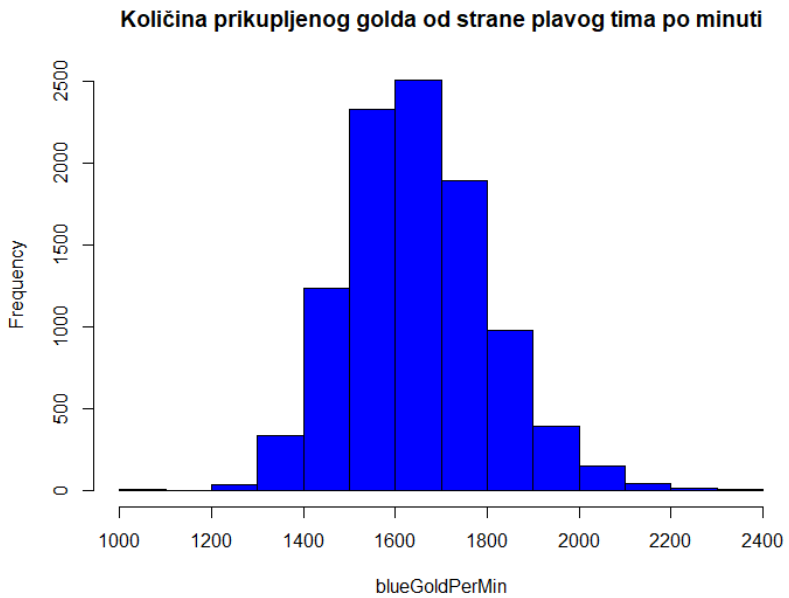


Slika 21. Distribucija vrijednosti varijable blueGoldDiff



Slika 22. Distribucija vrijednosti varijable blueExperienceDiff

Razlike u *goldu* te *experienceu* plavog i crvenog tima imaju normalnu distribuciju, što ukazuje na to kako strana mape ne utječe na prikupljeni *gold* i *experience*.



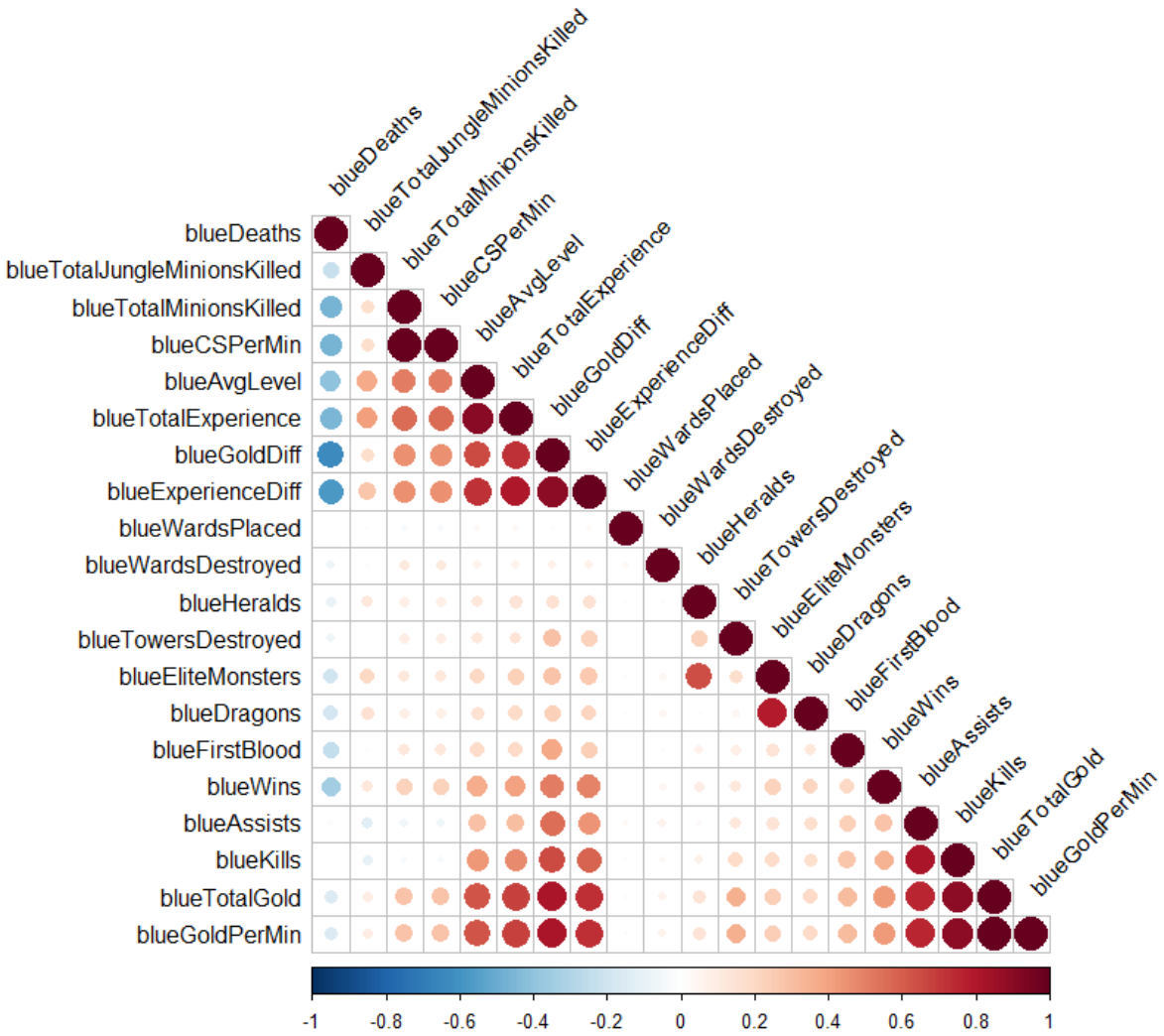
Slika 23. Distribucija vrijednosti varijable blueGoldPerMin

Isto tako, prikupljeni *gold* po minuti ima približno normalnu distribuciju uz relativno mali broj outliera (10).

Priprema podataka

Selekcija varijabli

Prije provođenja identifikacije outlier vrijednost i imputacije nedostajućih podataka će se provjeriti koje varijable će se koristiti. Naime, s obzirom da korišteni skup podataka ima 40 varijabli, neke od njih prikazuju gotovo iste činjenice te ih je potrebno ukloniti. Primjer takvih varijabli je “blueTotalMinions” i “blueCSPerMin” gdje prva govori koliko je miniona tim ubio, a druga koliko je “CS” ili miniona tim ubio po minuti. Metoda koja će se koristiti za identifikaciju takvih varijabli je matrica korelacija izrađena u alatu R Studio.

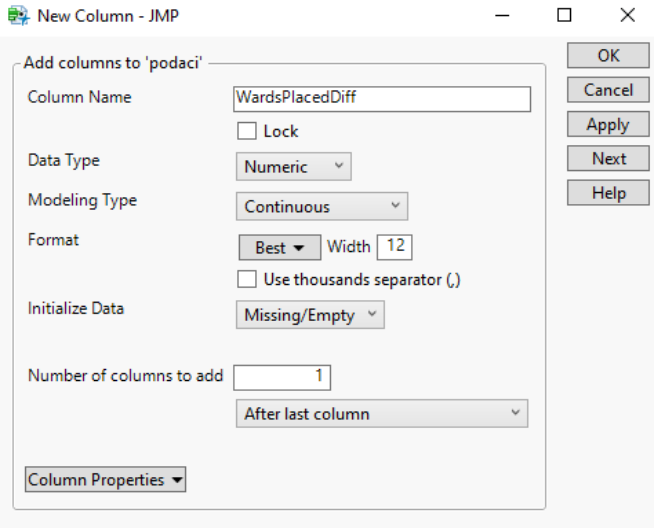


Slika 24. Matrica korelacija varijabli

Iz matrice korelacije je vidljiva snažna korelacija između spomenutih “blueCSPerMin” i “blueTotalMinionsKilled” pa će se “blueCSPerMin” izbaciti. Gotovo ista situacija je vidljiva sa “blueGoldPerMin” i “blueTotalGold” pa će se i “blueGoldPerMin” ukloniti. Isto tako, vidljiva je snažna korelacija između “blueAvgLevel” i “blueTotalExperience” pa će se “blueAvgLevel” ukloniti s obzirom da *level* direktno ovisi o količini *experiencea* koje igrač ima. Jaka korelacija je još vidljiva između varijabli “blueKills” i “blueTotalGold”, ali se ni jedna od tih varijabli neće ukloniti zato što postoje slučajevi u kojima tim može prikupljati više *golda* od *turreta* i *miniona* te biti u vodstvu iako imaju manje *killova*. Sličan slučaj je i sa varijablama “blueExperienceDiff” i “blueGoldDiff” gdje se unatoč jakoj korelaciji smatra da su obje varijable potrebne u modelu. Još jedna jaka korelacija je između blueKills i blueAssists, no opet se smatra kako su obje varijable potrebne s obzirom da *kill* kojem neko asistira timu donosi više *golda* od onog koji je postignut bez asistencije. Preostale jake korelacije nalaze se između varijabli poput “blueTotalGold” - “blueGoldDiff” i “blueTotalExperience” - “blueExperienceDiff”.

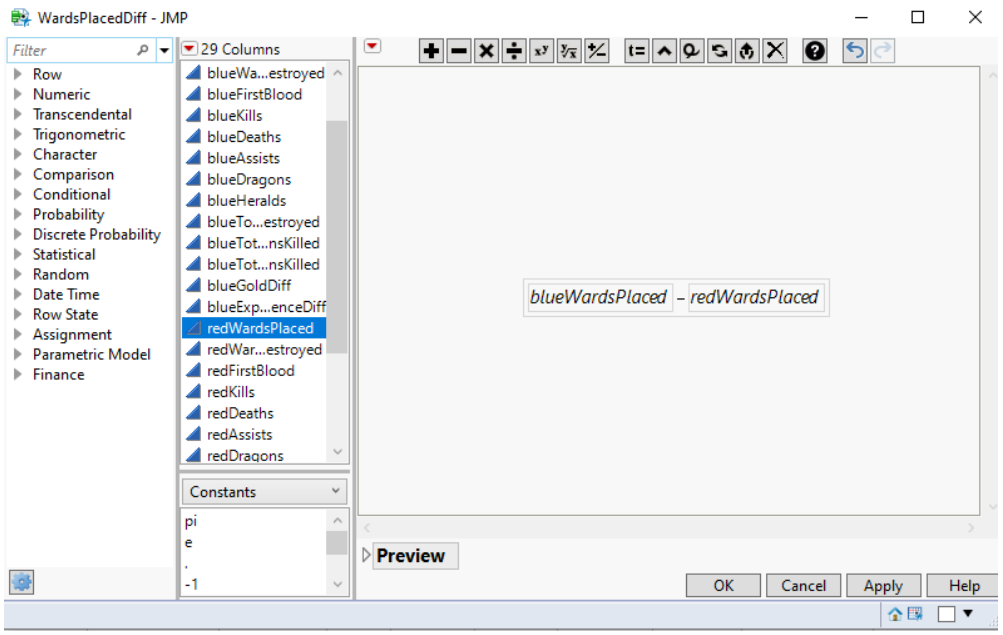
S obzirom da je League of Legends kompetitivna igra između dva tima, varijable koje prikazuju razliku između timova će uvijek biti korisnije, pa će se “blueTotalGold” i “blueTotalExperience” ukloniti. Naposljetku, ukloniti će se i varijabla “blueEliteMonsters” koja ima jake korelacije sa varijablama “blueDragons” i “blueHeralds” što ima smisla zato što direktno ovisi o njima (elite monsters = heralds + dragons).

Nakon uklanjanja početni skup podataka od 20 varijabli po timu smanjuje se na 14 varijabli. Kao što je već spomenuto, varijable koje prikazuju razliku između timova će uvijek biti korisnije od varijabli koje prikazuju stanje tima bez konteksta. Iz toga razloga će se sve preostale varijable koje već ne prikazuju razliku pretvoriti na način da prikazuju razliku između timova. Drugi pozitivan utjecaj te promjene je da sada više neće biti potrebne posebne varijable za plavi i crveni tim, već će se izražavati putem razlike između vrijednosti za crveni i plavi tim. Za postizanje navedene promjene koristi se JMP, nakon što se najprije uklone varijable identificirane kao redundantne analizom matrice korelacije koristeći Excel.



Slika 25. Izrada novog stupca

Radi jednostavnosti nazivlja varijabla će se nazvati “WardsPlacedDiff” umjesto “BlueWardsPlacedDiff”, a u nastavku će se pretpostavljati da radimo sa varijablama i modelima koji pokazuju vjerojatnost pobjede plavog tima na temelju varijabli koje prikazuju razlike oblika (plavi_tim - crveni_tim).



Slika 26. Prikaz oblika varijable “WardsPlacedDiff”

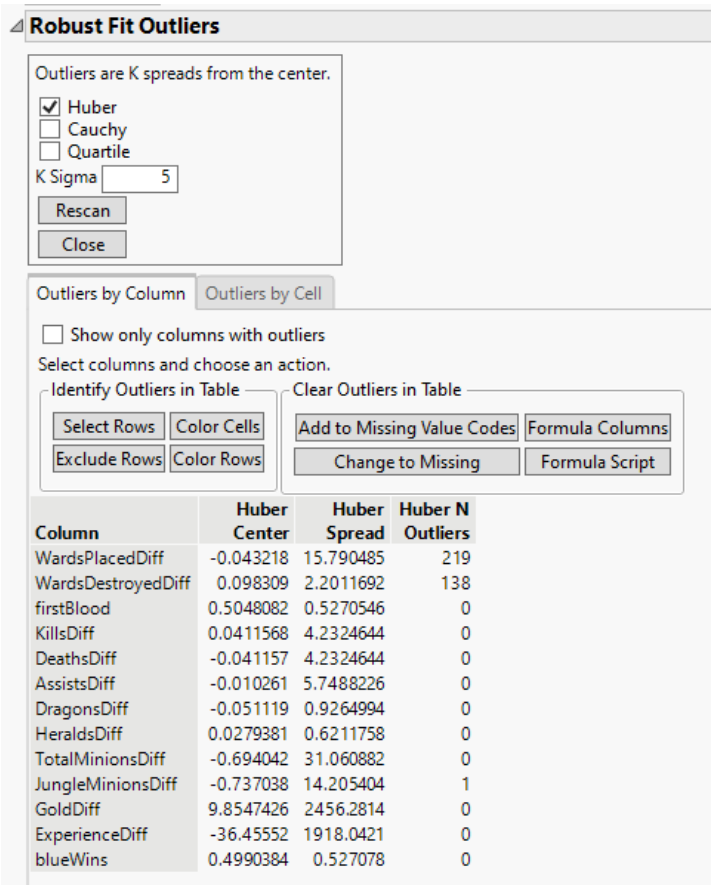
Nakon toga se prikazani proces ponavlja za preostalih 12 varijabli kako bi se dobio konačan rezultat selekcije ulaznih varijabli. Time je finalna selekcija ulaznih varijabli:

- WardsPlacedDiff,

- WardsDestroyedDiff,
- firstBlood,
- KillsDiff,
- DeathsDiff,
- AssistsDiff,
- DragonsDiff,
- HeraldsDiff,
- TowersDestroyedDiff,
- TotalMinionsDiff,
- JungleMinionsDiff,
- GoldDiff,
- ExperienceDiff.

Uklanjanje outlier vrijednosti

Uklanjanje outlier vrijednosti provodi se opcijom Robust Fit Outliers koristeći Huber funkciju te parametar K Sigma postavljen na vrijednost 5. Vidljivo je kako varijable WardsPlacedDiff i WardsDestroyedDiff jedine imaju signifikantan udio outliera, te se oni uklanjaju na način da se zamjenjuju nedostajućim vrijednostima.



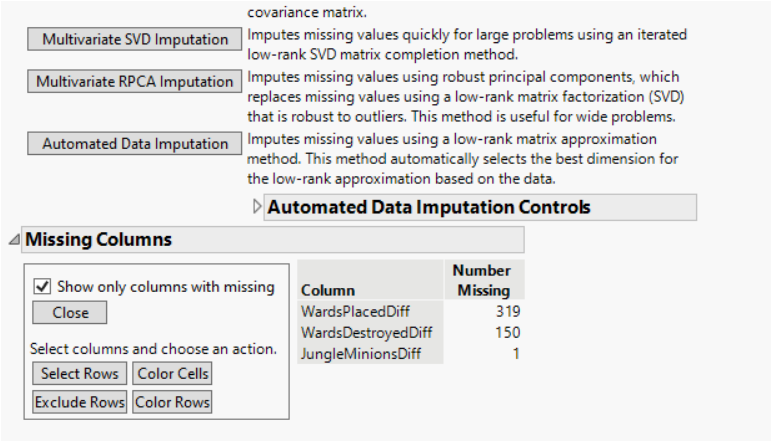
Column	Huber Center	Huber Spread	Huber N Outliers
WardsPlacedDiff	-0.043218	15.790485	219
WardsDestroyedDiff	0.098309	2.2011692	138
firstBlood	0.5048082	0.5270546	0
KillsDiff	0.0411568	4.2324644	0
DeathsDiff	-0.041157	4.2324644	0
AssistsDiff	-0.010261	5.7488226	0
DragonsDiff	-0.051119	0.9264994	0
HeraldsDiff	0.0279381	0.6211758	0
TotalMinionsDiff	-0.694042	31.060882	0
JungleMinionsDiff	-0.737038	14.205404	1
GoldDiff	9.8547426	2456.2814	0
ExperienceDiff	-36.45552	1918.0421	0
blueWins	0.4990384	0.527078	0

Slika 27. Opcija Robust Fit Outliers

Ovdje je bitno napomenuti da i TurretsDestroyedDiff isto tako ima signifikantan broj prepoznatih outlier vrijednosti, no ta varijabla je isključena iz postupka automatskog uklanjanja outliera. Razlog tome je činjenica da JMP prepoznaje vrijednosti 1 i 2 kao outliere, iako iz domenskog znanja znamo da su te vrijednosti u granicama normale za stanje meča. Iz toga razloga su se outlier vrijednosti za varijablu TurretsDestroyedDiff (kojih je bilo samo 7) ručno uklonile.

Imputacija nedostajućih vrijednosti

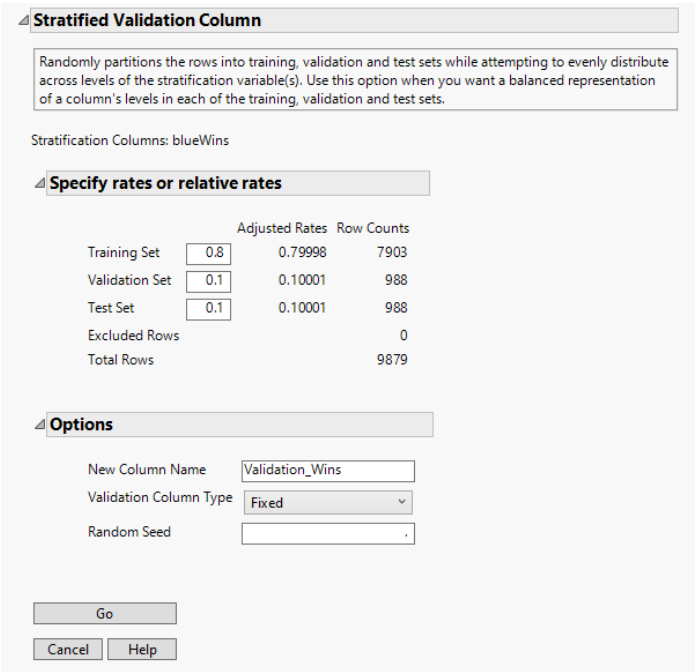
Zatim su se nedostajuće vrijednosti koje zamjenjuju outliere imputirale koristeći opciju Automated Data Imputation.



Slika 28. Prikaz automatske imputacije podataka

Izrada validacijskog stupca

Posljednji korak pripreme podataka za korištenje u modelima je izrada validacijskog stupca koristeći opciju Stratified Validation Column. Skup podataka podijeljen je na skup za treniranje (80%), skup za validaciju (10%) te skup za testiranje (10%).

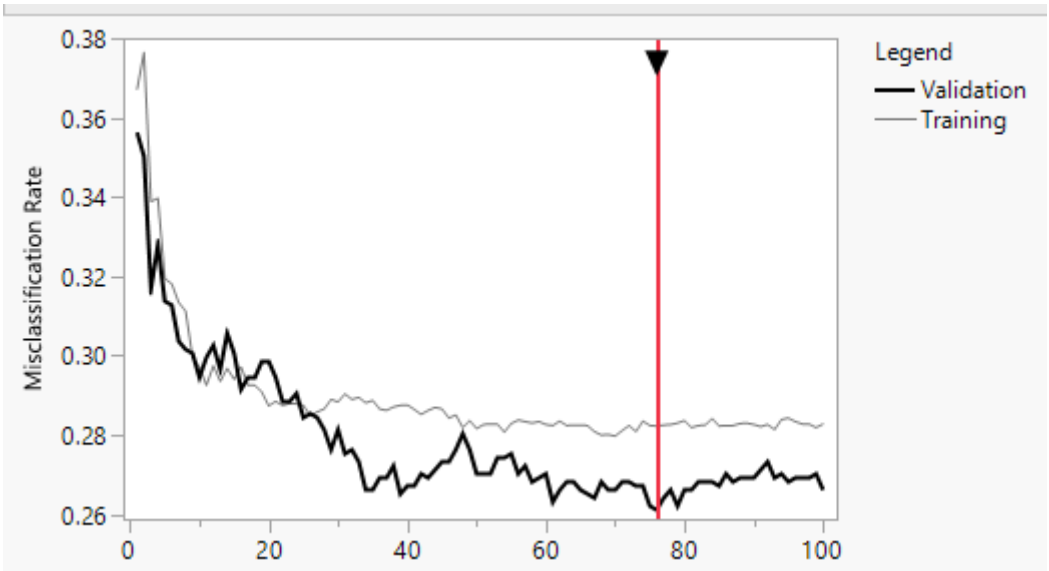


Slika 29. Prikaz izrade validacijskog stupca

Modeliranje podataka

k-najbližih susjeda

Kao prvi model koristi se k-najbližih susjeda. Ovaj model je odabran s obzirom da se koristio u obrađenoj literaturi. Osim toga, k-najbližih susjeda je pogodan za probleme klasifikacije sa jednom zavisnom varijablom. Problem ovog rada je upravo takav, pri čemu je jedna zavisna varijabla "blueWins" te ima dvije klase (0 i 1). Kod izrade modela je potrebno odrediti vrijednost parametra *k*, što se računa formulom $\sqrt{n}$, gdje je n broj instanci. U ovom slučaju je broj instanci 9879, pa će vrijednost parametra *k* biti postavljena na 100. Rezultat modela prikazan je sljedećom slikom:



Slika 30. Model k-najbližih susjeda

Vidljivo je kako je u validacijskom setu stopa pogrešnih klasifikacija pala ispod 0,28. Konkretno, najmanja stopa klasifikacije u skupu za testiranje postignuta je za *k* = 60, pri čemu je pogrešno klasificirano 26,9% instanci. S druge strane, maksimalna RSquared vrijednost postignuta je za *k* = 75.

Test

K	Count	RSquare	Misclassification Rate	Misclassifications	
60	988	0.21016	0.26923	266	*
61	988	0.20988	0.27328	270	
62	988	0.21054	0.27632	273	
63	988	0.21001	0.27530	272	
64	988	0.21027	0.27632	273	
65	988	0.20989	0.27733	274	
66	988	0.20928	0.27834	275	
67	988	0.20883	0.27328	270	
68	988	0.2088	0.27530	272	
69	988	0.20985	0.27227	269	
70	988	0.20969	0.27632	273	
71	988	0.20968	0.27429	271	
72	988	0.20981	0.27632	273	
73	988	0.20934	0.27733	274	
74	988	0.21059	0.27834	275	
75	988	0.21134	0.27632	273	
76	988	0.211	0.27530	272	

Slika 31. Prikaz stope pogrešne klasifikacije

S obzirom da je ovo klasifikacijski problem, model će se birati na temelju stope pogrešne klasifikacije. Kako je preporučljivo izabrati neparni *k,* u ovom slučaju će se odabrati *k* = 61. Sljedeća slika prikazuje matricu konfuzije za model sa *k* = 61.

Confusion Matrix for Best K=61

Training

Actual blueWins	Predicted Count 0	1
0	2877	1082
1	1150	2794

Actual blueWins	Predicted Rate 0	1
0	0.727	0.273
1	0.292	0.708

Validation

Actual blueWins	Predicted Count 0	1
0	367	128
1	132	361

Actual blueWins	Predicted Rate 0	1
0	0.741	0.259
1	0.268	0.732

Test

Actual blueWins	Predicted Count 0	1
0	362	133
1	137	356

Actual blueWins	Predicted Rate 0	1
0	0.731	0.269
1	0.278	0.722

Slika 32. Matrica konfuzije za model k-najbližih susjeda

Vidljivo je kako je postotak točne predikcije poraza plavog tima jednak 73,1%, dok je postotak točne predikcije pobjede plavog tima jednak 72,2%. Time je sposobnost točne predikcije izrađenog modela jednaka 72,67%.

U nastavku su prikazane neke od odrađenih optimizacija kako bi se vidjelo ako drugačije upravljanje outlierima, standardizacija varijabli ili izrada novih varijabli mogu poboljšati prediktivnu sposobnost modela.

Optimizacija modela

Prvo će se provjeriti ako normalizacija varijabli utječe na prediktivnu sposobnost modela. U ovom slučaju korištena je Min-Max normalizacija ulaznih varijabli.

Optimalan k	Stopa pogrešne klasifikacije	Sposobnost predikcije pobjede	Sposobnost predikcije poraza
61	0,27328	73,1%	72,2%

S obzirom da su rezultati u potpunosti jednaki, zaključuje se kako alat sam provodi normalizaciju prilikom izrade modela.

Zatim je istražen utjecaj broja susjeda na prediktivnu sposobnost modela, pa se kao broj susjeda odnosno *k* postavio na 200. Ipak nakon 10 iteracija izrade modela zaključilo se kako optimalan *k* nikada ne prelazi 85, pa je početna vrijednost od 100 dovoljna. Nakon toga odrađen je pokušaj optimizacije modela izradom novih varijabli:

- ObjectiveControl - (DragonsDiff + HeraldsDiff + TowersDestroyedDiff) / 3
    - Ova varijabla će pokazivati razliku kod timova što se tiče kontrole bitnih ciljeva na mapi.
- MapControl - (TowersDestroyedDiff + WardsPlacedDiff + WardsDestroyedDiff) / 3
    - Varijabla koja grupira varijable vezane uz kontrolu mape koja se postiže ubijanjem *turreta* što omogućuje postavljanje linije *visiona,* odnosno *wardova* dublje na protivničku stranu mape.
- TurretPressure - (TowersDestroyedDiff / (TotalMinionsDiff + 1)
    - Varijabla koja pokazuje koliko dobro se pritisak generiran kroz bolje ubijanje *miniona* prevodi u ubijanje *turreta*. Ubijanje više *miniona* od protivnika najčešće znači da se generira pritisak na tom dijelu mape, odnosno da *minioni* provode više vremena pod *turretom* protivnika što omogućuje lakše ubijanje *turreta*.
- KillConversion - (DragonsDiff + HeraldsDiff + TowersDestroyedDiff) / (KillsDiff + AssistsDiff + 1)

    - Varijabla koja pokazuje koliko dobro tim upotrebljava *killove* i *assistove* kako bi bolje kontrolirali ciljeve na mapi.
- Varijable TurretPressure i ObjectiveConversion se interpretiraju na sljedeći način:
    - ako je vrijednost blizu 0 ili 0, znači da razlika u *minionima*, odnosno u *killovima* i *assistovima* ne utječe značajno na broj srušenih *turreta*, odnosno na broj dobivenih *objectivea.*
    - ako je vrijednost pozitivna to znači da tim sa većim brojem *killova* i *assistova*, odnosno *miniona*, dobiva više *objectivea*.
    - ako je vrijednost negativna to znači da tim sa manjim brojem *killova*, *assistova* ili *miniona* dobiva više *objectivea.*

Na temelju ovih varijabli izrađena su dva modela. U prvom modelu navedene varijable dodane su kao nezavisne varijable zajedno sa svim drugim varijablama. Najbolja stopa klasifikacije dobivena je za *k* = 53 te iznosi 0.28644. Iz toga se zaključuje kako je taj model lošiji od modela bez novih varijabli. U drugom modelu, nove varijable su se koristile kao nezavisne varijable, ali ovaj puta su se uklonile one ulazne varijable koje sudjeluju u izračunu novih varijabli. Kod ovog modela postignuta je ista sposobnost predikcije kao i kod modela bez novih varijabli. Ipak kod ovog modela ta stopa pogrešne klasifikacije postignuta je za značajno manji *k*, u ovom slučaju 27. Iz toga se zaključuje kako je ovaj model za sada najbolji s obzirom da ukoliko dva modela imaju istu sposobnost predikcije, bira se onaj s manjim brojem susjeda zbog same jednostavnosti interpretabilnosti modela.

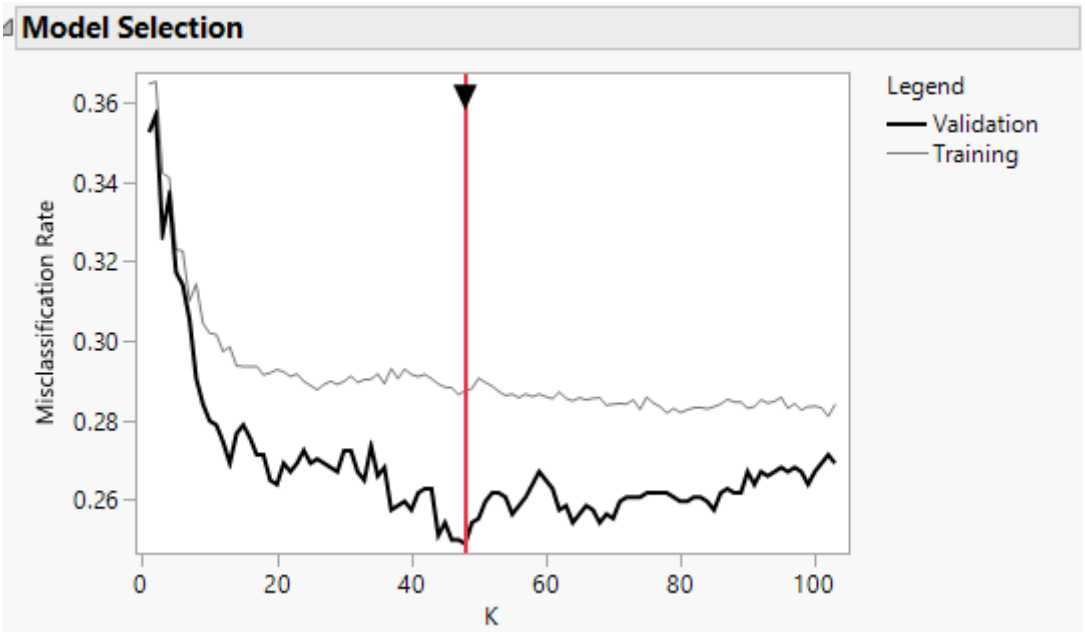
Iduće dvije optimizacije usmjerene su na rukovanje outlierima. Najprije će se napraviti model u kojem su instance koje sadrže outliere isključene iz modela, a nakon toga model u kojem outlieri nisu uklonjeni.

Optimalan k	Stopa pogrešne klasifikacije	Model
103	0,26496	Redovi s outlierima su uklonjeni
44	0,28340	Sa uključenim outlierima

Zaključuje se kako outlieri ipak negativno utječu na prediktivnu sposobnost modela. Pri tome kao najbolji način rukovanja outlierima u ovom slučaju pokazalo se uklanjanje redova koji sadrže outliere, a ne zamjena outliera nedostajućim vrijednostima. Time je

najbolji dobiven model za k-najbližih susjeda model nad podacima sa uklonjenim redovima outliera.

Evaluacija najboljeg modela



Slika 33. Najbolji model k-najbližih susjeda

Sa slike grafa kretanja stope pogrešne klasifikacije skupova za trening i validaciju vidljivo je kako je optimalan *k* = 48, nakon kojeg stopa pogrešne klasifikacije u skupu za validaciju opet raste. Ipak, više nas zanima prediktivna sposobnost modela nad skupom za testiranje, gdje stopa klasifikacije pada sve do *k* = 103. U tom trenutku je stopa pogrešne klasifikacije nad testnim skupom jednaka 0,26496, što znači da se ovim modelom uspješno može predvidjeti pobjednik na temelju stanja meča u 73,5% slučajeva.

Confusion Matrix for Best K=48

Training

Actual blueWins 2	Predicted Count 0	1
0	2670	1075
1	1078	2667

Actual blueWins 2	Predicted Rate 0	1
0	0.713	0.287
1	0.288	0.712

Validation

Actual blueWins 2	Predicted Count 0	1
0	357	111
1	122	346

Actual blueWins 2	Predicted Rate 0	1
0	0.763	0.237
1	0.261	0.739

Test

Actual blueWins 2	Predicted Count 0	1
0	336	132
1	132	336

Actual blueWins 2	Predicted Rate 0	1
0	0.718	0.282
1	0.282	0.718

Slika 34. Matrica konfuzije najboljeg modela

S obzirom da alat JMP prikazuje matricu konfuzije za *k* koji je najbolji za skup za validaciju, a ne testiranje, u ovom slučaju stopa pogrešne klasifikacije kod skupa za testiranje je nešto veća. Neovisno o tome, ono što se iz prikazane matrice konfuzije može zaključiti je da je prediktivna sposobnost pobjede i poraza plavog tima pojednaka (71,8%), te da nije potrebna izmjena granice klasifikacije (*category bias*). Za analizu osjetljivosti je bilo potrebno filtrirati skup podataka na način da se smanji broj redova, s obzirom da provođenje analize osjetljivosti nad cijelim skupom podataka nije bilo moguće. Iz toga razloga je provedena analiza osjetljivosti vezana uz model izrađen nad 800 redova nasumično uzetih iz skupa podataka:

Summary Report

Column	Main Effect	Total Effect
GoldDiff	0.137	0.408
ExperienceDiff	0.128	0.389
AssistsDiff	0.114	0.378
DeathsDiff	0.115	0.375
KillsDiff	0.111	0.371
TotalMinionsDiff	0.053	0.216
JungleMinionsDiff	0.028	0.143
WardsDestroyedDiff	0.025	0.126
DragonsDiff	0.031	0.123
WardsPlacedDiff	0.025	0.114
TowersDestroyedDiff	0.029	0.095
firstBlood	0.023	0.078
HeraldsDiff	0.024	0.078

Slika 35. Analiza osjetljivosti signifikatnih varijabli

Vidljivo je kako gotovo sve nezavisne varijable signifikantno utječu na izlaznu varijablu. Pri tome su najbitnije GoldDiff, ExperienceDiff, te varijable vezane uz *killove* i asistencije.

Neuronska mreža

Idući model koji će se izraditi je neuronska mreža, s obzirom da je model neuronske mreže isto bio prisutan u više obrađenih istraživanja te su rezultati bili zadovoljavajući te je isto pogodna za klasifikacijske probleme. S obzirom da je definirano 13 ulaznih varijabla i jedna izlazna, koristiti će se metoda izračuna broja neurona u skrivenom sloju kao aritmetička sredina broja ulaznih i izlaznih varijabli. Time će u ovom slučaju broj neurona u skrivenom sloju biti 7, pa će se izgraditi modeli sa 6, 7 i 8 neurona. S obzirom da je kod k-najbližih susjeda najbolji model izgrađen na temelju skupa podataka u kojem su redovi koji sadrže outliere uklonjeni, ovdje će se prvi model izgraditi upravo nad tim skupom podataka.

Sljedeća tablica prikazuje stope pogrešne klasifikacije za različite modele nad navedenim skupom podataka.

Broj neurona	Aktivacijska funkcija	Penalty method	Broj tura	Stopa pogrešne klasifikacije
7	TanH	Squared	10	0,25534
7	TanH	Squared	20	0,25747
6	TanH	Squared	10	0,26388
8	TanH	Squared	10	0,25854
8	Linear	Squared	10	0,26068
7	TanH	Weight Decay	10	0,27671

Nakon provedenih dodatnih varijacija uočeno je kako ni Penalty method ni aktivacijska funkcija nemaju signifikantan utjecaj na predikcijsku sposobnost modela (rezultati su uvijek unutar 1%). Model koji konzistentno daje najbolje rezultate je onaj sa 7 neurona, TanH aktivacijskom funkcijom, Squared Penalty metodom te je kod njega 10 tura dovoljno. Povećanje broja tura nije utjecalo na poboljšanje predikcijske sposobnosti, pa se zaključuje kako se najbolji model pronalazi u prvih 10.

Parametri navedenog modela zatim su se primijenili i kod kreiranja modela nad ostalim skupovima podataka, a sljedeća tablica prikazuje rezultate tih modela.

Skup podataka	Stopa pogrešne klasifikacije
Sa uključenim outlierima	0,26923
Sa outlierima zamijenjenima sa nedostajućim vrijednostima i imputiranim nedostajućim vrijednostima	0,26721
Sa dodatnim varijablama (objašnjene u prethodnom poglavlju)	0,26310

Na temelju provedenih varijacija zaključeno je kako je model sa 7 neurona u skrivenom sloju, TanH aktivacijskom funkcijom, Squared penalty metodom te uključenom opcijom Transform Covariates napravljen nad skupom podataka sa uklonjenim instancama koje sadrže outliere najbolji prema metrici stope pogrešne klasifikacije.

Evaluacija najboljeg modela

Kako bi se dobila prosječna prediktivna sposobnost modela, izrada modela se pokrenula 10 puta te se je izračunala prosječna stopa pogrešne klasifikacije, odnosno prosječan postotak točne klasifikacije kao i postotak odstupanja. Dobivena prosječna stopa pogrešne klasifikacije je 0,26185, odnosno 26,19% (+/- 0,5%). Time je prosječan postotak točnog predviđanja pobjednika dobivenim modelom neuronske mreže 73,81% (+/- 0,5%). Sljedeća slika prikazuje rezultate dobivenog modela.

Training

blueWins 2

Measures	Value
Generalized RSquare	0.3724608
Entropy RSquare	0.2363103
RASE	0.4216215
Mean Abs Dev	0.3540006
Misclassification Rate	0.2682243
-LogLikelihood	3964.8268
Sum Freq	7490

Confusion Matrix

Actual blueWins 2	Predicted Count 0	1
0	2763	982
1	1027	2718

Confusion Rates

Actual blueWins 2	Predicted Rate 0	1
0	0.738	0.262
1	0.274	0.726

Validation

blueWins 2

Measures	Value
Generalized RSquare	0.4098848
Entropy RSquare	0.264967
RASE	0.411682
Mean Abs Dev	0.340659
Misclassification Rate	0.2478632
-LogLikelihood	476.87895
Sum Freq	936

Confusion Matrix

Actual blueWins 2	Predicted Count 0	1
0	360	108
1	124	344

Confusion Rates

Actual blueWins 2	Predicted Rate 0	1
0	0.769	0.231
1	0.265	0.735

Test

blueWins 2

Measures	Value
Generalized RSquare	0.3686513
Entropy RSquare	0.233456
RASE	0.4220978
Mean Abs Dev	0.3538809
Misclassification Rate	0.2628205
-LogLikelihood	497.32282
Sum Freq	936

Confusion Matrix

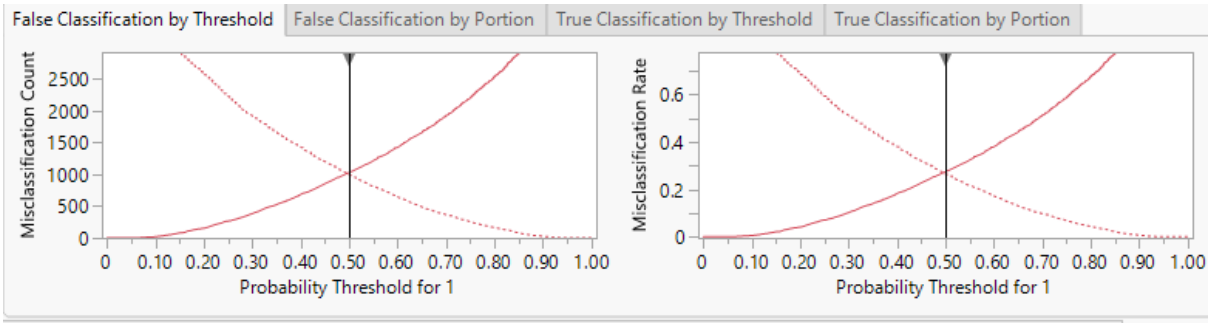
Actual blueWins 2	Predicted Count 0	1
0	349	119
1	127	341

Confusion Rates

Actual blueWins 2	Predicted Rate 0	1
0	0.746	0.254
1	0.271	0.729

Slika 36. Matrica konfuzije najboljeg modela neuronske mreže

Iz matrice konfuzije je vidljivo kako model nešto bolje predviđa poraz (74,6% slučajeva) nego pobjedu (72,9%). Ipak, višestrukim pokretanjem modela nisu uočeni uzorci u tom neslaganju te je zaključeno kako ne postoji potreba za modifikacijom parametra granice klasifikacije (*decision threshold*). To je vidljivo i sa sljedeće slike gdje se najmanja stopa pogrešne klasifikacije nalazi na *default* vrijednosti 0,5.



Slika 37. Graf granice klasifikacije

Overall

Column	Main Effect	Total Effect
GoldDiff	0.716	0.787
KillsDiff	0.086	0.125
ExperienceDiff	0.083	0.12
DeathsDiff	0.036	0.054
AssistsDiff	0.009	0.014
TowersDestroyedDiff	0.006	0.01
TotalMinionsDiff	0.002	0.004
DragonsDiff	0.003	0.004
JungleMinionsDiff	0.002	0.003
WardsDestroyedDiff	0.001	0.002
WardsPlacedDiff	0.001	0.001
firstBlood	3e-4	3e-4
HeraldsDiff	2e-4	2e-4

Slika 38. Analiza osjetljivosti najboljeg modela neuronske mreže

Iz analize osjetljivosti vidljivo je kako se četiri ulazne varijable mogu smatrati signifikantnim prediktorima, a to su GoldDiff, KillsDiff, ExperienceDiff i DeathsDiff. Situacija je slična kao i kod modela k-najbližih susjeda, gdje su najbitniji *gold*, *experience* te varijable vezane uz *killove*. Razlika je u tome što je kod neuronske mreže *GoldDiff* daleko signifikantniji prediktor od ostalih varijabli.

Naivni Bayesov klasifikator

Posljednji model koji će se izraditi je Naivni Bayesov klasifikator, s obzirom da radimo sa klasifikacijskim problemom. Iz prethodnih modela zaključuje se kako će najvjerojatnije najbolji model biti onaj koji se radi nad skupom podataka u kojem su redovi koji sadrže outlier vrijednosti isključeni iz izrade modela. Za provjeru, model se izradio nad tim skupom podataka, ali i nad skupom gdje su outlieri zamijenjeni nedostajućim vrijednostima koje su zatim automatski imputirane, te nad skupom u kojem su outlieri uključeni. S obzirom da dodatne ulazne varijable koje su izrađene u poglavlju k-najbližih susjeda do sada nisu značajno poboljšale nijedan model, ovdje se model sa njima neće raditi. Iduća tablica prikazuje rezultate izrađenih modela.

Skup podataka	Stopa pogrešne klasifikacije
redovi sa outlier vrijednostima uklonjeni	0,27778
outlieri uključeni	0,28947
outlieri zamijenjeni nedostajućim vrijednostima koje su zatim imputirane	0,28138

Kao što je i pretpostavljeno, najbolji model je onaj izrađen nad skupom podataka sa uklonjenim outlierima.

Evaluacija najboljeg modela

Kao što je navedeno, najbolji dobiveni model ima stopu pogrešne klasifikacije jednaku 0,27778. To znači da će točno predvidjeti pobjednika meča u 72,2% slučajeva.

Confusion Matrix

Training

Actual blueWins 2	Predicted Count 0	1
0	2700	1045
1	1049	2696

Actual blueWins 2	Predicted Rate 0	1
0	0.721	0.279
1	0.280	0.720

Validation

Actual blueWins 2	Predicted Count 0	1
0	355	113
1	121	347

Actual blueWins 2	Predicted Rate 0	1
0	0.759	0.241
1	0.259	0.741

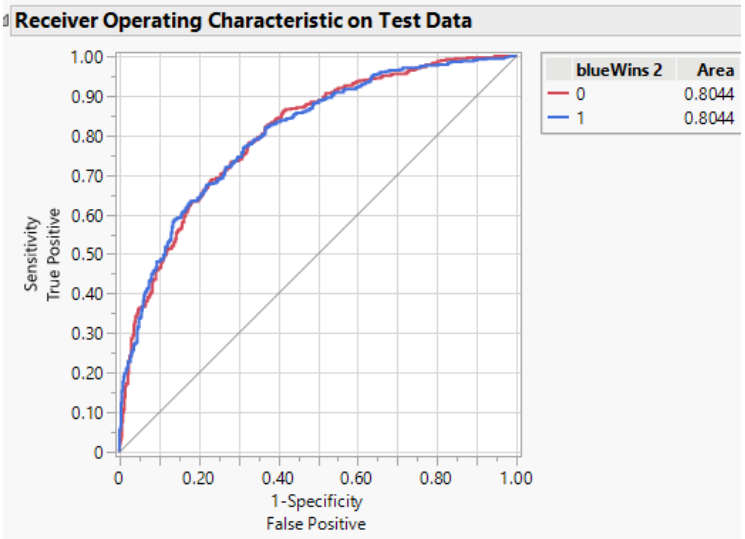
Test

Actual blueWins 2	Predicted Count 0	1
0	344	124
1	136	332

Actual blueWins 2	Predicted Rate 0	1
0	0.735	0.265
1	0.291	0.709

Slika 39. Matrica konfuzije najboljeg modela

Iz matrice konfuzije je vidljivo kako opet nešto bolje predviđa poraz plavog tima od pobjede, odnosno ima preveliki *bias* prema porazu. Na žalost u alatu JMP nije pronađen način da se *classification threshold* promijeni, no s obzirom da je razlika samo 3% zaključuje se kako se ne bi postigla velika poboljšanja.



Slika 40. ROC krivulja

ROC krivulja isto tako pokazuje kako je model dovoljno dobar, s obzirom da je površina ispod krivulje bliže 1 nego 0,5, odnosno bliža je savršenom modelu nego modelu koji nema nikakav kapacitet razdvajanja klasa. S obzirom da se često kao granica uzima vrijednost 0,8, ovaj model uspijeva zadovoljiti kriterij dobrog modela.

Overall

Column	Main Effect	Total Effect
ExperienceDiff	0.175	0.347
GoldDiff	0.159	0.337
DeathsDiff	0.132	0.276
KillsDiff	0.131	0.273
AssistsDiff	0.077	0.177
TotalMinionsDiff	0.045	0.102
TowersDestroyedDiff	0.025	0.058
JungleMinionsDiff	0.018	0.038
WardsDestroyedDiff	0.007	0.015
DragonsDiff	0.002	0.005
WardsPlacedDiff	0.001	0.003
firstBlood	0.001	0.003
HeraldsDiff	0.001	0.003

Slika 41. Analiza osjetljivosti najboljeg modela

Analiza osjetljivosti pokazuje sličnu situaciju kao i kod prethodnih modela, gdje su najbitniji prediktori varijable vezane uz *gold*, *experience* te *killove*. Ipak, ovaj model ima

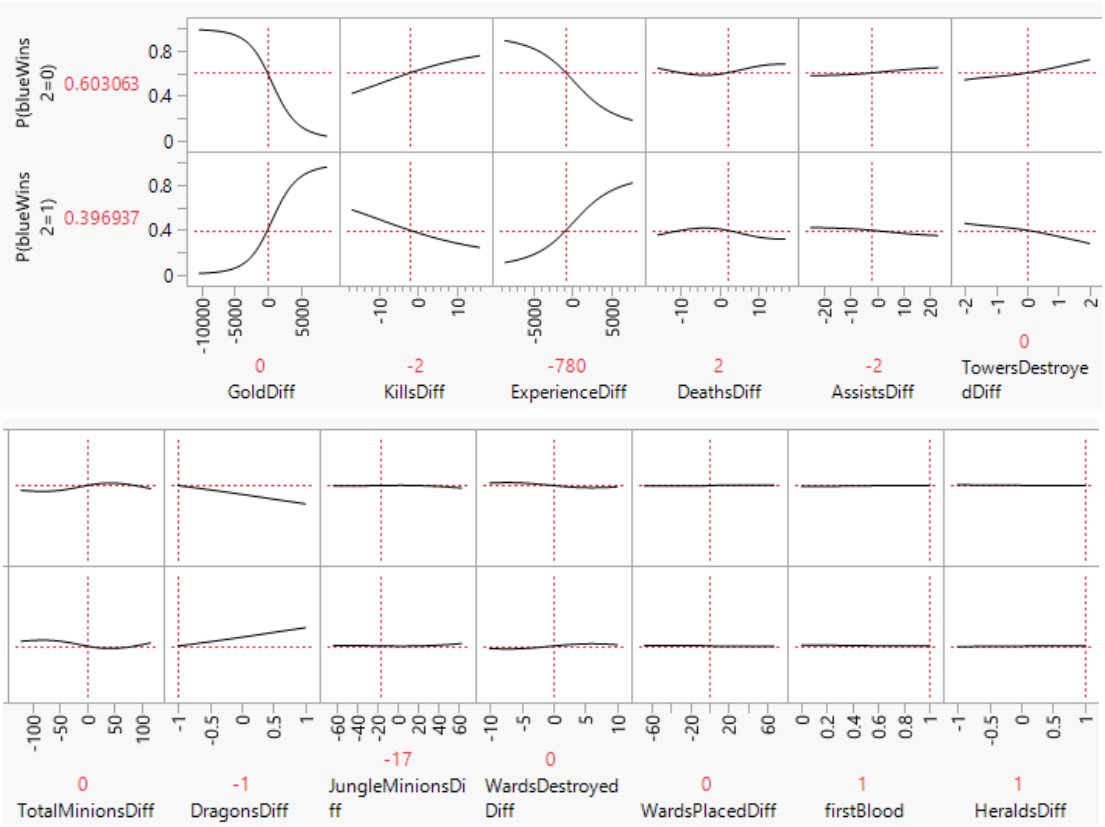
nešto više signifikantnih prediktora, pa se osim navedenih kao signifikantni smatraju i *TotalMinionsDiff* te *TowersDestroyedDiff*.

Interpretacija rezultata i primjena najboljeg modela

Postotak uspješnog predviđanja pobjednika meča na temelju stanja u desetoj minuti za najbolji model k-najbližih susjeda je 73,5%, za najbolju neuronsku mrežu je 73,81% (+/- 0,5%), dok je postotak uspješnog predviđanja za model Naivnog Bayesovog klasifikatora jednak 72,2%. Analizom osjetljivosti svakog modela je utvrđeno da su varijable koje su signifikantne neovisno o modelu varijable *GoldDiff*, *ExperienceDiff* te *KillsDiff* i ostale varijable vezane uz *killove* (*AssistsDiff* i *DeathsDiff*). Pri tome se *GoldDiff* pokazao kao najbolji prediktor, dok je *ExperienceDiff* drugi po razini signifikantnosti.

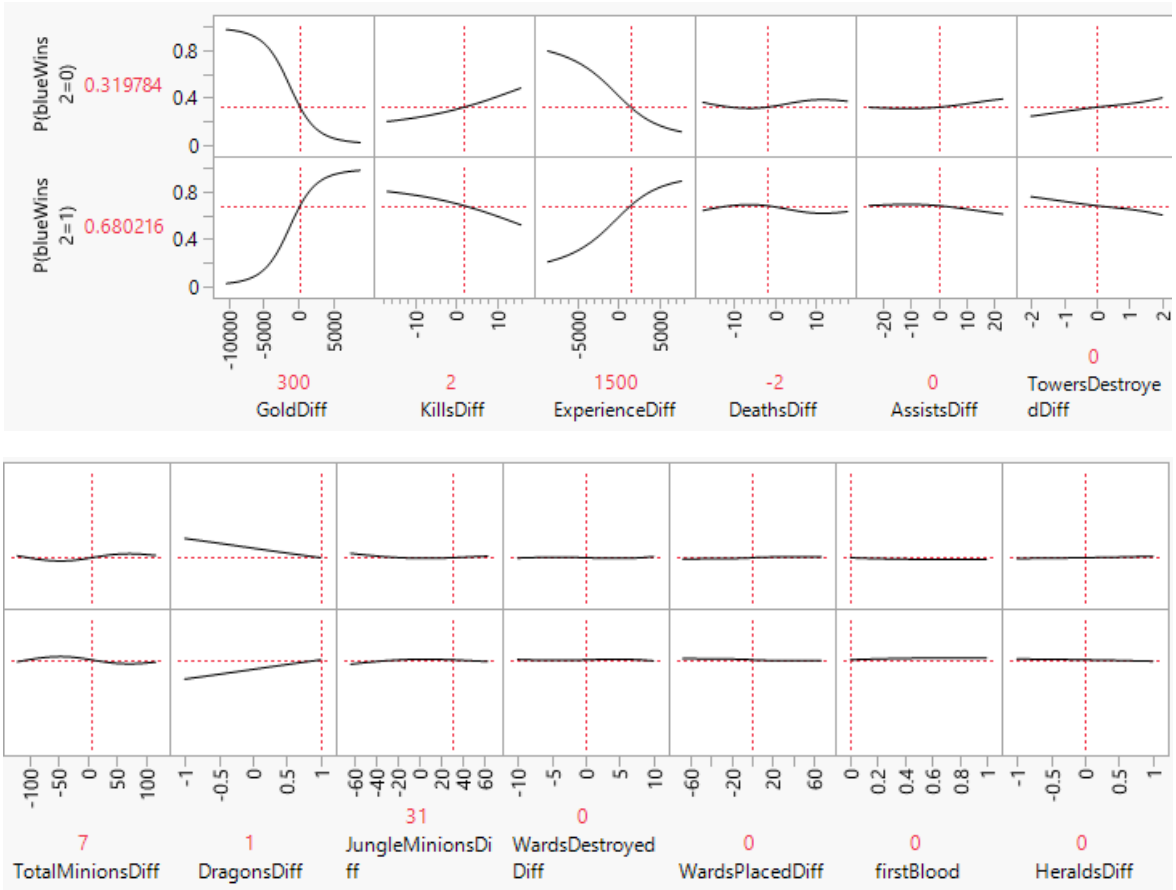
S obzirom da model neuronske mreže ima najbolju, iako samo malo veću prediktivnu sposobnost od modela k-najbližih susjeda, koristiti će se profiler tog modela za dobivanje daljnjih zaključaka i primjenu nad novim slučajevima. Osim toga, zbog količine ulaznih podataka profiler modela k-najbližih susjeda ne radi.

Kako bi primijenili model, odabrali smo 10 mečeva u kojima smo mi sudjelovali da vidimo ako će model uspješno predvidjeti pobjednika. Prvi meč je imao vrlo podjednako stanje između timova u desetoj minuti. Najveće prednosti protivničkog tima bile su kod varijabli *TotalMinionsDiff* te *KillsDiff*, kao i mala prednost u *ExperienceDiff*. S druge strane, naš tim (plavi) je dobio prvog *heralda* te prvi *kill* u meču. Jedine varijable za koje nismo dobili vrijednosti su *WardsPlacedDiff* i *WardsDestroyedDiff*, ali s obzirom da su one ionako vrlo slabi prediktori smatramo da je aproksimacija dovoljno dobra ako njihove vrijednosti postavimo na 0. Nakon što se podaci prvog meča stave u profiler, dobiva se sljedeći rezultat:



Slika 42. Izrada slučaja za prvi meč u Profileru

Model na temelju ulaznih podataka zaključuje kako postoji 60% šansa da izgubimo taj meč, a 40% šansa da pobijedimo. U ovom slučaju predikcija je točna, s obzirom da smo taj meč izgubili. Profiler će se prikazati za još jedan primjer meča, a zatim će se proces ponoviti za idućih 8 mečeva kako bi se dobili podaci o prediktivnoj sposobnosti izrađenog modela nad skupom podataka od naših 10 mečeva.



Slika 43. Izrada slučaja za drugi meč u Profileru

U ovom slučaju model predviđa 68% vjerojatnosti da naš tim pobijedi, te je opet u pravu. Nakon provedene analize za svih 10 mečeva, model je točno predvidio pobjednika u 8 mečeva. Iako je to mali skup podataka, zanimljivo je vidjeti kako model radi u skladu s očekivanjima nad skupom podataka od nekoliko naših, stvarnih mečeva. Isto tako, važno je napomenuti da je u slučajevima gdje nije uspješno predvidio pobjednika model bio manje od 60% siguran, tako da zaključujemo da ako je model dovoljno siguran u odluku (75+%) tada je njegova prediktivna sposobnost vrlo visoka.

Iz obrade prethodnih istraživačkih radova na ovu temu pozitivna činjenica je to da je izrađeni model po prediktivnoj sposobnosti usporediv sa modelima iz tih radova (točnost predikcije od 70-75%). Ipak, smatramo kako predikcija na temelju stanja meča u desetoj minuti ima više potencijala od prikazanog u ovom radu. Naime, smatramo kako bi glavno poboljšanje izrađenog modela bilo usmjereno na dodavanje varijabli koje označuju koje *championе* koji tim igra. Naime, neki *championi* su jači što duže meč traje, dok snaga drugih *championa* leži u borbama koje se odvijaju u ranim fazama meča. Time i podjednako stanje meča u desetoj minuti može biti povoljnije za jedan tim nego za drugi upravo zbog toga što jedan tim igra *championе* koji samo žele preživjeti rane faze meča i koji će postati jači

kasnije. Smatramo kako bi izrada varijable *scaling* koja pokazuje snagu kompozicije tima s vremenom ovisno o odabranim *championima* te uključivanje varijable u postojeći model signifikantno utjecalo na performanse modela.

Što se tiče glavnih prediktora, kao što je već i spomenuto, glavni prediktori su *GoldDiff*, *ExperienceDiff* te varijable vezane uz *killove.* Kada se uzme u obzir da *killovi* donose *gold*, a da kada netko nije živ ne dobiva *experience*, zaključuje se kako varijable vezane uz *killove* zapravo utječu istovremeno na *gold* i *experience*, pa su *GoldDiff* i *ExperienceDiff* time glavni prediktori. Iz toga razloga je zaključak to da bi se igrači trebali fokusirati na sve aktivnosti vezane uz generiranje *golda* te minimizirati vrijeme koje provode dalje od *miniona* kako bi maksimizirali dobiveni *experience*. Smatralo se kako će dobivanje ciljeva na mapi u prvih deset poput *dragona*, *heralda* i *towera* imati veći utjecaj, no ipak nema. To čini rane faze meča dosta jednostavnima, s obzirom da igrači trebaju razmišljati samo o dva resursa koja trebaju maksimizirati (*gold* i *experience*). Ipak, izgleda da je i sam *Riot Games* to uočio te se igra sve više mijenja prema tome da se dodavaju novi načini generiranja *golda* u ranim fazama meča te da rani ciljevi poput *dragona* i *heralda* postaju jači. Primjer toga je dodavanje *turret platinga* što veže mnogo više *golda* uz uništavanje *turreta* te *killovi* više nisu glavni način za generiranje što više *golda*. Isto tako, dodavanje *elemental dragona* je dalo novu dimenziju u rane faze meča s obzirom da *dragoni* sada više ne daju samo *gold* već i direktno poboljšavaju *championе*. Iz toga razloga se smatra kako bi izrađeni i modificirani model primijenjen na novijim podacima imao nešto zanimljiviju analizu osjetljivosti u kojoj bi signifikantno utjecao veći broj varijabli. Unatoč tome, smatra se kako je trenutni model zadovoljavajući te kako daje zanimljivi pogled u rane faze meča, a pri tome bi istaknuli važnost *experiencea* koji igrači često zanemaruju, a koji se pokazao kao drugi najvažniji prediktor kod predviđanja pobjednika.

Literatura

[1] Accessed: Oct. 20, 2023. [Online]. Available: https://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf

[2] Wikipedia, “League of Legends”, Accessed: Oct. 21, 2023. [Online]. Available: https://en.wikipedia.org/wiki/League_of_Legends

[3] T. D. Do, S. Ioi Wang, D. S. Yu, M. G. McMillian, and R. P. McMahan, “Using Machine Learning to Predict Game Outcomes Based on Player-Champion Experience in League of Legends”, Accessed: Oct. 22, 2023. [Online]. Available: https://arxiv.org/pdf/2108.02799.pdf

[4] J. A. Hitar-Garcia, L. Moran-Fernandez, and V. Bolon-Canedo, “Machine Learning Methods for Predicting League of Legends Game Outcome”, Accessed: Oct. 23, 2023. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9720122&tag=1

[5] S. Jagtap, “How We Trained a Machine to Predict the Winning Team in League of Legends”, Accessed: Oct. 24, 2023. [Online]. Available: https://medium.com/trendkite-dev/machine-learning-league-of-legends-victory-predictions-8bc6cbc7754e

[6] F. Bahrololloomi, S. Sauer, F. Klonowski, R. Horst, and R. Dorner, “A Machine Learning based Analysis of e-Sports Player Performances in League of Legends for Winning Prediction based on Player Roles and Performances”, Accessed: Oct. 24, 2023. [Online]. Available: https://www.scitepress.org/PublishedPapers/2022/108959/108959.pdf

[7] R. Smit, “A machine learning approach for recommending items in League of Legends,” Mar. 2023, Accessed: Oct. 24, 2023. [Online]. Available: https://www.cs.ru.nl/bachelors-theses/2019/Robin_Smit___4043561___A_machine_learning_approach_for_recommending_items_in_League_of_Legends.pdf

[8] AWS Amazon, “AWS GameDay: LoL Esports Edition”, Accessed: Oct. 24, 2023. [Online]. Available: https://aws.amazon.com/buildon/gamedays/league-of-legends-esports-edition/

[9] Y. Lan Ma, “League of Legends Diamond Ranked Games (10 min),” 2019, Accessed: Oct. 20, 2023. [Online]. Available: https://www.kaggle.com/datasets/bobbyscience/league-of-legends-diamond-ranked-games-10-min

[10] P. C Shih, “Figure 1”, Accessed: Oct. 25, 2023. [Online]. Available: https://www.researchgate.net/figure/A-Map-of-the-League-of-Legends-game-play-in-the-classic-mode_fig1_319839481

[11] League of Legends Wiki, “Item (League of Legends)”, Accessed: Oct. 25, 2023. [Online]. Available: https://leagueoflegends.fandom.com/wiki/Item_(League_of_Legends)

Popis slika

Slika 1. Summoner’s Rift mapa [10]
Slika 2. Sučelje shopa [11]
Slika 3. Nedostajuće i outlier vrijednosti
Slika 4. Distribucija vrijednosti varijable blueWins
Slika 5. Distribucija vrijednosti varijable blueWardsPlaced
Slika 6. Distribucija vrijednosti varijable blueWardsDestroyed
Slika 7. Distribucija vrijednosti varijable blueFristBlood
Slika 8. Distribucija vrijednosti varijable blueKills
Slika 9. Distribucija vrijednosti varijable blueDeaths
Slika 10. Distribucija vrijednosti varijable blueAssists
Slika 11. Distribucija vrijednosti varijable blueEliteMonsters
Slika 12. Distribucija vrijednosti varijable blueDragons
Slika 13. Distribucija vrijednosti varijable blueHeralds
Slika 14. Distribucija vrijednosti varijable blueTowersDestroyed
Slika 15. Distribucija vrijednosti varijable blueTotalGold
Slika 16. Distribucija vrijednosti varijable blueAvgLevel
Slika 17. Distribucija vrijednosti varijable blueTotalExperience
Slika 18. Distribucija vrijednosti varijable blueTotalMinionsKilled
Slika 19. Distribucija vrijednosti varijable blueTotalJungleMinionsKilled
Slika 20. Distribucija vrijednosti varijable blueCSPerMin
Slika 21. Distribucija vrijednosti varijable blueGoldDiff
Slika 22. Distribucija vrijednosti varijable blueExperienceDiff
Slika 23. Distribucija vrijednosti varijable blueGoldPerMin
Slika 24. Matrica korelacija varijabli
Slika 25. Izrada novog stupca
Slika 26. Prikaz oblika varijable “WardsPlacedDiff”
Slika 27. Opcija Robust Fit Outliers
Slika 28. Prikaz automatske imputacije podataka
Slika 29. Prikaz izrade validacijskog stupca
Slika 30. Model k-najbližih susjeda
Slika 31. Prikaz stope pogrešne klasifikacije
Slika 32. Matrica konfuzije za model k-najbližih susjeda
Slika 33. Najbolji model k-najbližih susjeda
Slika 34. Matrica konfuzije najboljeg modela
Slika 35. Analiza osjetljivosti signifikatnih varijabli
Slika 36. Matrica konfuzije najboljeg modela neuronske mreže
Slika 37. Graf granice klasifikacije
Slika 38. Analiza osjetljivosti najboljeg modela neuronske mreže
Slika 39. Matrica konfuzije najboljeg modela
Slika 40. ROC krivulja
Slika 41. Analiza osjetljivosti najboljeg modela
Slika 42. Izrada slučaja za prvi meč u Profileru

Slika 43. Izrada slučaja za drugi meč u Profileru