Prepoznavanje malicioznih napada pomoću strojnog učenja

Carević, Antonio

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Sveučilišni prijediplomski studij Računarstvo

PREPOZNAVANJE MALICIOZNIH NAPADA POMOĆU STROJNOG UČENJA

Završni rad

Antonio Carević

Osijek, 2024.

SADRŽAJ

1. UVOD

Brz razvoj Interneta omogućio je ljudima jednostavniju dostupnost informacijama što je značajno olakšalo osobne i poslovne potrebe mnogima. Međutim, velika količina informacija koje su prikupljene na internetu, od kojih su brojne osobne informacije, svakodnevno je izložena sigurnosnim prijetnjama. Jedna od glavnih sigurnosnih prijetnji današnjice su maliciozni napadi. Kako se brojnost zlonamjernih softvera iz dana u dan povećava, antivirusni programi nisu u mogućnosti zadovoljiti sve sigurnosne potrebe. Napadači imaju pristup automatiziranim tehnologijama koje su moćnije, a nove opasnosti od istih javljaju se gotovo svake sekunde te sigurnost čine još osjetljivom. Maliciozni napadi su namjerno dizajnirani s ciljem nanošenja štete računalnom sustavu, uglavnom u svrhu špijunaže ili financijske dobiti. Tradicionalne metode otkrivanja ovih napada postale su manje učinkovite, te je upravo ta dinamičnost modernih zlonamjernih softvera dovela je do potrebe naprednijih strategija otkrivanja napada. Upravu tu, u središte pozornosti, dolazi strojno učenje kao revolucija u detekciji zlonamjernih softvera. Strojno učenje omogućuje razvoj modernijih metoda za otkrivanje malicioznih napada posebice onih koji još nisu poznati. Korištenjem određenog algoritma strojnog učenja mogu se razviti sustavi koji imaju veliku moć prepoznavanja malicioznih napada, kako onih poznatih tako i novonastalih koji nas čekaju u skoroj budućnosti. Mogućnost prilagodbe i otkrivanje novih napada koji nisu još toliko poznati čini strojno učenje odličnim alatom za unaprjeđenje kibernetičke sigurnosti.

Ovaj rad ima za cilj utvrditi i implementirati standardni tok učenja klasifikacijskih algoritama iz skupova podataka koji opisuju maliciozne napade. Odabirom algoritama i tehnika predobrade stvorit će se modeli za prepoznavanje malicioznih napada. Tehnikama predobrade izvršiti će se odabir značajki kako bi se poboljšale performanse i povećala sposobnost generalizacije modela. Također, uz pomoć tehnike nasumičnog uzorkovanja smanjiti će se stupanj neuravnoteženosti skupova podatak isto sa ciljem poboljšavanja modela i povećavanja točnog prepoznavanja malicioznih napada.

U drugom poglavlju predstavljen je problem prepoznavanja malicioznih napada te su opisani česti napadi i njihov način funkcioniranja. Dodatno, u ovom poglavlju objašnjeno je strojno učenje i koncept klasifikacije kao i osnovni pojmovi za njihovo razumijevanje. U trećem poglavlju objašnjen je uobičajeni slijed procesa strojnog učenja, opisani su odabrani klasifikacijski algoritmi i tehnike predobrade skupova podataka. Zatim se u četvrtom poglavlju provodi eksperimentalna analiza i vrednuju se rezultati. U posljednjem poglavlju utvrđuje se koji se algoritam pokazao najboljim i uz pomoć kojih tehnika.

1.1 Zadatak završnog rada

Zadatak završnog rada je opisati problem prepoznavanja malicioznih napada kao problem klasifikacije nadziranog strojnog učenja. Analizom dostupne literature potrebno je utvrditi standardni tijek učenja iz skupova podataka koji opisuju ovaj problem, s posebnim osvrtom na tehnike predobrade skupova podataka i klasifikacijske algoritme. U praktičnom dijelu rada implementacijom odabranih klasifikacijskih algoritama i uz primjenu istaknutih tehnika predobrade nužno je napraviti eksperimentalnu analizu na javno dostupnim skupovima podataka. Pomoću provedene eksperimentalne analize potrebno je utvrditi kvalitetu odabranih algoritama za zadatak prepoznavanja malicioznih napada.

2. MALICIOZNI NAPADI NA RAČUNALA

U današnje doba gotovo je nezamisliv život bez Interneta. Sve velike organizacije i značajne tvrtke za poslovanje koriste Internet i računalnu tehnologiju u svakodnevnom radu. Olakšava im pristup potrebnim informacijama i poslovnu komunikaciju. Uz njih, veliki je broj privatnih korisnika Interneta koji ga koriste u različitim segmentima života. Gotovo sve osobne informacije moguće je pronaći na Internetu, poput primjerice preuzimanje osobnih dokumenata putem sustava e-Građanin, upotreba Internet bankarstva za izvršavanje transakcija, podaci o školovanju i slično.

Masovna primjena Interneta uz sve svoje dobre strane krije i mnogobrojne opasnosti. Razvojem tehnologije koja je omogućila dostupnost i široku primjenu Interneta stvorene su i nove mogućnosti ugrožavanja podataka. Broj različitih malicioznih napada povećava se svakodnevno, a njihov brz razvoj predstavlja veliku opasnost za svakog korisnika na Internetu. Najčešća meta napada su utjecajne državne institucije i privatni podaci korisnika. Postojeće tehnologije za obranu od malicioznih napada ne mogu držati korak sa njihovim brzim razvojem. Upravo zbog toga, svakodnevno se radi na uvođenju novih tehnika i suvremenih metoda kako bi se obrambeni sustavi pojačali, a rizik od napada smanjio.

2.1. Uvod u problem

Maliciozni napad (engl. *malware*) je generalni naziv za sve programe koji imaju zlu namjeru te koriste razne mehanizme kako bi zaobišli obranu sustava, a cilj im je neovlašteni pristup podacima. U smislu računalne sigurnosti, maliciozni napad je program koji se koristi za probijanje obrambenog sustava računala i kompromitiranje integriteta, dostupnosti i provjerljivosti [1]. Oni iskorištavaju postojeće slabosti sustava ili kreiraju nove, te ulaze u sustav bez da budu primijećeni. Načini na koje maliciozni programi ulaze u sustav su mnogobrojni, a neki od njih su: skidanje zaraženog programa na računalo, spajanje sa zaraženim uređajem, otvaranje lažnih Internet stranica, instaliranje lažnih programa, otvaranje elektroničke pošte koja za cilj ima krađu identiteta i ostali načini [2]. Najčešći razlozi napada zloćudnim programima su: financijska dobit, osveta, prikupljanje osobnih podataka ili osjetljivih informacija.

Žrtvom malicioznog napada može postati bilo tko od krajnjeg korisnika, mrežnih uređaja, kontrolnih ploča, servera. Važno je samo da uređaj ima neku vrstu računalne logike u sebi [2]. Svatko može napraviti pogrešku koja će dovesti njegov uređaj u opasnost, neovisno o poznavanju sigurnosnog rada na Internetu. Iako su pojedinci upoznati sa načinima na koje napadači pokušavaju ubrizgati maliciozni program u njihov sustav, intenzivan razvoj tehnologija omogućio je kreiranje

novih i sofisticiranijih metoda koje ih čine nemoćnima. Brojnost novih metoda onemogućuje stručnjacima iz područja računalne sigurnosti da ostanu u tijeku s novinama.

Maliciozni napadi mogu se klasificirati u više kategorija, a najpoznatije su: reklamni program, ucjenjivački program, virus, crv, mrežna krađa identiteta, ubrizgavanje URL-ova, napad distribuiranim onemogućivanjem pružanja usluge, špijunski program, spam poruke i trojanski konj. Reklamni program (engl. *adware*) je zlonamjerni program koji prikazuje neželjene reklame u obliku skočnih prozora ili unutar web preglednika. Glavna svrha ovakvih programa je prikupljanje zarade ali nekada znaju u sebi imati ugrađen špijunski program koji prikuplja podatke žrtve i prati njenu aktivnost [3]. Osim prikupljanja zarade, koristi se i u svrhu reklamiranja proizvoda i povećanja popularnosti stranica. Dijele se u dvije skupine. Prvu skupinu čine legitimni programi za reklamiranje koji nemaju zlih namjera ali konstantnim prikazivanjem reklama mogu narušiti koncentraciju u radu. Drugu skupinu čine zlonamjerni reklamni programi koji uz prikazivanje reklama prikupljaju podatke. U slučaju da se instalira u web preglednik, može prikupljati lozinke, korisnička imena, podatke elektroničke pošte i IP adresu računala.

Ucjenjivački program (engl. *ransomware*) je oblik malicioznog napada u kojem napadač blokira pristup podacima ili računalnim resursima dok se ne ispune njegovi zahtjevi. Najčešći zahtjev je uglavnom plaćanje određene količine novca [4]. Podatke najčešće enkriptira ili zaključa cijeli sustav dok se na ekranu prikazuje prijeteća poruka koja prisiljava na plaćanje. Kako bi povećao uvjerljivost poruke, napadač često u kreaciji koristi obilježja vladinih organizacija ili države. Primjerice, poruke mogu sadržavati grb države ili izgledati kao da su korisniku upućene od strane Ministarstva unutarnjih poslova. Dodatna opasnost kod ove vrste napada je što nakon uplate novca nije zagarantirano da će napadač dati ispravan ključ za pristup podacima [4].

Virus je maliciozni kod koji se u normalne programe i tako ih inficira, a ima sposobnost samorazmnožavanja. Jednom kada dospije u računalni sustav, stvara svoje nove kopije i širi se inficirajući druge programe. Ovisno o vrsti virusa, okidač koji pokreće njegovu funkciju može varirati. Najčešći oblik okidača je pokretanje zaraženog programa, pri čemu se pokreće i izvršava maliciozni kod. Neki virusi, kao Michelangelo , pokreću se na točno određeni dan. U slučaju ovog virusa to je 6. travnja kada je rođendan Michelangela Buonarroti-a po kome je virus dobio naziv [5]. Funkcija virusa najčešće ima maliciozne namjere kao što su brisanje datoteka na računalu, krađa novaca i privatnih podataka, blokiranje rada računala i mnoge druge.

Crv je maliciozni program sličan virusu. Isto kao i virus iskorištava ranjivosti instaliranih aplikacija ili sustava kako bi izvršio infekciju. Širi se pomoću Interneta, najčešće putem elektroničke pošte ili preuzimanjem inficiranih programa. Također ima sposobnost samorazmnožavanja, ali za razliku od virusa nije im potrebna ljudska aktivnost za aktivaciju i

širenje na druge sustave. U većini slučajeva su zlonamjerni te se koriste za krađu ili brisanje podataka. Kada nemaju zlu namjeru tada koriste resurse sustava kako bi stvorili kanale kojima inficiraju ostale sustave i tako usporavaju rad računala [3].

Mrežna krađa identiteta (engl. *phishing*) je oblik malicioznog napada u kojem napadač navodi žrtve da podijele svoje privatne informacije kao što su broj bankovne kartice ili korisnički podatci na osnovu prijevare. Najčešći oblik je prijevara putem elektroničke pošte ili navođenje na lažne Internet stranice koje imaju ista obilježja kao i prave stranice te tako zavaravaju žrtvu da su na pravoj stranici. Na taj način napadači nude primamljive ponude koje je teško odbiti [6]. Poznati oblik ovoga napada je i elektronička pošta koja informira osobu da je dobila znatnu količinu novaca u naslijeđe ili nude primamljive proizvode po niskim cijenama.

Ubrizgavanje URL-a (engl. *URL Injection*) vrsta je malicioznog napada sličan mrežnoj krađi identiteta. Kod ovog oblika napada napadač kreira lažnu Internet stranicu te ju postavlja unutar normalne stranice. Na taj način želi preusmjeriti korisnike na svoju lažnu stranicu s ciljem prikupljanja podataka. Poseban oblik ovog napada je ubrizgavanje u SQL (engl. *SQL Injection*), koji ima za cilj ostvariti pristup bazi podataka Internet stranice. To ostvaruje na način da postavi maliciozni kod na mjesto unosa podataka [7]. Nakon što se maliciozni kod implementira u bazu, napadač ju može preurediti po želji i omogućen mu je pristup svim podacima.

Distribuirani napad uskraćivanjem usluga (engl. *distributed denial-of-service*, DDoS) poseban je oblik napada uskraćivanjem usluga (engl. *denial-of-service*, DoS) koji može biti usmjeren raznim metama, od visoko utjecajnih organizacija, banaka i vladinih institucija pa do napada na osobna računala kako bi se skratila usluga pristupanja Internetu. Nedavni napadi na Zagrebačku banku, KBC Zagreb, Ministarstvo financija i Hrvatsku narodnu banku su pokazatelji kako ova vrsta napada može biti učinkovita i u kratkom vremenskom periodu prouzrokovati veliku štetu. Slanjem velike količine beznačajnog prometa napadači zatrpavaju sustave te ih tako dodatno opterećuju i onemogućuju normalno funkcioniranje [8].

Špijunski program (engl. *spyware*) je maliciozni program koji špijunira aktivnosti korisnika tako što koristi mogućnosti sustava kojeg su zarazili. Bez znanja korisnika prate različite aspekte njegovog ponašanje na računalu i na internetu te prikupljaju podatke o tipkanju po tipkovnici. Svi se podaci šalju autoru programa koji ih tada može koristiti u razne svrhe. Mogu imati i dodatne funkcionalnosti kao što je ometanje Internetske veze s ciljem oslabljivanja obrambenog mehanizma sustava. Prenose se putem normalnih aplikacija u obliku Trojanskog konja kako bi neprimijećeno ušli u sustav ili iskorištavaju slabosti sustava [3].

Spam poruke predstavljaju prijetnju računalnim i mrežnim resursima jer ih zatrpavaju svojom brojnošću i zauzimaju veliku količinu propusnosti te tako usporavaju ili u potpunosti onemogućuju njihovo funkcioniranje. Danas su to poruke u obliku elektroničke pošte koje služe za promociju ili prikaz neželjenog sadržaja [4]. Iako su u većini slučajeva bezopasne, neke od njih mogu sadržavati dodatan oblik malicioznog napada ili programa koji može prouzrokovati dodatnu štetu uređaju.

Trojanski konj je program koji se na prvu čini kao normalni program koji je bezopasan, ali tek kada se instalira na računalo započinje sa izvođenjem malicioznog napada, kao što je krađa povjerljivih podataka [9]. Ovaj oblik napada jedan je od najopasnijih jer žrtva u većini slučajeva ne primjećuje da je u opasnosti dok nije prekasno.

2.2. Prepoznavanje malicioznih napada

Maliciozni napadi postaju sve sofisticiraniji i predstavljaju sve veći problem u prostoru kibernetičke sigurnosti. Broj malicioznih napada sve je veći, raste iz dana u dan, te obrambeni sustavi ne mogu ostati uz korak sa njima. Oskudni broj tehnika, loše konfiguracije sustava kibernetičke sigurnosti i nekvalificirano osoblje samo su neki od razloga koji uzrokuju povećani rizik od napada u računalnim sustavima. Klasični mehanizmi zaštite od malicioznih napada sve brže postaju zastarjeli. Nemogućnost prepoznavanja novih napada njihova je najveća mana koja ih čini nemoćnima protiv do sada neviđenih napada [10].

Potreba za stručnjacima u području kibernetičke sigurnosti svake je godine sve veća, a njihov broj sve manji. Njihov manjak je jedan od glavnih razloga smanjenje kvalitete obrambenih sustava jer malobrojno osoblje ne može pratiti puno brži razvoj tehnika za izvršavanje malicioznih napada. Glavna prepreka u povećavanju broja ljudi sposobnih za izvršavanje zadataka kibernetičke sigurnosti je nemogućnost za njihovim treniranjem i usavršavanjem. Za njihovo treniranje tvrtke bi trebale osigurati prostor za vježbanje kao i praktičnu obuku na korištenoj opremi i procesima da bi postali dovoljno vješti za rad. Međutim, edukacijski sektor ne izučava dovoljno sposobnog osoblja kako bi zadovoljili potrebu svjetskog tržišta [11].

Prva pomisao na spomen obrane od malicioznih napada su antivirusni programi, ali oni predstavljaju samo programe koji se koriste za obranu od malog dijela malicioznih napada, uglavnom za sprječavanje raznih vrsta virusa u njihovoj namjeni infekcije računala. Njihova je primjena ograničena i oni štite samo računalo na kojem su instalirani, dok su ostali uređaji sa kojima je računalo umreženo i dalje u opasnosti od infekcije. Puno bolju i potpuniju uslugu obrane pruža sustav za otkrivanje upada (engl. *Intrusion Detection System,* IDS).

Sustav za otkrivanje upada može se opisati kao program, uređaj ili aplikacija koja nadzire sustav ili računalnu mrežu kako bi otkrila maliciozne napade ili kršenje pravila [2]. Otkrivanje upada predstavlja temelj istraživanja u grani kibernetičke sigurnosti još od 1980-ih [12]. Kako bi sustav neprestano čuvali od različitih vrsta napada, analiziraju aktivnosti i sav promet u mreži. Na taj način mogu reagirati na bilo koju sumnjivu aktivnost i prekinuti ju te tako zaustaviti nastanak štete. Sustavi koji nemaju takvu mogućnost, prijavljuju opasnosti timu za zaštitu koji zatim poduzima potrebnu akciju. Kvalitetni sustav za otkrivanje upada mora biti sposoban da s velikom preciznošću prepozna različite vrste napada u stvarnom vremenu. Mora biti fleksibilan u smislu promjene dizajna i načinu rada [12]. Upravo je ta mogućnost promjene i prilagodbe na nove tehnike ono što čini ove sustave boljom obrambenom opcijom od drugih dobro poznatih obrambenih sustava kao što su vatrozid, kontrola pristupa, enkripcija podataka, korisnička autentifikacija i mnogi drugi.

Kada govorimo o klasifikaciji ovih sustava, postoji nekoliko načina na koje ih možemo razvrstati. Dva osnovna faktora po kojima razvrstavamo sustave za otkrivanje upada su: tehnike otkrivanja upada i izvor podataka na koji će se ovi sustavi primijeniti [2]. Prema izvoru podataka sustavi za otkrivanje upada dijele se na sustave za otkrivanje upada na poslužitelju i mrežne sustave za otkrivanje napada. Podjela prema tehnikama korištenima za otkrivanje upada je sustav za otkrivanje upada temeljen na potpisu (engl. *Signature-based IDS*) i sustav za otkrivanje upada temeljen na anomaliji (engl. *Anomaly-based IDS*).

Sustavi za otkrivanje upada na poslužitelju bili su prvo područje istraživanja u okviru otkrivanja upada. Ciljano okruženje, za koje su prvi sustavi za otkrivanje upada bili dizajnirani, su mainframe računala. U takvom okruženju interakcija sa vanjskim sustavim je rijetka, jer su svi korisnici bili spojeni lokalno [13]. Prema tome, ovakvi sustavi analiziraju te prate promet i podatke unutar jednog sustava.

Mrežni sustavi za otkrivanje upada nastali su kao rezultat razvoja Interneta i njegove široke primjene. Povezivanjem više uređaja preko Interneta pojavila se potreba za sustavom koji će moći analizirati mrežu i braniti ju od napada. Oni konstantno prate sva događanja u mreži i brane ju od malicioznih napada.

Sustavi za otkrivanje upada temeljeni na potpisu još su poznati kao sustavi temeljeni na znanju jer koristi poznate potpise različitih malicioznih napada koji su prethodno otkriveni i analizirani, kako bi spriječili nove upade u sustav od strane takvih napada. Potpis koji koriste može biti predefinirani niz znakova, uzorak ili neko pravilo koje ukazuje na poznatu vrstu napada [2]. Primjer ovih sustava su antivirusni programi koji se koriste za prepoznavanje malicioznog koda.

Preciznost kod ovih sustava je velika te obavljaju dobar posao u sprječavanje poznatih napada, ali loša strana je ta što nisu upotrebljivi za zaštitu od napada koji se konstanto mijenjaju i unaprjeđuju.

Sustavi za otkrivanje upada temeljeni na anomaliji pretpostavljaju da se upad može otkriti prepoznavanjem ponašanja koje se razlikuje od normalnog i očekivanog ponašanja sustava [13]. Zbog toga se ovakvi sustavi još nazivaju sustavima temeljeni na ponašanju. Svako neuobičajeno ponašanje, koje nije u skladu sa naučenim ponašanjem za taj sustav, podiže upozorenje i smatra se pokušajem napada. Prednost ovih sustava je njihova mogućnost za otkrivanjem novih, do tada neviđenih vrsta napada, ali visoka stopa lažnih uzbuna je velika mana jer se svako ponašanje koje ne odgovara uobičajenim smatra malicioznim.

Zbog sve bržeg stvaranja novih i naprednijih vrsta malicioznih napada, tradicionalne tehnike obrane sustava pokazale su se neefikasnima. U slučaju novog malicioznog virusa, ako sustav ne zna nikakve podatke o njemu neće moći spriječiti njegovo prodiranje u sustav. Problem nastaje u rapidnom povećanju takvih slučajeva. Dok obrambeni sustav dobije ažuriranje za sve nove napade, pojavi se još nekoliko novijih verzija za koje je sustav opet nespreman. Upravo iz tog razloga, sve se više novijih tehnologija implementira u područje kibernetičke sigurnosti, od kojih strojno učenje ima najveći doprinos. Slabosti i ograničenja konvencionalnih metoda detekcije malicioznih napada mogu se unaprijediti primjenom raznih tehnika strojnog učenja. Glavna prednost tehnika strojnog učenja je njihova sposobnost učenja na temelju prošlih iskustava i velike količine prikupljenih podataka o napadima iz prošlosti [10]. Posebna vrsta strojnog učenja, koja je korisna u prepoznavanju napada jer omogućava stvaranje modela koji mogu raspoznati maliciozne napade od bezopasnih aktivnosti, je nadzirano strojno učenje. Metoda klasifikacije je najpoznatija tehnika ove vrste strojnog učenja te se može primijeniti u razvrstavanju ili predviđanju ishoda povezanog sa određenim napadom. Primjerice, metoda klasifikacije može se koristi u kibernetičkoj sigurnosti kako bi prepoznala napad uskraćivanja usluga gdje dodjeljuje oznaku „da“ ili „ne“ aktivnostima u ovisnosti o tome jesu li napad ili nisu [2].

2.2.1. Klasifikacija kao zadatak nadziranog strojnog učenja

Strojno učenje je grana umjetne inteligencije koja se može opisati kao skup algoritama i metoda koje se koriste za kreiranje modela. Tako dobiveni modeli sposobni su donositi zaključke i prepoznati uzorke [14]. Različiti algoritmi strojnog učenja kreiraju modele na temelju znanja prikupljenog iz dostupnih podataka. Model zatim koristi to znanje kako bi samostalno riješio buduće probleme o kojima nema prethodno znanje. Možemo reći, obzirom da stroj uči na podacima, što mu je dostupna veća količina podataka veća je i vjerojatnost da će preciznije riješiti problem. Strojno učenje se može podijeliti u dvije skupine: nadzirano i nenadzirano strojno učenje.

U nadziranom strojnom učenju postoji prethodno znanje o ulaznim podacima i njihovim oznakama [10]. Kod nenadziranog učenja to nije slučaj te se ono koristi u slučajevima kada grupe u koje treba raspodijeliti podatke nisu unaprijed poznate i treba provesti analizu velikog skupa podataka [15].

Podatak u strojnom učenju predstavlja činjenicu ili osnovnu informaciju na temelju kojih algoritam dolazi do šireg znanja. Podatak **x** se može smatrati primjerom (engl. *instance*) za koji želimo napraviti neko predviđanje. Svaki primjer $\mathbf{x}_i$, i = 1,…,*n* prikazan je vektorom značajki (x_i^1, … x_i^m) koji ga opisuje. Svi podaci zajedno čine skup podataka. Skup podataka od *n* podataka može se zapisati pomoću matrice **X** i vektora **y**. Matrica **X** ili matrica značajki sadrži *n* redaka i *m* stupaca. Broj *n* predstavlja broj podatkovnih primjera koji se nalazi u tom skupu podataka, a *m* je broj značajki svakog podatka. Vektor **y** ili vektor oznaka ima jedan stupac i *n* redaka te sadrži izlazne veličine za svaki podatkovni primjer iz matrice **X** kao što je ilustrirano slikom 2.1.

$$\mathbf{X} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_m^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_m^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(n)} & x_2^{(n)} & \dots & x_m^{(n)} \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{bmatrix}$$

Sl. 2.1. Matrični prikaz skupa podataka

Za ovaj rad posebno je značajna klasifikacija kao jedna od najpoznatijih tehnika nadziranog strojnog učenja zbog njezine široke primjene u obrani od malicioznih napada. Klasifikacija je tehnika nadziranog strojnog učenja kojom se treniraju algoritmi na označenom skupu podataka i potom koriste za podjelu podataka u predefinirani broj klasa. Skup podataka opisuje problem klasifikacije kada su oznake u vektoru **y** kategoričke vrijednosti. Glavni zadatak ove tehnike je izgraditi model za pronalazak funkcije koja će moći dodijeliti oznake iz unaprijed definiranog skupa svakoj instanci iz skupa podataka i ostalim novim podacima koji će joj u budućnosti biti dodijeljeni. Ta se funkcija još naziva hipotezom i ona se može smatrati početnom pretpostavkom prilikom rješavanja određenog problema strojnog učenja. Model predstavlja skup svih mogućih hipoteza koje se mogu koristiti za rješavanje danog problema. U problemu klasifikacije, taj se model naziva klasifikacijskim modelom. Svaki model sadrži veliki broj hipoteza od kojih ne daju sve jednako dobro rješenje. Iz tog razloga se koristi algoritam za

klasifikaciju, odnosno klasifikator. Njegova zadaća je pronalazak hipoteze koja najbolje opisuje problem i ima najveći postotak točno klasificiranih primjera. Taj proces otkrivanja najbolje hipoteze predstavlja treniranje klasifikatora [16]. U ovisnosti o broju klasa na koji se podaci mogu podijeliti, postoje dva tipa klasifikacije. Binarna klasifikacija je vrsta klasifikacije u kojoj postoje dvije klase u koje se podaci mogu podijeliti. Višeklasna klasifikacija je vrsta klasifikacije kod koje je broj klasa veći od dva [17]. Primjer binarne klasifikacije bio bi problem razvrstavanja elektroničke pošte u spam poštu ili normalnu poštu. Ako je pošta okarakterizirana kao spam, hipoteza će joj dodijeliti izlaznu oznaku 1, a ako je normalna pošta biti će joj dodijeljena oznaka 0. Problem koji se rješava višeklasnom klasifikacijom je prepoznavanje rukom pisanih znamenki, gdje su oznake klase znamenke od 0 do 9.

Nakon što se završi sa treniranjem modela, potrebno je izabrati onaj najbolji. U tome nam pomažu mjere za vrednovanje koje pokazuju koliko je dobar pojedini klasifikator. Jedna od najpoznatijih mjera za vrednovanje je matrica zabune. To je kvadratna matrica kojoj je broj stupaca i redaka jednak broju oznaka klasa koje se mogu dodijeliti podacima. Razlika matrice zabune i ostalih mjera za vrednovanje je u tome što iz matrice možemo dobiti potpunu sliku o modelu. Matrica zabune daje nam uvid u 4 vrijednosti, to su: istinito pozitivan rezultat (engl. *true positives*, TP), istinito negativan rezultat (engl. *true negatives*, TN), lažno pozitivan rezultat (engl. *false positives*, FP) i lažno negativan rezultat (engl. *false negatives*, FN), kao što je prikazano na slici 2.2. [10].

Matrica zabune		Predviđeno klasifikatorom	
		Klasa „1“	Klasa „0“
Stvarna klasa	Klasa „1“	TP	FN
	Klasa „0“	FP	TN

Sl. 2.2. Matrica zabune

Stvarno pozitivan rezultat prikazuje (engl. *True Positive Rate*, TPR) broj podataka koje je klasifikator točno prepoznao kao pozitivne primjere. Stvarno negativan rezultat (engl. *True Negative Rate*, TNR) prikazuje broj podataka koje je klasifikator netočno prepoznao kao pozitivne primjere. Lažno pozitivan rezultat prikazuje broj podataka koje je klasifikator netočno prepoznao kao negativan primjer. Lažno negativan rezultat prikazuje broj podataka koje je klasifikator točno prepoznao kao negativne primjere. Iako su ovi rezultati sami po sebi dobri pokazatelji koliko je određena hipoteza dobra za dani problem, iz nje se mogu izračunati još neke veličine koje nam daju dodatne informacije.

Te veličine su:

- Točnost – pokazuje udio točno klasificiranih primjera u odnosu na ukupan broj primjera

$$točnost = \frac{TP + TN}{TP + TN + FP + FN} \quad . \tag{2-1}$$

- Preciznost – pokazuje udio točno klasificiranih primjera u odnosu na sve primjere koje je klasifikator prepoznao kao pozitivne

$$preciznost = \frac{TP}{TP + FP} \quad . \tag{2-2}$$

- Odziv – pokazuje udio točno klasificiranih primjera u odnosu na sve pozitivne primjere

$$odziv = \frac{TP}{TP + FN} \quad . \tag{2-3}$$

- Specifičnost – prikazuje udio točno klasificiranih primjera u odnosu na sve negativne primjere

$$specifičnost = \frac{TN}{TN + FP} \quad . \tag{2-4}$$

- F1 mjera – predstavlja harmonijsku sredinu između odziva i preciznosti

$$F1 = 2 * \frac{preciznost * odziv}{preciznost + odziv} \quad . \tag{2-5}$$

Najjednostavnija mjera za vrednovanje klasifikacijskog modela je pogreška klasifikacije koja predstavlja postotak netočno klasificiranih primjera. Izravno iz ove veličine možemo dobiti i suprotnu joj vrijednost, točnost klasifikacije, koja predstavlja postotak točno klasificiranih podataka. Navedene mjere se prvotno koriste u problemima binarne klasifikacije, ali mogu biti i od koristi u problemima višeklasne klasifikacije.

Kod problema višeklasne klasifikacije matrica zabune je i dalje kvadratna ali većih dimenzija. Točnost se može izračunati izravno iz takve matrice, dok su mjere odziva i preciznosti definirane samo za probleme binarne klasifikacije. Kako bi se one izračunale matricu zabune prikazujemo kao niz binarnih matrica, gdje je broj tih matrica jednak broju klasa na koje se mogu

podijeliti podatkovni primjerci. Konačna vrijednost mjera odziva i preciznosti dobije se izračunavanjem prosjeka tih mjera dobivenih za svaku binarnu matricu.

2.2.2. Pregled literature

Strojno učenje je ključna komponenta kibernetičke sigurnosti za obranu od malicioznih napada s mnogobrojnim tehnikama korištenim za obranu od napada koji konstantno napreduju. Metode za otkrivanje malicioznih napada temeljene na strojnom učenju istaknule su se svojom kvalitetom u odnosu na tradicionalne metode otkrivanja. Najveći doprinos tome je njihova sposobnost prilagođavanja na nove napade i mogućnost prepoznavanja do sada još neviđenih napada.

U svom radu Fraley i Cannady [11] istraživali su mogućnost korištenja strojnog učenja za rješavanje problema upravljanja sigurnosnim događajima. Naveli su kako broj malicioznih napada na tvrtke može dostići do visokih tri milijuna napada po satu. Ova činjenica dobiva dodatnu težinu ako znamo da većina navedenih tvrtki ima nedovoljan broj ljudi specijaliziranih za rješavanje tih problema. Broj sati koji je u prosjeku potreban za timove da riješe ove probleme prelazi 2 000, a mogućnost tima je između 96 i 144 sata. Iz tog razloga, autori su zajedno sa timom za obranu odlučili implementirati strojno učenje u sustav upravljanja sigurnosnim događajima. Nakon dogovorenih ciljeva, prikupili su sve podatke o prošlim napadima i svrstali ih u pet kategorija. Kao najbolju tehniku prikladnu za ovaj problem odabrali su neuronske mreže (engl. *neural network*, NN) i duboke neuronske mreže (engl. *deep neural network*). Cilj je bio stvoriti modele koji će moći prepoznati različite napade, svrstati ih u jednu od pet određenih kategorija i odrediti njegovu ozbiljnost. Jednostavnije napade model je trebao sam riješiti, a složenije proslijediti stručnjacima. Kao rezultat, stvoren je model koji je uspješno klasificirao 90% svih napada. Implementirajući ovaj model u svoj obrambeni sustav, količina vremena potrebna za rješavanje problema trebala bi se smanjiti za 78%.

Torres, Comesana i Garcia - Nieto u svom radu [15] dali su pregled istraživanja koja su se bazirala na upotrebi nekoliko tehnika strojnog učenja u kibernetičkoj sigurnosti. Maliciozne napadi na koje su se fokusirali su prepoznavanje spam poruka, malicioznih programa i phishing-a. Za prepoznavanje spam poruka korišteni su razni Bayesovi algoritmi (engl. *naive Bayes*). Osim ovih algoritama, koji su najpopularniji u prepoznavanju spam poruka, korišten je i stroj potpornih vektora (engl. *support vector machine,* SVM). Navode kako se u početku problem prepoznavanja malicioznih programa pokušao riješiti tehnikama nenadziranog strojnog učenja. Zbog loših rezultata ukazala se potreba za novim tehnikama. Korišteni su algoritmi strojnog učenja kao što su: stabla odluke (engl. *decision tree*, DT), umjetne neuronske mreže (engl. *artificial neural network*, ANN), Naive Bayes algoritam, stroj potpornih vektora i metoda *k*-najbližih susjeda (engl.

k-nearest neighbours, *k*-NN). Zadnji maliciozni napad koji su prikazali je phishing. Tehnike koje su imale najviše uspjeha u obrani od ovih napada su algoritam nasumične šume (engl. *random forest*) i logistička regresija (engl. *logistic regression*).

Ahsan, Nygard, Gomes et al. [2], dali su prikaz tehnika strojnog učenja korištenih u kibernetičkoj sigurnosti kako bi ublažili maliciozne napade, a s ciljem prikazivanja njihove efikasnosti. Ukazuju na važnost podataka iz kibernetičke sigurnosti jer su oni temelj strojnog učenja. Razumijevanje podataka je ključno, a njihovom analizom se može dobiti uvid u razne napade te prema tome odabrati prikladan način obrane. Metode nadziranog strojnog učenja koje su najkorisnije za klasifikaciju je li nešto napad ili ne su: Naive Bayes algoritam, stroj potpornih vektora, stablo odluke, algoritam *k*-najbližih susjeda i logistička regresija. Korišteni podaci prikupljali su se četiri mjeseca na bežičnoj mreži u kampusu Dartmouth Sveučilišta. Algoritam nasumične šume koristio se za otkrivanje upada i zlouporabe. Glavni problem, prema autorima, koji još uvijek predstavlja veliki problem za strojno učenje u kibernetičkoj sigurnosti je nedostatak podataka. Iako je količina podataka o starijim malicioznim napadima poprilično velika, problem predstavljaju podaci o novijim napadima kojih nema ni približno dovoljno. Još je jedan problem vezan uz podatke, a to je kvaliteta skupova podataka. Oni mogu biti neuravnoteženi, nepotpuni, mogu sadržavati puno nepotrebnih podataka i krivih mjerenja. Ti nedostatci dovode do manje kvalitetnih modela ili u potpunosti onemogućuju pronalazak kvalitetnog modela sa zadovoljavajućim performansama. Problem kvalitete skupova podataka može se riješiti inženjeringom značajki. Autori kažu kako je i nedostatak u tome što su najkorišteniji oblici za otkrivanje upada sustavi temeljeni na potpisu koji su korisni samo za poznate napade, ali ne pružaju nikakvu zaštitu od novih napada.

D'hooge, Wauters, Volckaert et al. u svom radu [12] opisali su proces analize CICIDS2017 skupa podataka i prikazuju dobivene rezultate. Za vrednovanje CICIDS2017 skupa podataka korišteno je devet tehnika nadziranog strojnog učenja. Među njima su algoritmi temeljeni na stablu odluke, na metodi najbližih susjeda, na stroju potpornog vektora, logistička regresija i tehnike ansambla. Algoritmi temeljeni na stablu odluke imali su najbolje rezultate za sva tipove napada. Jedini drugi algoritam koji je imao približno dobru razinu preciznosti je algoritam *k*-najbližih susjeda koji je pokazivao dobre rezultate za pet od sedam vrsta napada. Ostali algoritmi se nisu pokazali toliko dobrim, a čak i za napade na kojima su pokazali donekle dobre rezultate nisu bili ni približno dobri kao ostali algoritmi.

Shaukat, Luo, Varadharajan et al. u svom radu [10] predstavili su korištenje tehnike strojnog učenja za poboljšanje kibernetičke sigurnosti u raznim sustavima. Najčešći problemi za koje se ovi algoritmi koriste su detekcija prijevare, upada, spam poruka i malicioznih programa.

Autori su dali pregled često korištenih skupova podataka. Algoritmi stroj potpornog vektora, stablo odluke i nasumične šume imaju najbolju preciznost i točnost kod problema prepoznavanja upada. Za problem otkrivanja upada najlošije rezultate su dali algoritmi umjetne i duboke neuronske mreže. Prema svemu navedenom, autori predlažu da se za pomoć sustavima za otkrivanje upada treba koristiti jedan od sljedećih algoritama: stroj potpornih vektora, stablo odluke, Naive Bayes ili nasumična šuma. Kada je u pitanju otkrivanje malicioznih programa, najbolje je koristiti stoj potpornih vektora ili stablo odluke, a za detekciju spam poruka najbolji algoritmi su nasumična šuma i Naive Bayes. Na kraju, autori navode da iako se primjena strojnog učenja za detekciju malicioznih napada pokazala puno učinkovitijom obranom od tradicionalnih metoda, još uvijek postoje brojne prepreke. Nedostatak kvalitetnih metoda za vrednovanje rezultata i manjak skupova podataka sa podacima o novijim napadima predstavljaju veliki problem. Nedostatni rezultati prilikom upotrebe istih algoritama na istom skupu podataka otežavaju odabir najboljeg modela. Povezani problem s ovima je i loša kvaliteta skupova podataka u kojima se i dalje pojavljuju pogreške, duplicirani ili nepotrebni podaci, te nebalansiranost broja podataka koji pripadaju pojedinoj klasi.

2.3 Kritički osvrt

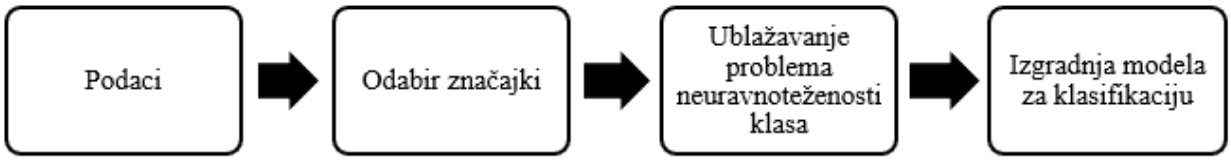
Pregledom literature vidljivo je kako strojno učenje ima značajan doprinos u području kibernetičke sigurnosti. Otkako su se počele koristiti različite tehnike strojnog učenja za detekciju malicioznih napada razina sigurnosti se podigla na novu razinu. U određenim situacijama, u kojima dobiveni modeli ne mogu riješiti sve probleme samostalno, opet mogu smanjiti broj problema za ljudske stručnjake na minimum. Prema rezultatima je vidljivo kako su sustavi koji koriste neku od tehnika strojnog učenja puno bolji od tradicionalnih rješenja. Velika brojnost i raznovrsnost tih tehnika omogućuje da se za iste probleme koriste različiti pristupi ovisno o parametrima koji se žele optimizirati. Prednost brojnosti je i osiguravanje da ako jedna tehnika ne daje zadovoljavajuće rezultate vjerojatno postoji neka druga koja ju može zamijeniti. Svaka tehnika ima područje u kojem je bolja od ostalih te svaka ima svoju primjenu u jednom od raznih sustava kibernetičke sigurnosti. Stroj potpornih vektora se u najviše slučajeva spominje kao optimalno rješenje i ima najširi spektar primjene. Problem se pojavljuje u tome kada shvatimo da se tehnike strojnog učenja ne primjenjuju samo na strani obrane. Strojno učenje u velikoj je mjeri pomoglo i napadačima koji kreiraju ove napade, što je vidljivo iz sve većeg broja novih vrsta napada. Napadi postaju sve sofisticiraniji i sve brže se kreiraju do sada neviđeni oblici napada. Nedostatak podataka o njima je najveći problem za strojno učenje. Ono je zasnivano na podacima te je njihova brojnost i kvaliteta krucijalna za izradu dobrih modela. Dostupni podaci također nisu

savršeni. Mnogi skupovi podataka imaju probleme kao što su duplicirani podaci ili nedostatne informacije, numeričke vrijednosti imaju lošu distribuciju, neki podaci unutar skupova su nepotrebni i još mnoštvo drugih problema koji utječu na kvalitetu izrađenih modela. Postoje tehnike za ublažavanje ovih modela. Informacije koje nedostaju mogu se zamijeniti srednjom vrijednosti ili najčešćom vrijednosti, duplicirani podaci se mogu ukloniti, čak se mogu stvoriti i novi podaci kombiniranjem informacija iz više postojećih. Ipak, to ne predstavlja dovoljno dobro rješenje kako bi se ta prepreka uklonila. Dodatan problem je i razlika u rezultatima. Ponekad kreirani model može dati dva različita rješenja na istom skupu podataka, što može dovesti do zabune prilikom odabira.

Navedeni radovi daju dobar pregled problema i pokazuju kako strojno učenje može pripomoći u zaštiti od malicioznih napada, ali imaju i određene nedostatke. Većina radova prikazuje različite metode strojnog učenja te upućuje koja je metoda najbolja za određeni problem, ali nedostaje detaljniji opis. Nedostaju opsežne eksperimentalne analize koje nam daju dublji uvid u kvalitete i nedostatke pojedine metode. Rijetko se navodi na koji način su odabrani određeni parametri u modelu niti kako se došlo do optimalnih parametara. Temeljno vrednovanje performansi također nije detaljno prikazano. Još jedan znatni nedostatak je manjak tehnika predobrade skupova podataka. Navedene su samo osnovne tehnike rukovanja nepostojećim i dupliciranim vrijednostima. Postoje mnoge druge tehnike koje se koriste u predobradi, a mogu imati znatni utjecaj na poboljšanje performansi modela.

3. OSTVARENO PROGRAMSKO RJEŠENJE

Pregledom literature utvrđen je uobičajeni tijek učenja iz skupova podataka koji opisuju maliciozne napade. Na temelju tog tijeka mogu se izdvojiti glavne procedure koje se koriste i imaju veliki značaj. Kako bi se lakše odredile važnije procedure, razvijen je okvir koji sadrži glavne korake za prepoznavanje malicioznih napada. Procedure obuhvaćene ovim okvirom su prikupljanje podataka, predobrada i kreiranje modela. Svaka procedura ima određenu ulogu u ovom procesu te je važno odabrati odgovarajuću proceduru u ovisnosti o tipu problema i karakteristikama skupa podataka.



Sl. 3.1. Koraci učenja iz skupova podataka koji opisuju maliciozne napade

Na slici 3.1. prikazan je tijek učenja iz podataka koji opisuju maliciozne napade s glavnim koracima. Prvi korak podrazumijeva prikupljanje podataka. Kao što je već rečeno, temelj strojnog učenja su podaci te je dobro da ih bude što više te da su što kvalitetniji. Brojni skupovi podataka koji opisuju ove probleme javno su dostupni i mogu se iskoristiti u ovom radu. Iduća dva koraka predstavljaju predobradu podataka. Predobrada podataka sastoji se od različitih tehnika kojima se skup podataka želi pojednostaviti i poboljšati. Cilj predobrade je olakšati algoritmima proces učenja iz podataka. U predobradu podataka spadaju dvije važne tehnike, a to su odabir značajki i rukovanje s neuravnoteženim skupovima podataka. Odabir značajki jedna je od najvažnijih tehnika u predobradi podataka te je postala neizostavni dio u procesu strojnog učenja. Ova tehnika podrazumijeva odabir samo onih značajki koje doprinose poboljšanju modela, a ostale nebitne značajke izbacuje iz skupa podataka [20]. Odabir značajki, kao što ime kaže, procedura je kojom se odabire podskup značajki kako bi se smanjio njihov broj i uklonile značajke koje ne doprinose poboljšanju modela. Na taj se način može poboljšati sposobnost predviđanja modela bez da se narušavaju njegove performanse [19]. Rukovanje s neuravnoteženim skupom podataka podrazumijeva odabir procedura koje rješavaju problem neuravnoteženosti skupa podataka. U takvim skupovima oznake klasa su neravnomjerno zastupljene, tj. oznaka jedne klase značajno je manje zastupljena za razliku od oznaka drugih klasa [16]. Nakon završetka ova dva koraka predobrade podataka može se započeti sa izradom modela za klasifikaciju malicioznih napada.

Potrebno je odabrati adekvatne algoritme klasifikacije koji će se koristiti za treniranje modela. Broj dostupnih algoritama za klasifikaciju poprilično je velik, ali nije svaki algoritam podjednako dobar za svaki problem. Stoga je potrebno odabrati algoritam koji će dati zadovoljavajuće rezultate za problem koji pokušavamo riješiti. Nakon odabira samog algoritma potrebno je podesiti njegove parametre kako bi model imao što bolju moć generalizacije. Nakon odabira željenog algoritma i podešavanja parametara, prelazi se na treniranje modela. Dobiveni model se zatim vrednuje te se u slučaju loših performansi vraća na postupak treniranja gdje se pokušava doraditi kako bi se postigla veća kvaliteta. Ovaj korak se ponavlja sve dok model ne prikaže dobre performanse prilikom vrednovanja.

3.1. Odabir algoritama za klasifikaciju

Strojno učenje ima veliki doprinos u području kibernetičke sigurnosti i prepoznavanju različitih vrsta malicioznih napada [2]. Posebice su korisni algoritmi klasifikacije koji se koriste za klasificiranje različitih napada ili predviđanje nekih budućih napada [20]. Od svih dostupnih klasifikacijskih algoritama nisu svi prikladni za upotrebu u kibernetičkoj sigurnosti. Dobro poznati algoritmi koji se koriste za prepoznavanje malicioznih napada su Naive Bayes, *k*-NN, stablo odluke, nasumična šuma, i logistička regresija [20].

Naive Bayes algoritam je popularan za prepoznavanje novijih oblika malicioznih napada [18]. Temelji se na Bayesovom teoremu i koristi uvjetnu vjerojatnost kako bi odredio pripadnost objekta određenoj klasi. Ovaj algoritam pretpostavlja kako su značajke u skupu podataka neovisne jedna o drugoj te ne postoji nikakva veza među njima. Kako bi se izračunalo kolika je vjerojatnost da pojedini podatkovni primjer pripada određenoj klasi koristi se uvjetna vjerojatnost, prethodna vjerojatnost te marginalna vjerojatnost.

Algoritam *k* najbližih susjeda koristi metriku sličnosti kako bi odredio klasu podatkovnog primjera. Metrika sličnosti prema kojoj se određuje kojoj klasi treba pripasti podatak je najčešće Euklidska udaljenost [21]. Tako algoritam pridružuje nove podatke klasi s čijim članovima ima najviše sličnih karakteristika. Parametar *k* odlučuje s koliko se će se susjednih podatka vršiti usporedba te se onda novi podatak pridružuje klasi kojoj pripada većina od *k* susjeda [22]. Parametar *k* smatra se hiperparametrom modela jer zahtjeva da se unaprijed odredi. Njegov odabir ima značajan utjecaj na performanse modela te se za svaki skup podataka mora izabrati zasebno kako bi model imao zadovoljavajuće performanse. Za lakši odabir mogu se koristiti razne tehnike njegovog podešavanja.

Stablo odluke je algoritam strojnog učenja koji kreira klasifikacijski model u obliku stabla grananjem na temelju raznih faktora. Ovaj algoritam sastoji se od grana i čvorova te predstavlja

binarno stablo. Korijenski čvor je početni čvor stabla iz kojeg izlaze grane. Svaki čvor predstavlja neku vrstu odabira na temelju vrijednosti značajke, a izlaz iz čvora su dvije grane koje predstavljaju vrijednost tog atributa [23]. Krajnji čvorovi se nazivaju listovi te iz njih ne izlaze grane i oni predstavljaju klase kojima podaci moraju biti dodijeljeni. Parametar ovog algoritma je dubina stabla koji raste s povećanjem broja podataka.

Klasifikacijski algoritam nasumične šume sličan je stablu odluke. Može se smatrati njegovim proširenjem te ovaj algoritam koristi više stabala odluke kako bi došao do zaključka [2]. Ovaj algoritam koristi se za identifikaciju korisnika na internetu, otkrivanje upada i prepoznavanje neželjenih poruka [15]. Nakon što se izgradi više stabala odluke, svako s nasumično odabranim uzorcima iz skupa podataka, donosi se konačna odluka glasanjem između stabala. Podatak će biti dodijeljen u onu klasu u koju ga je najviše stabala odluke klasificiralo.

Logistička regresija nastoji pronaći granicu između dvije klase. Najčešće ta granica je predstavljena pravcem koji odjeljuje dvije klase binarne klasifikacije. U idealnom slučaju svi primjeri jedne klase trebali bi se nalaziti na jednoj strani toga pravca, a svi primjeri druge klase na drugoj strani. Kao rezultat ovog algoritma dobije se kolika je vjerojatnost da podatak pripada određenoj klasi [21]. Zbog toga dobivene vrijednosti ovim algoritmom su između jedan i nula.

3.2. Predobrada skupova podataka odabirom značajki

Skupovi podataka s vremenom postaju sve veći te se sama dimenzionalnost povećava, što znači da je broj značajki kojima se opisuju podaci sve veći. Velika količina podataka uvelike doprinosi razvoju boljih modela strojnog učenja, ali njihova brojnost može biti i rizik jer skupovi podataka s velikom dimenzionalnosti otežavaju proces učenja modela. Prisutnost velikog broja redundantnih i nepotrebnih značajki uzrokuje veću složenost modela što sa sobom povlači rizik loše generalizacije modela [24]. Nemaju sve značajke jednako dobar doprinos klasificiranju podataka te neke čak narušavaju i otežavaju situaciju jer kreiraju nepoželjne karakteristike u skupu podataka. Kako bi se riješio ovaj problem koristi se procedura odabira značajki budući da odabire podskup samo onih značajki koje doprinose i poboljšavaju proces klasifikacije, a sve ostale značajke koje su nepotrebne zanemaruje. Odabir manjeg broja značajki obično rezultira smanjenjem dimenzionalnosti problema, a metode koje se koriste za postizanje tog cilja su filtri i omotači.

Filtri kreiraju podskup značajki iz skupa svih značajki na temelju statističkih mjera kao što su Pearsonov koeficijent korelacije, zajednička informacija i ANOVA F-vrijednost [25]. Oni računaju jednu od odabranih metrika za svaku značajku i onda ih poredaju prema najboljem rezultatu. Na kraju kreiraju podskup od unaprijed određenog broja značajki koji se zadaje kao

parametar ovim metodama. Jedan od rizika ove metode je odbacivanje značajki koje same po sebi nemaju veliki doprinos, ali u kombinaciji s drugim značajkama mogu olakšati proces klasifikacije. Pearsonov koeficijent korelacije mjeri stupanj linearne korelacije između dvije kontinuirane značajke. Vrijednost ovog koeficijenta leži između [-1, 1]. Vrijednosti 1 i -1 upućuju na to da postoji jaka pozitivna ili negativna korelacija između značajki, a vrijednost 0 označuje da nema korelacije među njima [25]. Filter koji se koristi kada su u pitanju kategoričke značajke je zajednička informacija. Ova mjera daje rezultat o količini informacija koju jedna informacija daje o drugoj, odnosno mjeri njihovu ovisnost. Veća vrijednost zajedničke informacije ukazuje na to da dvije značajke imaju visoku međusobnu ovisnost, a ako je zajednička informacija nula značajke su međusobno potpuno neovisne. Još jedna popularna mjera koja se koristi je ANOVA F-vrijednost, a koristi se kada su značajke kontinuirane, a oznake klasa kategoričke. ANOVA F-vrijednost prikazuje jesu li srednje vrijednosti dviju ili više skupina statistički različite. Uspoređuje varijancu između skupina s varijancom svake skupine kako bi se utvrdilo koliki je doprinos pojedine značajke.

Druga metoda koja se koristi za odabir značajki su omotači. Podskup značajki dobiven pomoću omotača u većini slučajeva ostvaruje bolje performanse modela od podskupa značajki dobivenog metodom filtriranja, ali su zato vremenski složeniji. Za razliku od filtra koji proces odabira značajki odvajaju od treniranja modela, omotači spajaju ta dva procesa u jedan. Omotači se u mogu smatrati kao mehanizmi pretraživanja, koji pretražuju skup podataka kako bi pronašli optimalni podskup značajki. Oni se omotavaju oko algoritma te ga koriste u procesu pretrage za vrednovanje performansi podskupa. Proces se zatim ponavlja više puta, sve dok se ne dosegne unaprijed definirani kriterij zaustavljanja ili performanse počnu naglo opadati [26]. Jedan od popularnijih omotača zbog svoje jednostavnosti je omotač slijedne pretrage unaprijed (engl. *sequential forward selection*, SFS). Ovaj omotač započinje s praznim podskupom te dodaje značajku po značajku, a nakon svake nove dodane značajke vrjednuju kvalitetu skupa. Dodavanjem značajki se prestaje u trenutku kada kvaliteta podskupa prestane rasti. Glavni je nedostatak ove metode nemogućnost uklanjanja značajki koje postanu nepotrebne nakon dodavanja drugih značajki [27].

Metode odabira značajki primijenjene su u programskom rješenju kako bi se smanjila složenost modela, izbjegla opasnost od prenaučenosti i povećale performanse modela. Prvo se pomoću filtera sa različitim statističkim mjerama, a zatim i pomoću SFS omotača za svaki skup podataka pronalazi podskup najboljih značajki. Nakon toga se vrednuje svaki odabrani algoritam strojnog učenja na dobivenim podskupovima kako bi se dobio uvid u promjenu performansi.

3.3. Rukovanje s neuravnoteženim skupovima podataka

Algoritmi strojnog učenja pretpostavljaju da je broj podataka svake klase otprilike podjednak, ali u većini slučajeva to nije istina. U mnogim situacijama iz stvarnoga života distribucija podatkovnih primjera je iskrivljena te broj primjera jedne klase znatno brojniji. Zbog toga dolazi do iskrivljenih rezultata jer će algoritam biti pristran većinskoj skupini. Istovremeno, podaci manjinske klase su često oni koji su od većeg značenja jer je stjecanje znanja o tim događajima puno važnije. Iz tog razloga, potrebno je dizajnirati model koji može efikasno riješiti ovaj problem, a poznat je kao problem neuravnoteženosti skupova [28]. Većina skupova podataka koji se koriste za treniranje modela za otkrivanje malicioznih napada su neuravnoteženi skupovi što predstavlja prepreku primjeni strojnog učenja u kibernetičkoj sigurnosti. Najčešće je broj malicioznih napada puno manji od broja normalnih aktivnosti koje detektiraju sustavi za otkrivanje upada kao i sustavi za rješavanje internih prijetnji [29]. Postoji mnogo različitih postupaka za rukovanje neuravnoteženim skupovima podataka od kojih su dva najčešća nasumično preuzorkovanje i nasumično poduzorkovanje. Nasumično preuzorkovanje, kao što ime predlaže, kopira nasumično odabrane primjerke iz manje zastupljene klase te ih dodaje u skup podataka. S druge strane, metoda nasumičnog poduzorkovanje uzima slučajno odabrane primjerke iz većinske klase te ih uklanja iz skupa podataka. Obje metode se ponavljaju dok skup ne postigne željeni stupanj uravnoteženosti [16]. U programskom rješenju, metode nasumičnog preuzorkovanja i poduzorkovanja primjenjuju se na skupove podataka uz izmjenu vrijednosti uzorkovanja kako bi se pronašao najbolji omjer i uklonila neuravnoteženost skupa. Nakon toga, vrednuju se performanse modela treniranih na novonastalim skupovima kako bi se procijenila učinkovitost ovih metoda.

4. EKSPERIMENTALNA ANALIZA I REZULTATI

U ovom poglavlju dan je pregled korištenih skupova podataka te upotrebom tehnika navedenih u prošlom poglavlju provedena je njihova analizu. Cilj je kroz provedeni postupak uočiti tijek učenja iz skupova podataka koji opisuju maliciozne napade te prikazati rezultate odabranih algoritama strojnog učenja. Korištenjem tehnika predobrade pokušava se poboljšati efikasnost svakog modela i utvrditi koji je algoritam prikladan za rješavanje problema prepoznavanja malicioznih napada.

4.1. Postavke i metodologija eksperimentalne analize

Korišteni skupovi podataka, kao neke njihove karakteristike prikazani su u tablici 4.1. Skupovi podataka preuzeti su sa Internet stranice Kaggle [30], UCI [31] te stranice Sveučilišta New Brunswick [32] i UNSW Sydney [33]. Većina je ovih skupova poznata te se koristi za potrebe strojnog učenja kako bi se poboljšali sustavi kibernetičke sigurnosti. Gotovo svi imaju dvije klase od kojih jedna predstavlja normalne aktivnosti, a druga vrstu malicioznog napada ovisno o sustavu za koji se koristi određeni skup podataka. Problem neuravnoteženosti klasa je prisutan u gotovo svakom skupu. Stupanj neuravnoteženosti računa se kao omjer broja primjeraka većinske klase i broja primjeraka manjinske klase. Kod višeklasnih skupova računa se prosjek omjera svih parova klasa. Kod nekolicine skupova stupanj neuravnoteženosti nije toliko značajan što ukazuje na podjednak broj primjeraka svake klase. Ostali imaju znatno veći stupanj neuravnoteženosti što predstavlja problem koji se mora ublažiti.

Tablica 4.1. Skupovi podataka

Naziv	Vrsta napada	Broj instanci	Broj značajki	Broj klasa	Stupanj neuravnoteženosti
DARPA	DoS	4 554 344	4	2	1.51
KDD99	DoS, ubrizgavanje URL-a, mrežna krađa identiteta…	494 020	42	23	3530.99
NSL-KDD	DoS, ubrizgavanje URL-a, mrežna krađa identiteta…	148 517	42	40	37.27
KYOTO	DDoS	303 849	24	3	50644.17
Malware	Virus, crv, trojanski konj	100 000	35	2	1
DREBIN	Različiti maliciozni napadi operacijskog sustava Android	15 036	215	2	1.70
ISCXIDS2012	DoS, ubrizgavanje URL-a, mrežna krađa identiteta…	171380	21	2	44.39
CICIDS2017	DDoS	225 745	79	2	1.31
DS2OS	Ucjenjivački, reklamni i špijunski programi	357 952	13	8	225.23
IMPACT	DoS, mrežna krađa identiteta…	19 940	9	20	11.44
UNSW-NB15	DDoS, ubrizgavanje URL-a, mrežna krađa identiteta…	257 673	45	2	1.77
CIC-DDOS2019	DDoS	300 000	88	19	787.62

Algoritmi klasifikacije korišteni za treniranje modela na ovim skupovima podataka su: stablo odluke, *k*-najbližih susjeda, logistička regresija, nasumična šuma i Naive Bayes. Treniranje i vrednovanje algoritama ponavlja se 10 puta za svaki algoritam svakom skupu podataka. Svaki skup podataka podijeljen je na skup za treniranje i skup za testiranje u omjeru 80:20 %. Prilikom upotrebe SFS omotača skup za treniranje dodatno je podijeljen na skup za treniranje i skup za validaciju u omjeru 65:35 % kako bi se testirali podskupovi dobiveni omotačem. Performanse svakog algoritma vrednovane su nakon procesa treniranja. Točnost i F1 mjera su korištene u tu svrhu te dodatno TPR i TNR koji se računaju iz matrice zabune. Konačne vrijednosti za svaku mjeru izražene su kao prosječna vrijednost svih vrijednosti dobivenih iz 10 iteracija eksperimenta, pri čemu se u svakoj iteraciji provodi nasumična podjela skupova podataka. Ove mjere pomažu u donošenju odluke koji algoritam je prikladan za koji skup. Svaki algoritam sadrži brojne hiperparametre koji se moraju podesiti prije procesa učenja modela te oni utječu na performanse modela. Potrebno ih je podesiti kako bi algoritam ostvario zadovoljavajuće performanse. Jedna od metoda koja je popularna, a korištena je u ovome radu je pretraživanje po mreži(engl. *grid search*). Za svaki parametar se definira nekoliko vrijednosti, nakon toga ova metoda isprobava sve kombinacije. Ona kombinacija hiperparametara za koju model ostvari najbolje performanse se odabire za daljnje treniranje modela na navedenim skupovima podataka. Hiperparametri korištenih algoritam za svaki skup podataka prikazani su u tablici 4.2. Nedostajuće vrijednosti iz skupova su obrisane te su sve kategoričke i ordinalne značajke pretvorene u numeričke upotrebom tehnike kodiranja oznaka (engl. *label encoding*). Nakon toga sve značajke standardizirane su upotrebom „min-max“ normalizacije na vrijednosti između 0 i 1.

4.2 Hiperparametri algoritama

Klasifikacijski algoritmi	**Hiperparametri**
Stablo odluke	kriterij za podjelu ∈ {gini, entropija} maksimalna dubina ∈ {5, 15, 20} strategija podjele ∈ {nasumična, najbolja}
k-najbližih susjeda	algoritam za pretraživanje susjeda ∈ {automatski} broj susjeda ∈ {2, 4, 5} težina susjeda ∈ {udaljenost}
Logistička regresija	C ∈ {1, 2} prilagodba presjeka ∈ {Da, Ne} kazna ∈ {l2} optimizacijski algoritam ∈ {lbfgs}
Nasumična šuma	maksimalna dubina ∈ {5, 10} minimalni broj uzoraka u list ∈ {1, 2} broj stabala ∈ {100, 150, 200}
Naive Bayes	izjednačavanje varijance ∈ {0.000000001, 0.0000001, 0.0001, 0.01, 0.1, 1}

Prilikom primjene filtera isprobano je nekoliko različitih vrijednosti za broj značajki koje filter treba izdvojiti. Isto tako za vrijeme korištenja nasumičnog uzorkovanja predefinirano je više različitih vrijednosti za omjer uzorkovanja. Vrijednost strategije uzorkovanja određuje postotak na koji se mora smanjiti broj primjeraka većinske klase u odnosu na broj primjeraka manjinske klase kada je u pitanju tehnika poduzorkovanja. Kada se koristi tehnika preuzorkovanja omjer strategije određuje postotak na koji se broj primjeraka manjinske klase mora povećati u odnosu na broj primjeraka većinske klase. Ove vrijednosti ručno su mijenjanje prilikom pokretanja svake nove iteracije treniranja modela.

4.2. Usporedba klasifikatora

Svaki klasifikacijski algoritam optimiziran je podešavanjem hiperparametara za svaki skup podataka te su uspoređeni najbolji algoritmi. Postavke hiperparametara prikazane su u tablicama u Prilogu. Na slici 4.1. prikazan je dijagram pravokutnika (engl. *box plot*) koji za svaki klasifikacijski algoritam uspoređuje vrijednosti F1 mjere te se pomoću njega može dobiti općeniti uvid u performanse algoritama. Algoritmi stablo odluke, *k*-najbližih susjeda, logistička regresija i nasumična šuma imaju usko grupiranje vrijednosti F1 mjere blizu 1.0 što ukazuje na njihove visoke performanse ostvarene na svakom skupu podataka. Stablo odluke se može izdvojiti kao najbolji algoritam, jer preostali algoritmi imaju povremeno lošije performanse na što je vidljivo iz stršećih vrijednosti. Nasuprot ovim algoritmima Naive Bayes algoritam ima veći raspon vrijednosti F1 mjere i ostvaruje lošije performanse na gotovo svim skupovima podataka. Njegovi rezultati su bili primjetno slabiji, a u IMPACT skupu podataka je imao najmanje vrijednosti mjera korištenih za vrednovanje kao što je vidljivo u tablici 4.8. Klasificiranje napada na Drebin UNSW-NB15 skupovima podataka manje je uspješnije nego kod ostalih skupova. Drebin sadrži informacije o malicioznim napadima na operacijskom sustavu Androidu. Kao najprikladniji algoritam na tom području pokazao se algoritam *k*-najbližih susjeda, a najlošiji je još jednom bio Naive Bayes. UNSW-NB15 skup podataka ima prikupljene podatke o napadima u mrežnom prometu. Naive Bayes ima i kod ovog skupa najlošije performanse dok je algoritam koji je ostvario najbolje performansama stablo odluke. Analizom vrijednosti metrika vidljivo je da točnost i TNR imaju veće vrijednosti od mjera F1 i TPR u slučajevima kada se metrike razlikuju. Razlog tome je neuravnoteženost podataka pa točnost i TNR lakše postižu visoke vrijednosti. Kod takvih skupova podataka istrenirani model većinu primjera može klasificirati kao primjere većinske klase jer će u većini slučajeva predikcija biti točna zbog većeg broja primjera iz te klase. Mjera TNR visoke rezultate postiže jer se u pravilu većinska klasa smatra negativnom klasom, a pošto se TNR fokusira na negativne primjere, lakše je točno klasificirati primjere iz te klase. Na slikama

4.2.,4.3.,4.4. i 4.5. prikazane su sve vrijednosti metrika u ovisnosti na omjer neuravnoteženosti pomoću dijagrama raspršenosti.

Tablica 4.3. Performanse algoritama na DARPA skupu podataka

Algoritam	**Točnost**	**F1 mjera**	**TPR**	**TNR**
Stablo odluke	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Logistička regresija	0.76 ± 0.01	0.83 ± 0.01	0.96 ± 0.01	0.46 ± 0.01
Nasumična šuma	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Naive Bayes	0.94 ± 0.01	0.95 ± 0.01	0.95 ± 0.01	0.93 ± 0.01

Tablica 4.4. Performanse algoritama na KDD99 skupu podataka

Algoritam	**Točnost**	**F1 mjera**	**TPR**	**TNR**
Stablo odluke	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Logistička regresija	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Nasumična šuma	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Naive Bayes	0.98 ± 0.01	0.97 ± 0.01	0.96 ± 0.01	0.98 ± 0.01

Tablica 4.5. Performanse algoritama na NSL-KDD skupu podataka

Algoritam	**Točnost**	**F1 mjera**	**TPR**	**TNR**
Stablo odluke	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Logistička regresija	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.98 ± 0.01
Nasumična šuma	0.98 ± 0.01	0.97 ± 0.01	0.98 ± 0.01	**0.99 ± 0.01**
Naive Bayes	0.83 ± 0.01	0.82 ± 0.01	0.83 ± 0.01	0.93 ± 0.01

Tablica 4.6. Performanse algoritama na KYOTO skupu podataka

Algoritam	**Točnost**	**F1 mjera**	**TPR**	**TNR**
Stablo odluke	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
k-najbližih susjeda	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Logistička regresija	0.98 ± 0.01	0.98 ± 0.01	0.98 ± 0.01	0.99 ± 0.01
Nasumična šuma	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Naive Bayes	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01

Tablica 4.7. Performanse algoritama na Malware skupu podataka

Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
k-najbližih susjeda	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	1.00 ± 0.00
Logistička regresija	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Nasumična šuma	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
Naive Bayes	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.98 ± 0.01

Tablica 4.8. Performanse algoritama na DREBIN skupu podataka

Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	0.97 ± 0.01	0.98 ± 0.01	0.97 ± 0.01	**0.99 ± 0.01**
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.98 ± 0.01**	**0.99 ± 0.01**
Logistička regresija	0.98 ± 0.01	0.98 ± 0.01	0.98 ± 0.01	**0.99 ± 0.01**
Nasumična šuma	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	**0.99 ± 0.01**
Naive Bayes	0.82 ± 0.01	0.82 ± 0.01	0.82 ± 0.01	0.92 ± 0.01

Tablica 4.9. Performanse algoritama na ISCXIDS2012 skupu podataka

Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Logistička regresija	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Nasumična šuma	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
Naive Bayes	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**

Tablica 4.10. Performanse algoritama na CICIDS2017 skupu podataka

Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	0.99 ± 0.01
k-najbližih susjeda	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	0.99 ± 0.01
Logistička regresija	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	0.98 ± 0.01
Nasumična šuma	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	**1.00 ± 0.00**
Naive Bayes	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**	0.97 ± 0.01

Tablica 4.11. Performanse algoritama na DS2OS skupu podataka

Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
k-najbližih susjeda	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
Logistička regresija	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Nasumična šuma	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01
Naive Bayes	0.98 ± 0.01	0.97 ± 0.01	0.98 ± 0.01	0.99 ± 0.01

Tablica 4.12. Rezultati algoritama na IMPACT skupu podataka

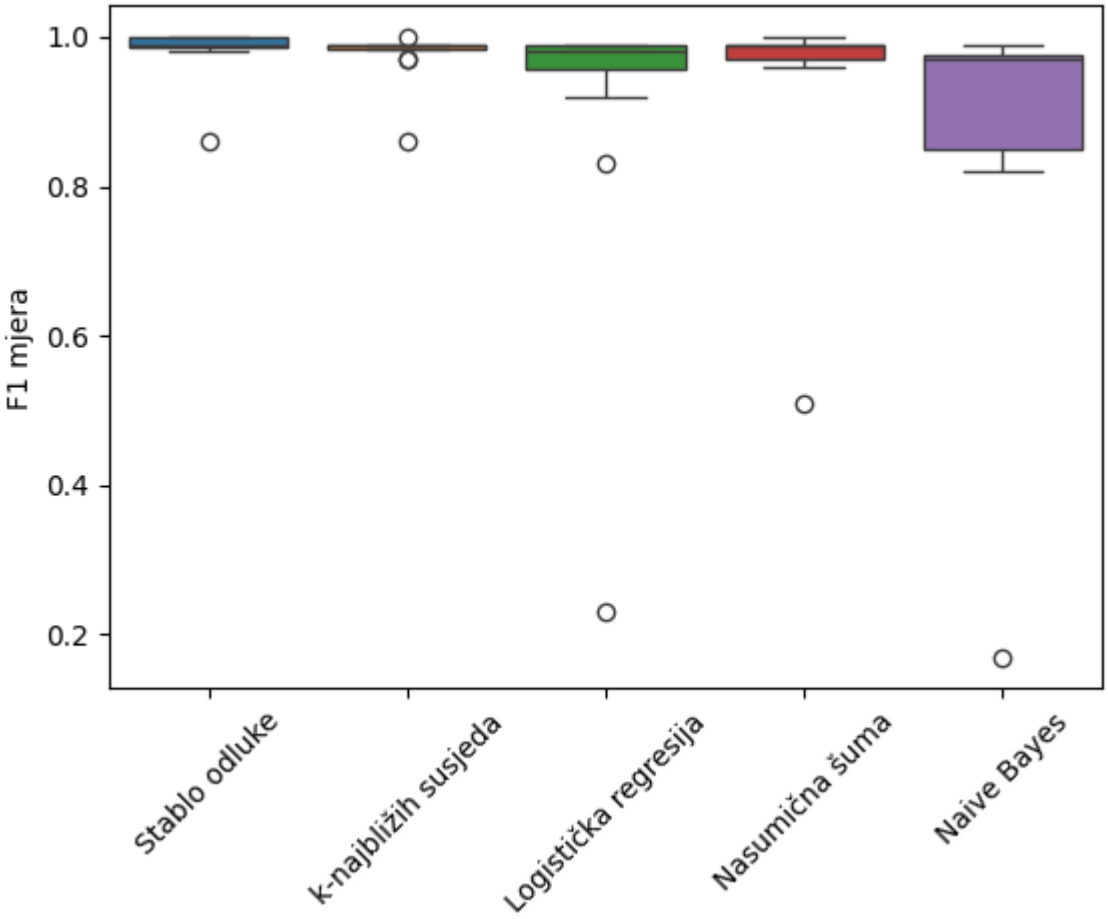
Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**0.86 ± 0.03**	**0.86 ± 0.03**	**0.86 ± 0.03**	0.94 ± 0.01
k-najbližih susjeda	0.86 ± 0.02	0.86 ± 0.02	0.86 ± 0.02	**0.95 ± 0.01**
Logistička regresija	0.31 ± 0.01	0.23 ± 0.01	0.31 ± 0.01	0.77 ± 0.02
Nasumična šuma	0.53 ± 0.02	0.51 ± 0.01	0.53 ± 0.02	0.84 ± 0.01
Naive Bayes	0.29 ± 0.02	0.17 ± 0.02	0.29 ± 0.02	0.79 ± 0.01

Tablica 4.13. Performanse algoritama na UNSW-NB15 skupu podataka

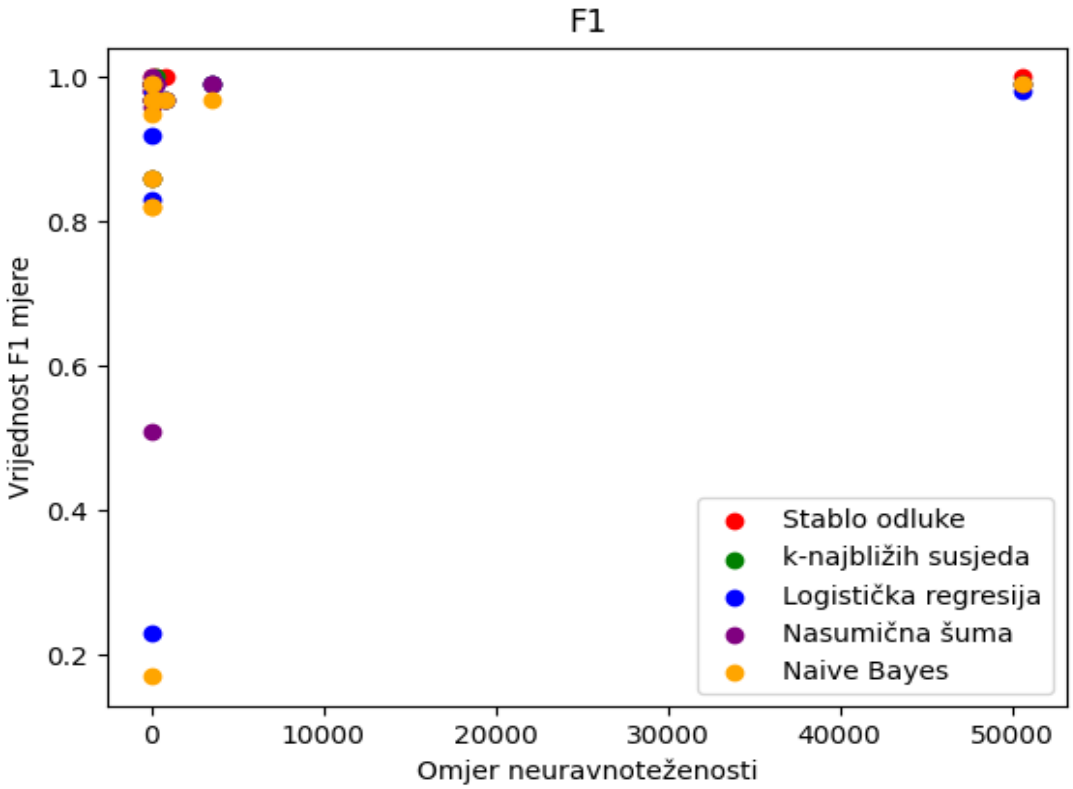
Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**0.98 ± 0.01**	**0.98 ± 0.01**	**0.98 ± 0.01**	**0.98 ± 0.01**
k-najbližih susjeda	0.95 ± 0.01	0.97 ± 0.01	0.95 ± 0.01	0.95 ± 0.01
Logistička regresija	0.90 ± 0.01	0.92 ± 0.01	0.96 ± 0.01	0.79 ± 0.01
Nasumična šuma	0.95 ± 0.01	0.96 ± 0.01	0.97 ± 0.01	0.93 ± 0.01
Naive Bayes	0.82 ± 0.01	0.86 ± 0.01	0.84 ± 0.01	0.80 ± 0.01

Tablica 4.14. Performanse algoritama na CIC-DDOS2019 skupu podataka

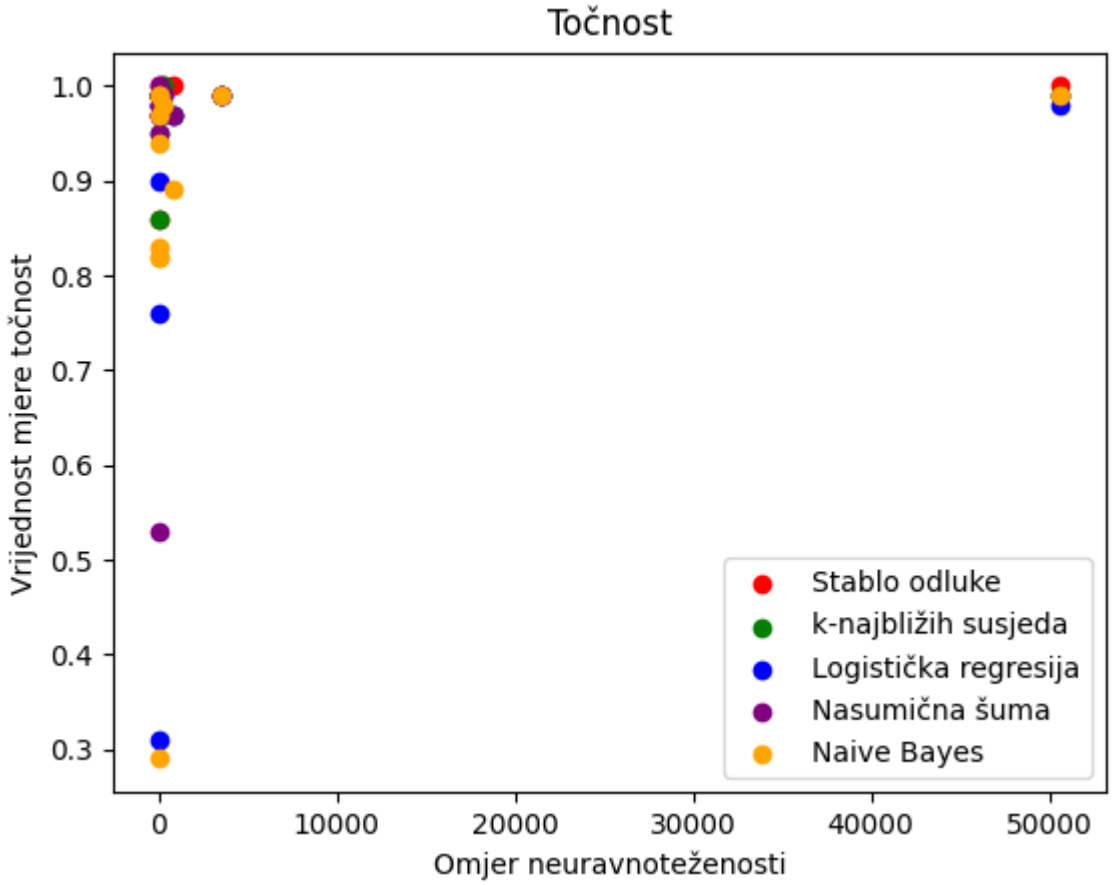
Algoritam	Točnost	F1 mjera	TPR	TNR
Stablo odluke	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**	**1.00 ± 0.00**
k-najbližih susjeda	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01
Logistička regresija	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01
Nasumična šuma	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.98 ± 0.01
Naive Bayes	0.89 ± 0.01	0.87 ± 0.01	0.89 ± 0.01	0.96 ± 0.01



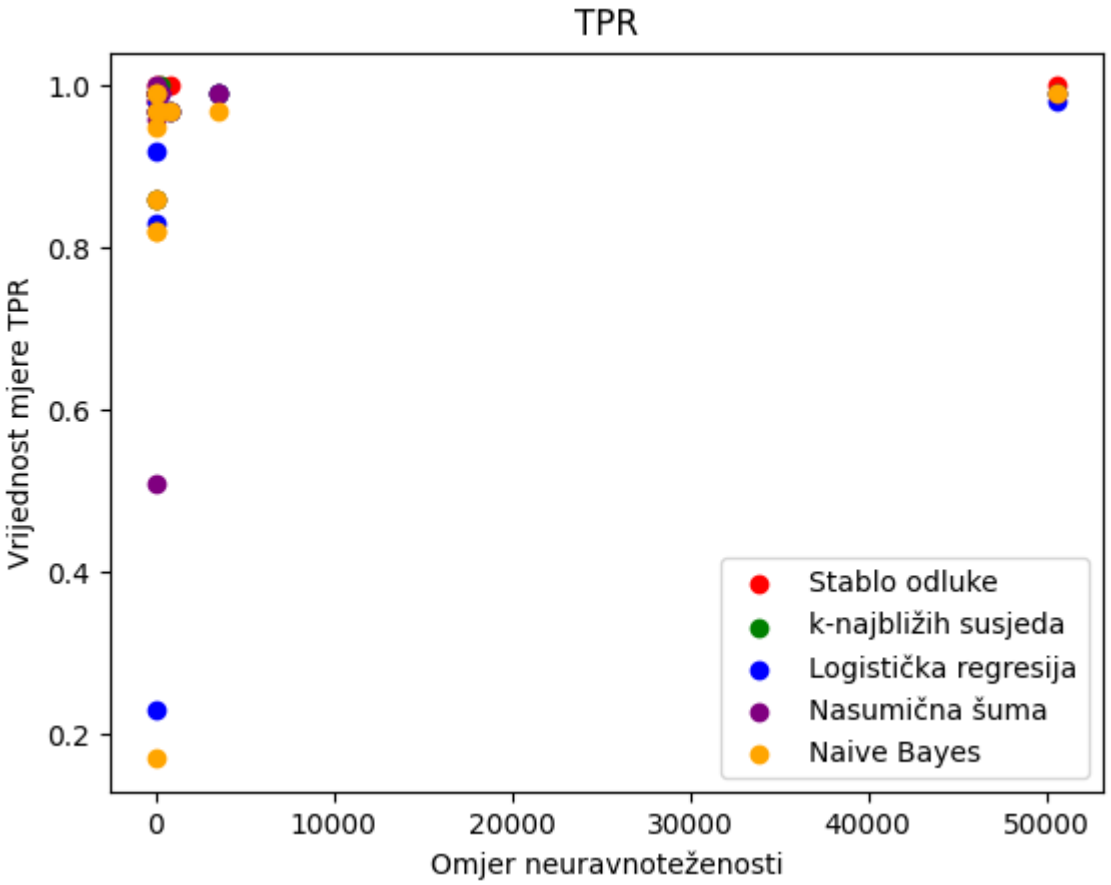
Sl. 4.1. Dijagram pravokutnika vrijednosti F1 mjere za svaki klasifikacijski algoritam



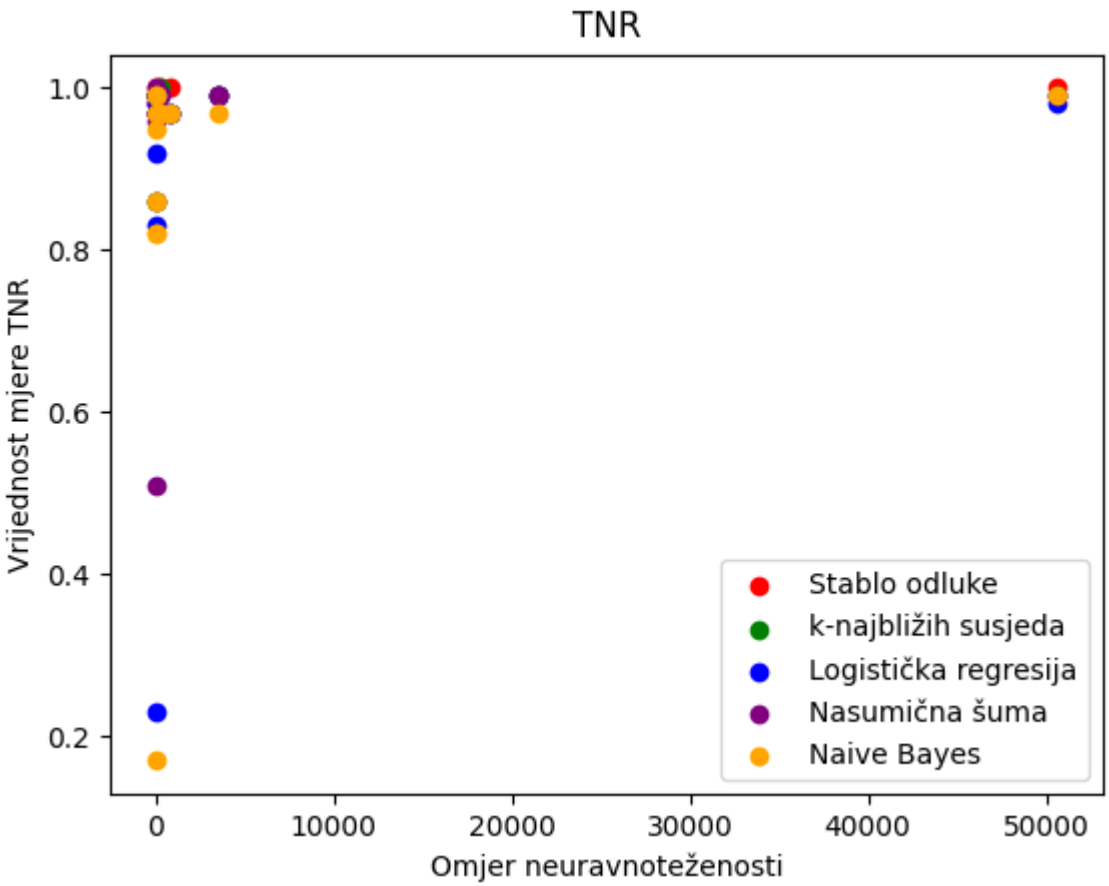
Sl. 4.2. Dijagram raspršenosti vrijednosti mjere F1 i omjera neuravnoteženosti



Sl. 4.3. Dijagram raspršenosti vrijednosti mjere točnost i omjera neuravnoteženosti



Sl. 4.4. Dijagram raspršenosti vrijednosti mjere TPR i omjera neuravnoteženosti



Sl. 4.5. Dijagram raspršenosti vrijednosti mjere TNR i omjera neuravnoteženosti

4.3. Analiza učinka odabira značajki

U ovom poglavlju tehnike odabira značajki primijenjene su na svakom skupu podataka kako bi se utvrdio njihov učinak na performanse algoritma. Ostale tehnike predobrade jednake su kao u prošlom poglavlju.

U tablicama 4.15., 4.16. i 4.17. prikazani su rezultati evaluacije performansi klasifikacijskih algoritama na skupovima podataka uz uporabu filtra za odabir značajki. Sve vrijednosti u tablicama dobivene su kao prosjek F1 mjera za svaku od 10 iteracija na podskupu odabranih značajki. U svakoj iteraciji broj značajki je stalan te predstavlja broj značajki za koji su dobivene najbolje performanse modela. Primjena filtra s Pearsonovom korelacijom je kod većine skupova podataka rezultirala smanjenim ili jednako dobrim performansama prilikom upotrebe manjeg broja značajki za treniranje modela. Smanjenje performansi posebice je vidljivo kod DREBIN skupa podataka koji originalno ima 215 značajki. Znatno smanjenje broja značajki ima negativan utjecaj na smanjenje performansi modela prilikom treniranja na ovom skupu podataka.

Modeli trenirani na ostalim skupovima imaju podjednake performanse kao i prije odabira značajki. Pearsonova korelacija kod nekolicine skupova podatka je bila iznimno niska za sve značajke tako da je bilo potrebno odabrati gotovo podjednak broj značajki kao u originalnom skupu podataka. Na slikama P.4.6., P.4.7., P.4.8., P.4.9. i P.4.10. iz priloga vidljivo je kako se performanse modela poboljšavaju s povećanjem broja značajki. Također se može zaključiti da što je veći broj značajki to model postaje konstantniji u prepoznavanju malicioznih napada.

Performanse dobivene upotrebom filtra sa zajedničkom informacijom slične su onima bez odabira značajki, ali je u ovom slučaju njihov broj smanjen. Na skupu CICIDS2017 prilikom korištenja logističke regresije performanse su se znatno razlikovale smanjenjem broja značajki. *k*-najbližih susjeda jedan je od algoritama koji je kod manjeg broja značajki ostvario bolje ili podjednako dobre performanse kao i s maksimalnim brojem značajki, što je vidljivo na slici P.4.17.

Filter s ANOVA F vrijednosti, isto kao i prethodna dva, u većini slučajeva je pokazao manje ili podjednake performanse smanjenjem broja značajki. Kod nekolicine skupova podataka došlo je do boljih rezultata kada sam smanjio broj značajki. Također je bilo primjetno da kod ovoga algoritma izbor broja značajki ima najveći utjecaj. Optimalni broj značajki je dao najbolje rezultate, a dodavanjem ili uklanjanjem samo jedne značajke performanse su naglo opale. Sa slika iz priloga vidljivo je da prilikom primjene filtra s ANOVA F vrijednosti performanse rastu uz porast broja značajki. Modeli primjenom ove tehnike ostvaruju lošije performanse za manji broj značajki nego modeli na kojima je primijenjen neki drugi filtar. To je posebice vidljivo na slikama P.4.13. i P.4.14. Za sva tri filtra zajedničko je povećanje performansi povećanjem broja značajki.

SFS omotač je zadnja korištena tehnika za odabir značajki te se rezultati njegove primjene mogu vidjeti u tablici 4.18. Za razliku od sva tri filtera, primjena SFS omotača je kod gotovo svakog skupa pridonijela poboljšanju klasifikacije ili su performanse ostale jednake uz smanjen broj značajki. Primjena ovog omotača je poboljšala performanse algoritama kod većine skupa podataka. Kao što je vidljivo na slikama 4.6., 4.7., 4.8., 4.9. i 4.10. povećanjem broj značajki poboljšavaju se performanse svakog algoritma, a kada je broj performansi oko 19 tada svi algoritmi postižu bolje rezultate klasifikacije nego na skupovima podataka sa svim značajkama.

Tablica 4.15. Rezultati F1 mjere nakon primjene filtera s Pearsonovom korelacijom

Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.82 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.90 ± 0.01 (-0.05)
KDD99	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
NSL-KDD	**0.95 ± 0.01** (-0.04)	0.94 ± 0.01 (-0.05)	0.80 ± 0.01 (-0.17)	0.91 ± 0.01 (-0.06)	0.72 ± 0.02 (-0.11)
KYOTO	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	0.98 ± **0.01** (+0.01)
DREBIN	0.93 ± 0.01 (-0.05)	0.93 ± 0.01 (-0.06)	0.90 ± 0.01 (-0.08)	**0.94 ± 0.01** (-0.03)	0.77 ± 0.01 (-0.05)
ISCXIDS2012	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)
CICIDS2017	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (-0.01)
DS2OS	**0.99 ± 0.00** (-0.01)	**0.99 ± 0.00** (-0.01)	0.97 ± 0.01 (-0.02)	**0.99 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
IMPACT	**0.87 ± 0.01** (+0.01)	0.86 ± 0.01 (-0.00)	0.19 ± 0.01 (-0.04)	0.52 ± 0.01 (+0.01)	0.19 ± 0.01 (+0.02)
UNSW-NB15	**1.00 ± 0.00** (+0.02)	**1.00 ± 0.00** (+0.03)	0.92 ± 0.00 (-0.00)	**1.00 ± 0.00** (+0.04)	0.99 ± 0.01 (+0.13)
CIC-DDOS2019	**1.00 ± 0.00** (-0.00)	0.99 ± 0.01 (+0.02)	0.82 ± 0.01 (-0.15)	0.99 ± 0.01 (+0.02)	0.89 ± 0.01 (+0.02)

Tablica 4.16. Rezultati F1 mjere nakon primjene filtera s zajedničkom informacijom

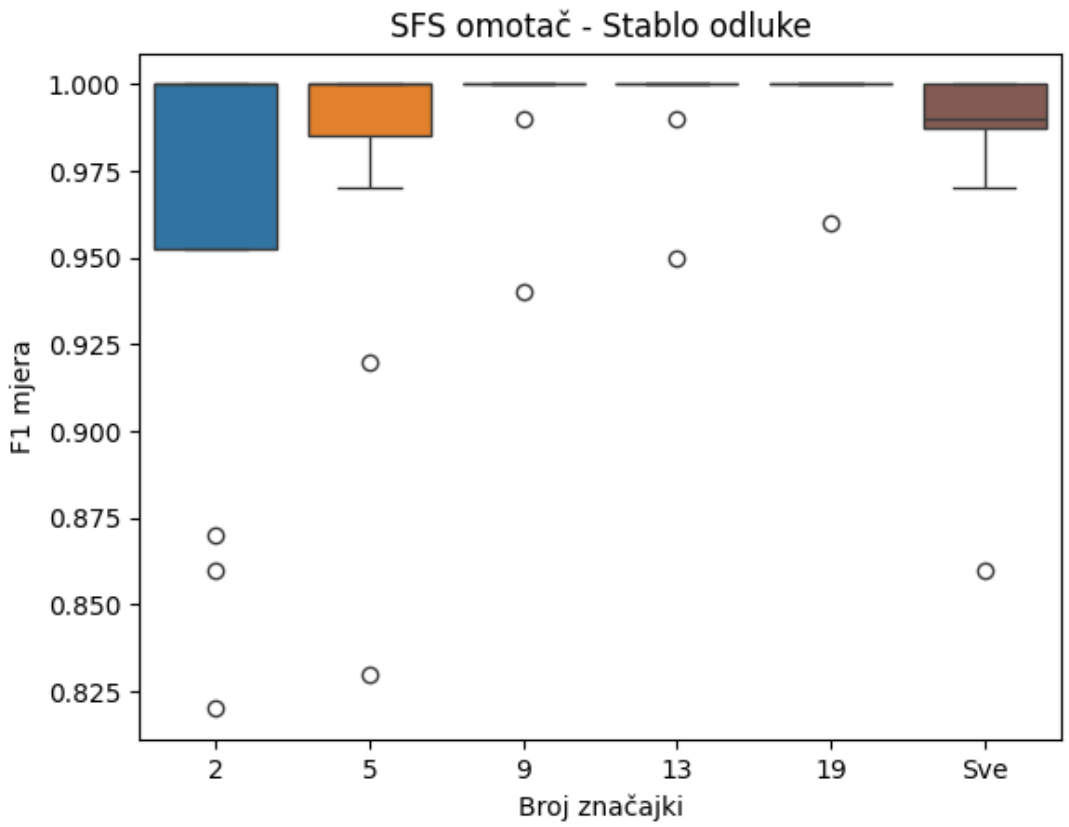
Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.82 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.90 ± 0.01 (-0.05)
KDD99	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (+0.01)
NSL-KDD	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.96 ± 0.01 (-0.01)	0.98 ± 0.01 (+0.01)	0.85 ± 0.01 (+0.03)
KYOTO	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
DREBIN	0.97 ± 0.01 (-0.01)	**0.98 ± 0.01** (-0.01)	0.96 ± 0.01 (-0.02)	0.96 ± 0.01 (-0.01)	0.80 ± 0.01 (-0.02)
ISCXIDS2012	**1.00 ± 0.00** (+0.01)	0.99 ± 0.00 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.00 (-0.00)	0.99 ± 0.00 (-0.00)
CICIDS2017	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (-0.01)
DS2OS	0.98 ± 0.01 (-0.02)	**0.99 ± 0.01** (-0.01)	0.98 ± 0.01 (-0.02)	**0.99 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
IMPACT	**0.87 ± 0.01** (+0.01)	0.86 ± 0.01 (-0.00)	0.24 ± 0.01 (+0.01)	0.51 ± 0.01 (-0.00)	0.17 ± 0.01 (-0.00)
UNSW-NB15	**1.00 ± 0.00** (+0.02)	0.99 ± 0.00 (+0.02)	0.92 ± 0.00 (-0.00)	0.98 ± 0.00 (+0.02)	**1.00 ± 0.00** (+0.14)
CIC-DDOS2019	**1.00 ± 0.00** (-0.00)	0.98 ± 0.01 (+0.01)	0.87 ± 0.01 (-0.11)	0.99 ± 0.01 (+0.02)	0.88 ± 0.01 (+0.01)

Tablica 4.17. Rezultati F1 mjere nakon primjene filtra s ANOVA F vrijednosti

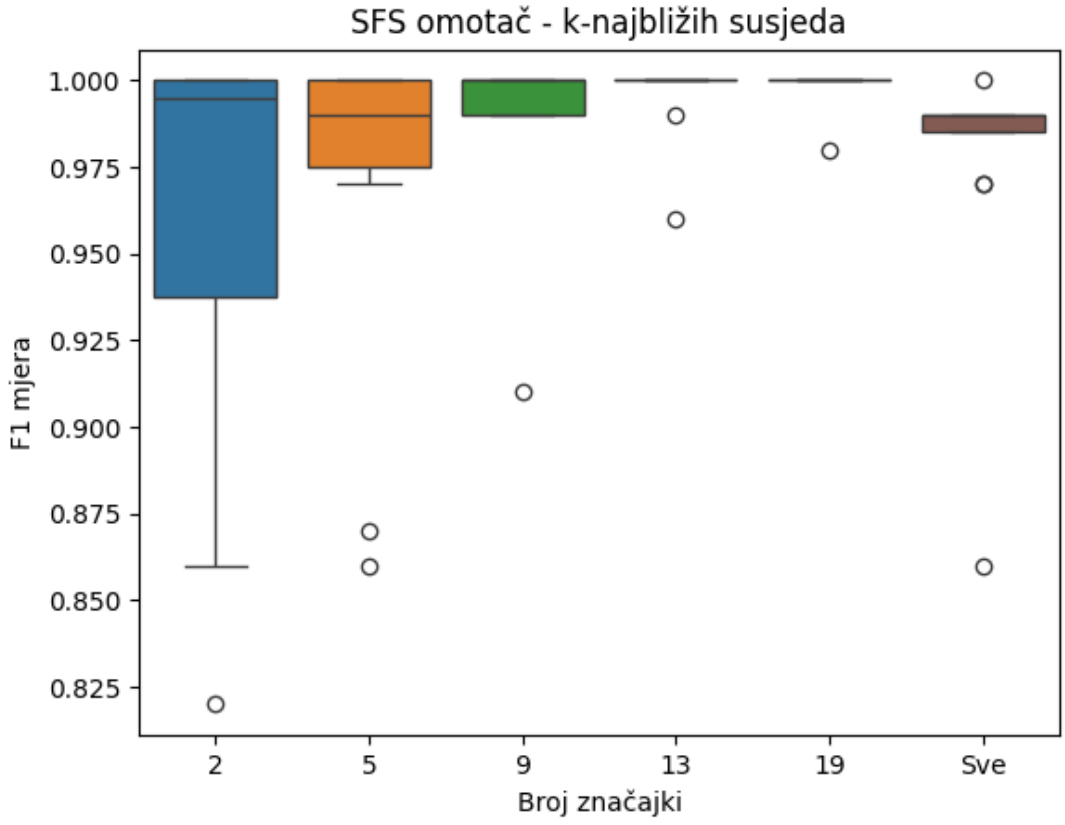
Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.82 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.91 ± 0.01 (-0.04)
KDD99	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.98 ± 0.01 (-0.01)	**0.99 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
NSL-KDD	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.98 ± 0.01 (+0.01)	0.98 ± 0.01 (+0.01)	0.83 ± 0.01 (+0.01)
KYOTO	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	0.98 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	0.97 ± 0.01 (-0.00)
DREBIN	0.97 ± 0.01 (-0.01)	**0.98 ± 0.01** (-0.01)	0.97 ± 0.01 (-0.01)	0.96 ± 0.01 (-0.01)	0.80 ± 0.01 (-0.02)
ISCXIDS2012	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)
CICIDS2017	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.02)	**1.00 ± 0.00** (+0.01)	0.98 ± 0.01 (-0.01)
DS2OS	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (-0.00)	0.99 ± 0.01 (-0.01)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.00)
IMPACT	0.86 ± 0.01 (-0.00)	**0.87 ± 0.01** (+0.01)	0.23 ± 0.01 (-0.00)	0.52 ± 0.01 (+0.01)	0.17 ± 0.01 (-0.00)
UNSW-NB15	**1.00 ± 0.00** (+0.02)	**1.00 ± 0.00** (+0.03)	0.95 ± 0.01 (+0.03)	**1.00 ± 0.00** (+0.04)	0.97 ± 0.01 (+0.11)
CIC-DDOS2019	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.01** (+0.03)	0.97 ± 0.01 (-0.00)	0.99 ± 0.01 (+0.02)	0.88 ± 0.01 (+0.01)

Tablica 4.18. Rezultati F1 mjere nakon primjene SFS omotača

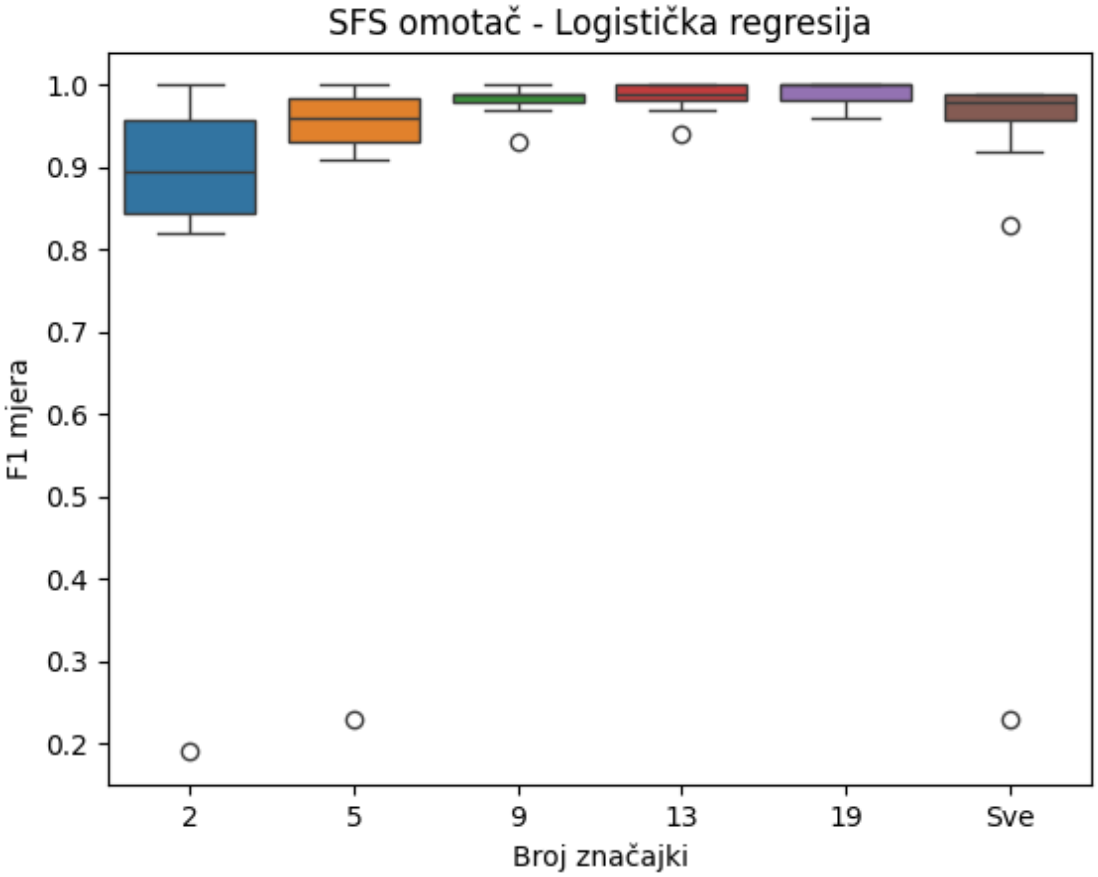
Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**0.99 ± 0.00** (-0.00)	**0.99 ± 0.00** (-0.00)	0.83 ± 0.01 (-0.00)	**0.99 ± 0.00** (-0.00)	0.91 ± 0.01 (-0.04)
KDD99	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (+0.01)	0.99 ± 0.01 (-0.00)	0.98 ± 0.01 (+0.01)
NSL-KDD	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (+0.02)	0.99 ± 0.01 (-0.00)	0.88 ± 0.01 (+0.06)
KYOTO	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.03)
DREBIN	**0.99 ± 0.01** (+0.01)	0.98 ± 0.01 (-0.01)	0.98 ± 0.01 (-0.00)	0.98 ± 0.01 (+0.01)	0.95 ± 0.01 (+0.013)
ISCXIDS2012	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)
CICIDS2017	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
DS2OS	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.00)	0.98 ± 0.01 (-0.02)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.00)
IMPACT	0.87 ± 0.01 (+0.01)	**0.88 ± 0.01** (+0.01)	0.22 ± 0.01 (-0.01)	0.54 ± 0.01 (+0.03)	0.16 ± 0.01 (-0.01)
UNSW-NB15	**1.00 ± 0.00** (+0.02)	**1.00 ± 0.00** (+0.03)	0.95 ± 0.01 (+0.03)	**1.00 ± 0.00** (+0.04)	**1.00 ± 0.00** (+0.14)
CIC-DDOS2019	**1.00 ± 0.00** (-0.00)	0.99 ± 0.01 (+0.02)	0.98 ± 0.01 (+0.01)	0.98 ± 0.01 (+0.01)	0.96 ± 0.01 (+0.09)



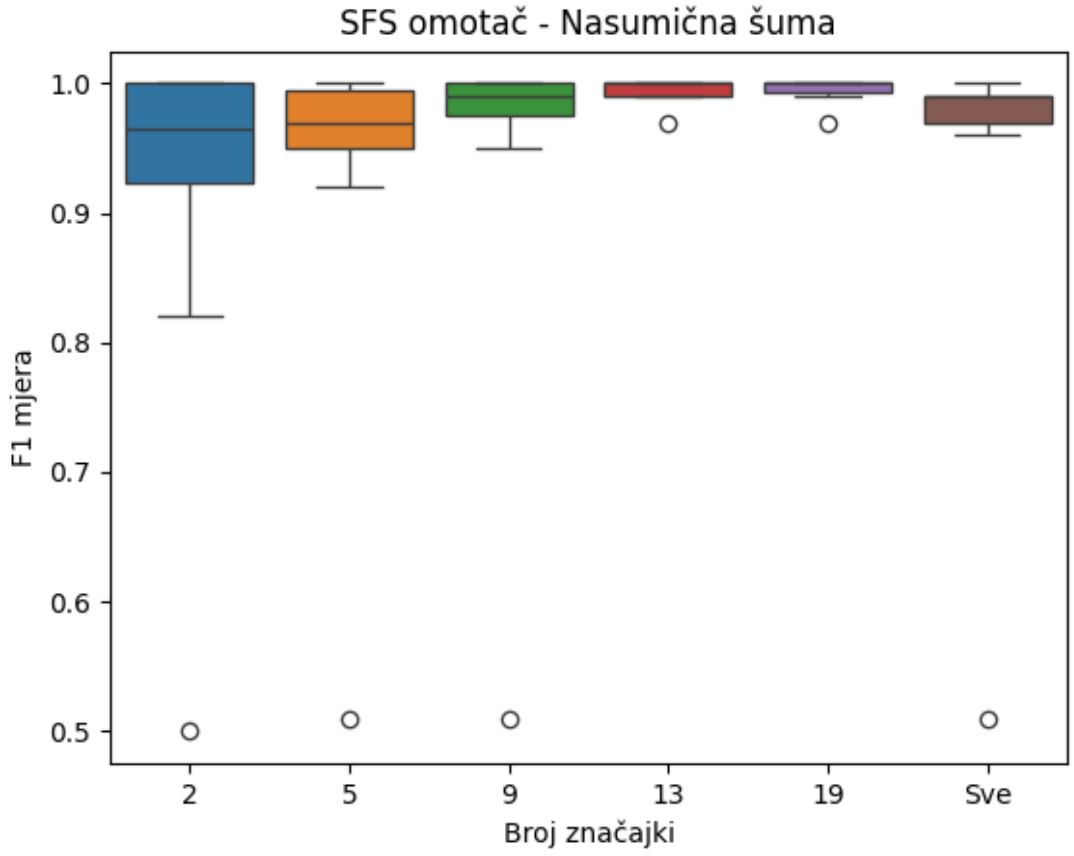
Sl. 4.6. Dijagram pravokutnika vrijednosti F1 mjere za algoritam stablo odluke pri različitom broju značajki odabranih SFS omotačem



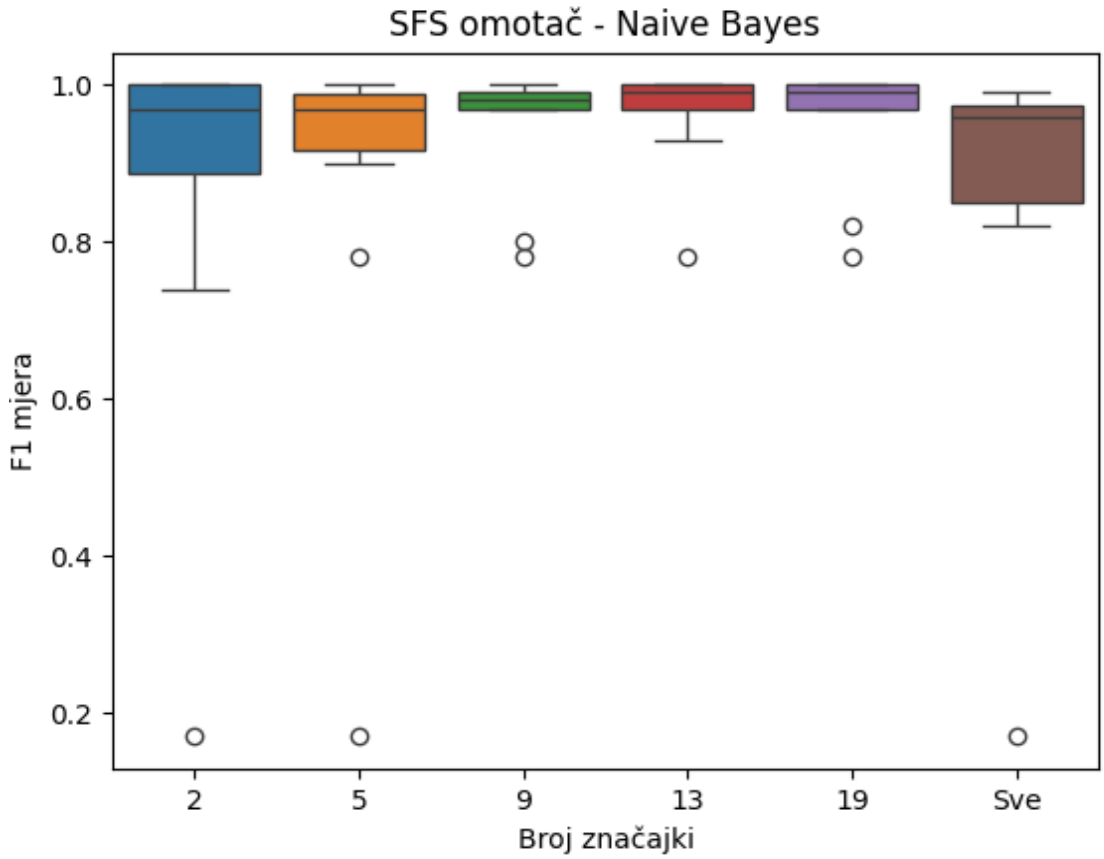
Sl. 4.7. Dijagram pravokutnika vrijednosti F1 mjere za algoritam *k*-najbližih pri različitom broju značajki odabranih SFS omotačem



Sl. 4.8. Dijagram pravokutnika vrijednosti F1 mjere za algoritam logističku regresiju pri različitom broju značajki odabranih SFS omotačem



Sl. 4.9. Dijagram pravokutnika vrijednosti F1 mjere za algoritam nasumična šuma pri različitom broju značajki odabranih SFS omotačem



Sl. 4.10. Dijagram pravokutnika vrijednosti F1 mjere za algoritam Naive Bayes pri različitom broju značajki odabranih SFS omotačem

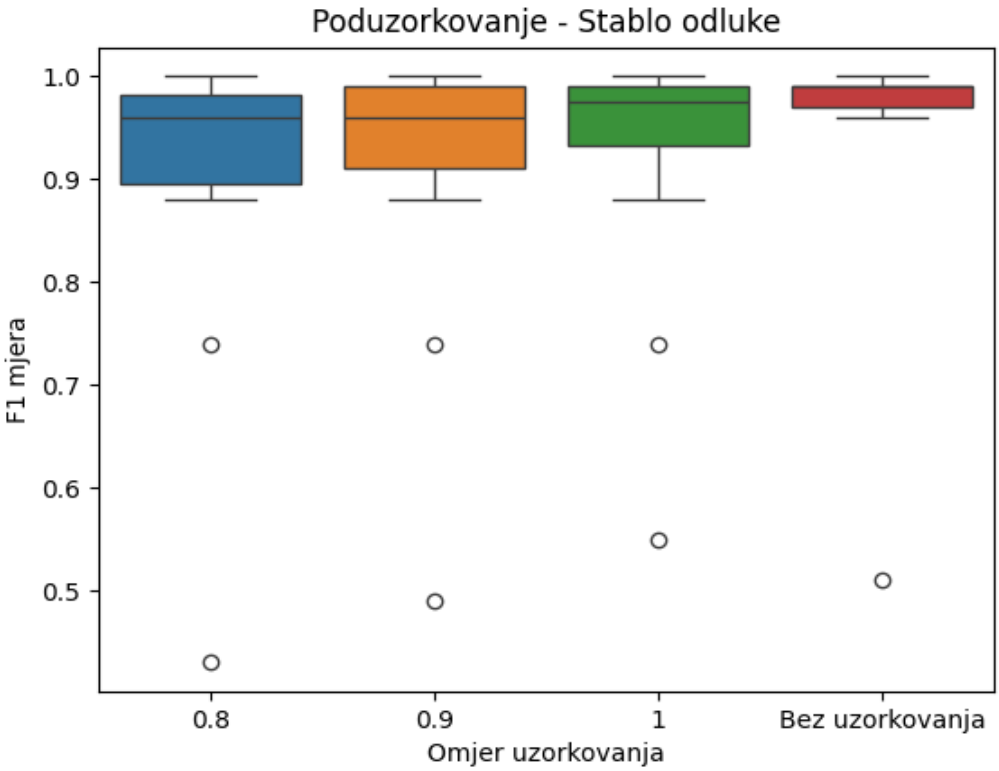
4.4. Analiza učinka uzorkovanja

U ovom poglavlju prikazan je učinak tehnike uzorkovanja na skupovima podataka za smanjenje stupnja neuravnoteženosti skupa podataka. Tehnike koje su korištene su nasumično preuzorkovanje i nasumično poduzorkovanje.

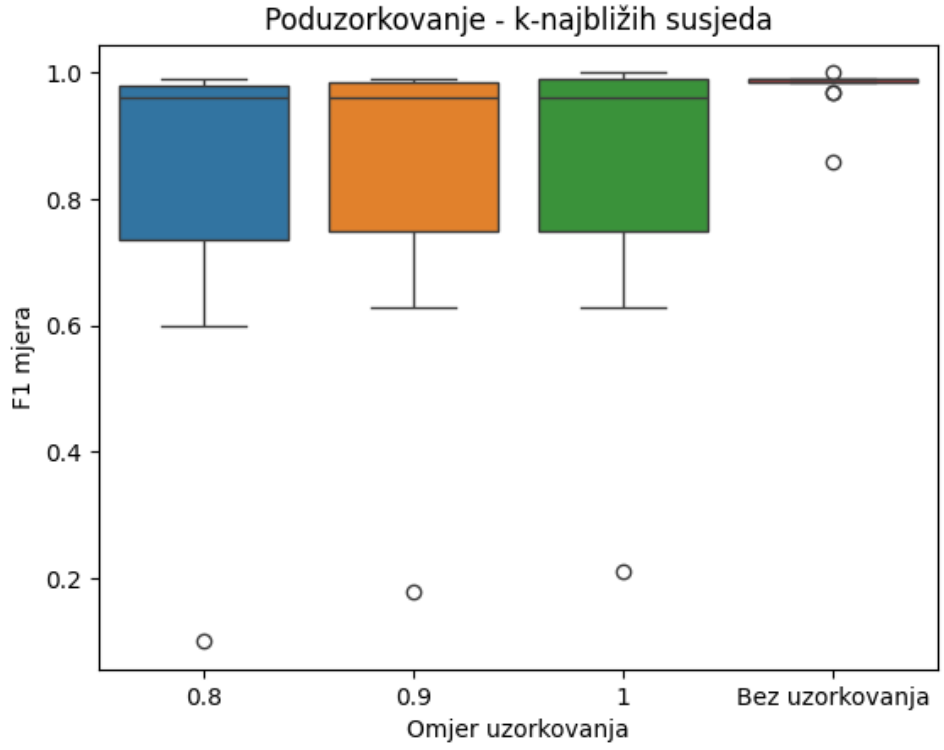
Prema rezultatima vidljivim u tablici 4.19. postupak poduzorkovanja nema preveliki doprinos poboljšanju performansi modela. Kod većine algoritama može se primijetiti veliki pad u performansama. Razlog tomu je veliki broj klasa koje imaju vrlo mali broj primjeraka te se prilikom poduzorkovanja gubi velika količina podataka. Ovaj slučaj je primjetan kod kod NSL-KDD skupa podataka koji ima 40 klasa. Nasuprot tome kod skupova podataka DARPA i CICIDS2017 performanse su malo poboljšane ili podjednako dobre izjednačavanjem broja klasa. Kao najbolji klasifikacijski algoritam nakon poduzorkovanja pokazao se algoritam stablo odluke koji ostvaruje zadovoljavajuće performanse. Ovaj algoritam postaje sve bolji što je broj podatkovnih primjera u klasama ujednačeniji kao što se je prikazano na slici 4.11.

Tablica 4.19. Rezultati F1 mjere nakon poduzorkovanja skupova podataka

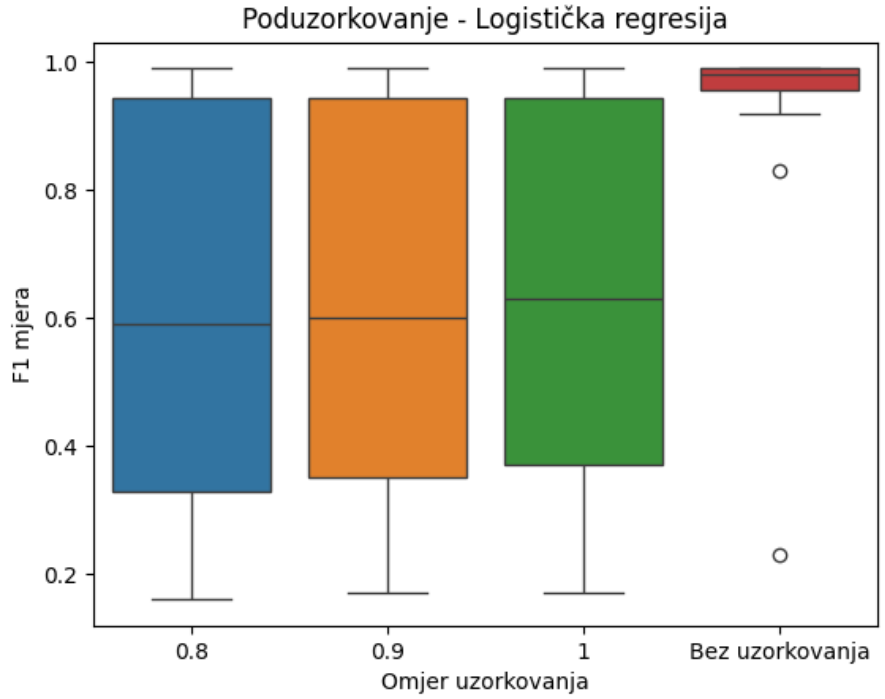
Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.84 ± 0.01 (+0.01)	**1.00 ± 0.00** (+0.01)	0.93 ± 0.03 (-0.02)
KDD99	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.32 ± 0.04 (-0.067)	0.95 ± 0.00 (-0.05)	0.30 ± 0.02 (-0.68)
NSL-KDD	0.58 ± 0.03 (-0.42)	**0.63 ± 0.02** (-0.37)	0.45 ± 0.03 (-0.54)	0.41 ± 0.01 (-0.58)	0.34 ± 0.01 (-0.54)
KYOTO	**0.74 ± 0.02** (-0.28)	0.22 ± 0.01 (-0.77)	0.25 ± 0.02 (-0.73)	0.24 ± 0.03 (-0.78)	0.66 ± 0.01 (-0.33)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.03)
DREBIN	0.97 ± 0.01 (-0.01)	**0.98 ± 0.01** (-0.01)	0.96 ± 0.01 (-0.02)	0.96 ± 0.01 (-0.01)	0.79 ± 0.03 (-0.03)
ISCXIDS2012	**0.98 ± 0.01** (-0.01)	0.97 ± 0.02 (-0.02)	0.95 ± 0.01 (-0.04)	1.00 ± 0.00 (+0.01)	**0.98 ± 0.01** (-0.01)
CICIDS2017	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
DS2OS	0.95 ± 0.02 (-0.05)	0.96 ± 0.01 (-0.04)	0.42 ± 0.01 (-0.48)	**1.00 ± 0.00** (+0.01)	0.01 ± 0.00 (-0.96)
IMPACT	**0.88 ± 0.01** (+0.02)	0.87 ± 0.02 (+0.01)	0.17 ± 0.01 (-0.04)	0.39 ± 0.01 (-0.11)	0.13 ± 0.02 (-0.04)
UNSW-NB15	**0.98 ± 0.01** (-0.00)	0.96 ± 0.01 (-0.01)	0.93 ± 0.01 (+0.01)	0.96 ± 0.01 (-0.00)	0.86 ± 0.01 (-0.00)
CIC-DDOS2019	**0.99 ± 0.01** (-0.01)	0.63 ± 0.02 (-0.34)	0.65 ± 0.01 (-0.33)	0.66 ± 0.01 (-0.31)	0.86 ± 0.03 (-0.01)



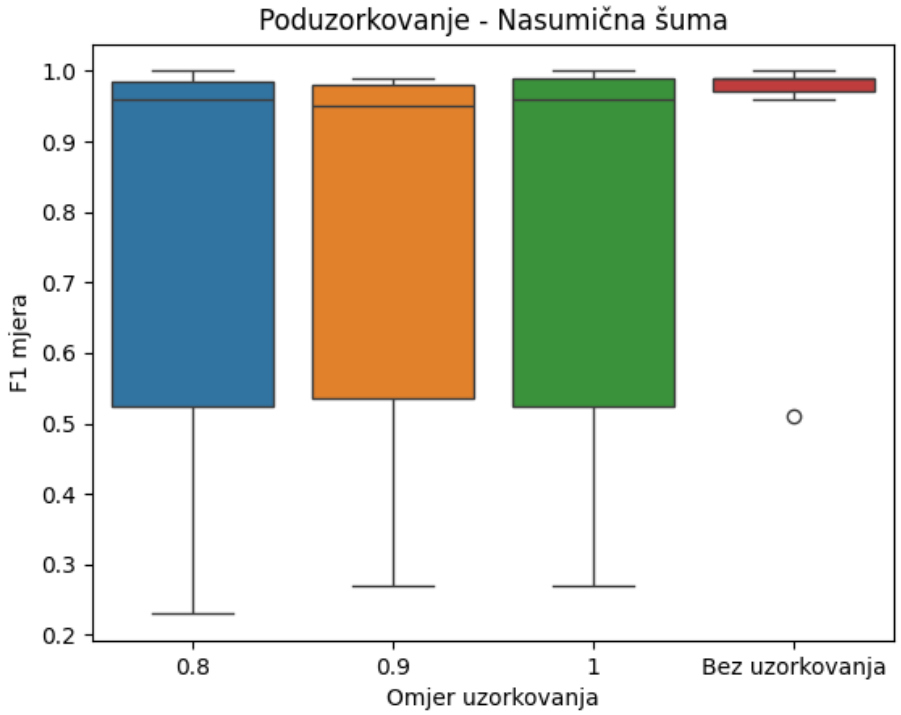
Sl. 4.11. Dijagram pravokutnika vrijednosti F1 mjere za algoritam stablo odluke pri različitom omjeru uzorkovanja



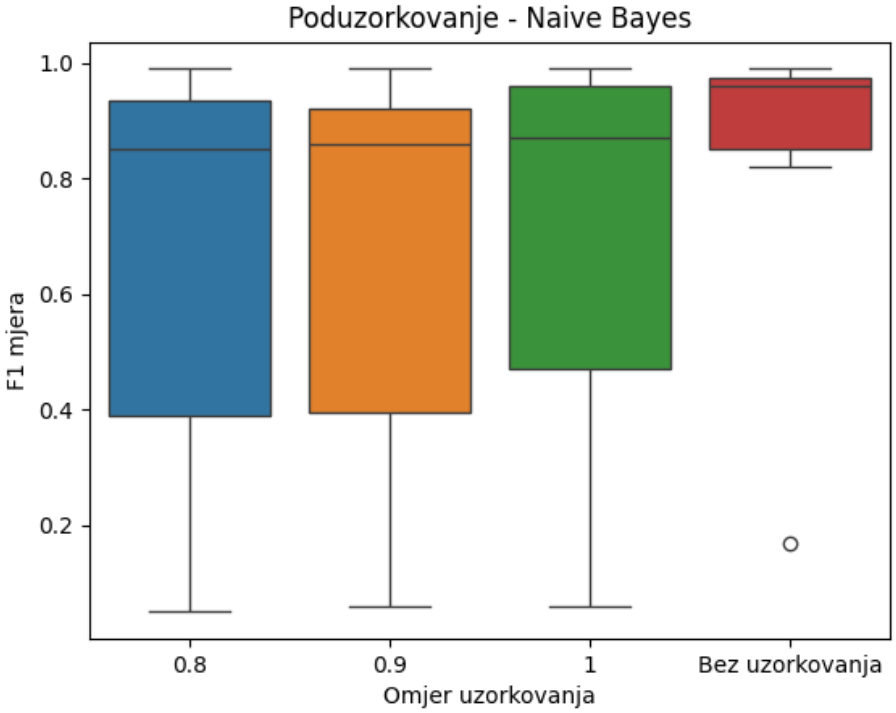
Sl. 4.12. Dijagram pravokutnika vrijednosti F1 mjere za algoritam *k*-najbližih susjeda pri različitom omjeru uzorkovanja



Sl. 4.13. Dijagram pravokutnika vrijednosti F1 mjere za algoritam logistička regresija pri različitom omjeru uzorkovanja



Sl. 4.14. Dijagram pravokutnika vrijednosti F1 mjere za algoritam nasumična šuma pri različitom omjeru uzorkovanja

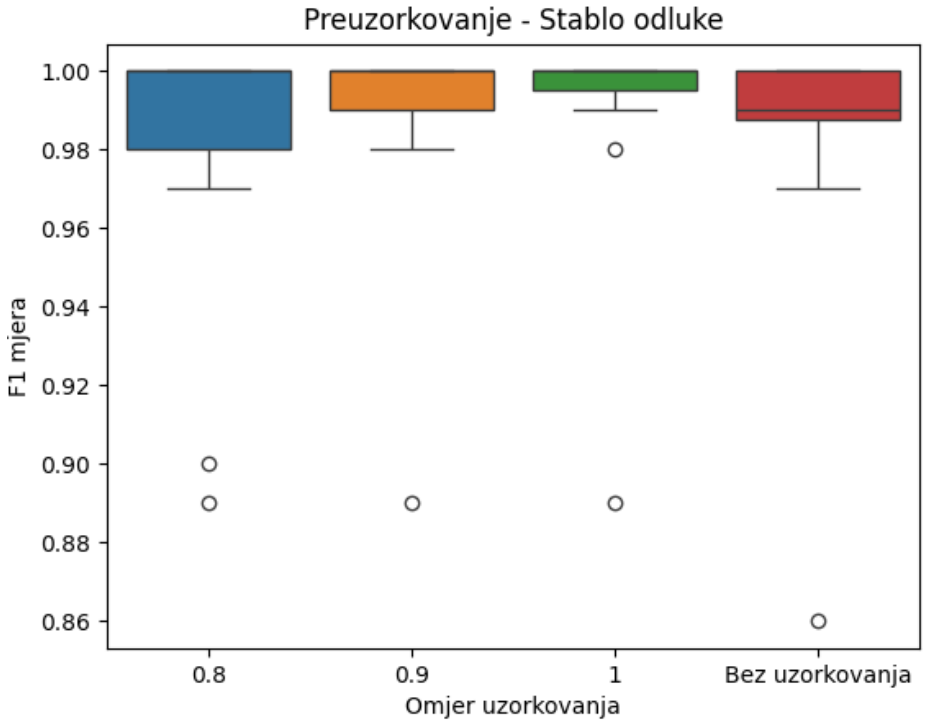


Sl. 4.15. Dijagram pravokutnika vrijednosti F1 mjere za algoritam Naive Bayes pri različitom omjeru uzorkovanja

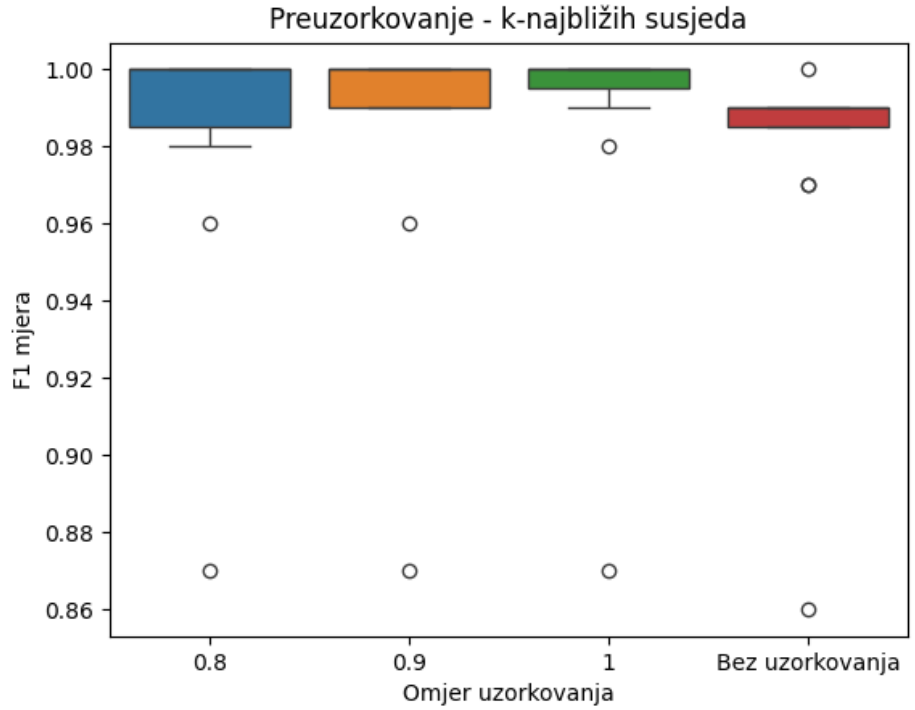
Preuzorkovanje je rezultiralo puno boljim performansama algoritama na svakom skupu podataka za razliku od poduzorkovanja. Svi algoritmi nakon primjene ove tehnike imaju visoke performanse te se kod većine vidi poboljšanje. Na slici 4.14. vidljivo je da algoritam stablo odluke ima poboljšanje performanse kada su klase izjednačene te kod svih skupova podataka ostvaruje gotovo savršene rezultate klasifikacije. Dva algoritma koja ostvaruju gotovo podjednake rezultate su *k*-najbližih susjeda i nasumična šuma. Posebice dobre performanse ima algoritam *k*-najbližih susjeda koji kod svake vrijednosti omjera uzorkovanja ostvaruje bolje rezultate na uzorkovanim skupovima podataka nego na originalnim što se može vidjeti na slici 4.15. Algoritam logistička regresija nešto je lošiji od prethodna tri jer rezultati više variraju i nisu toliko konstantni. Najlošiji od svih algoritama je Naive Bayes algoritam zbog velikih varijacija u rezultatima i najnižih performansi na gotovo svakom skupu podataka.

Tablica 4.20. Rezultati F1 mjere nakon preuzorkovanja skupova podataka

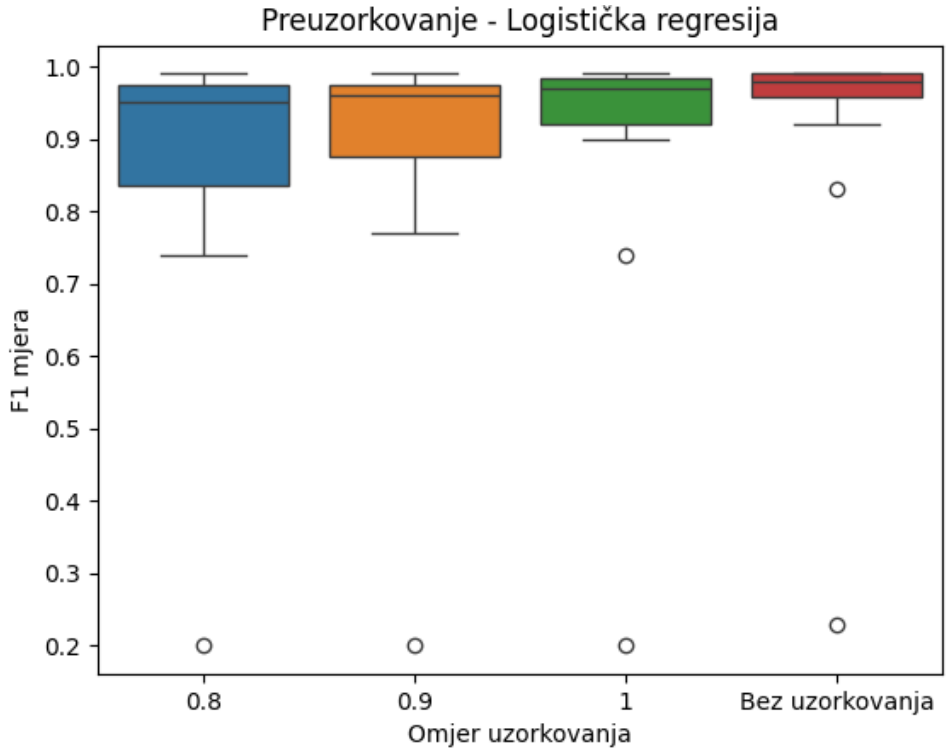
Skupovi podataka	**Stablo odluke**	***k*-najbližih susjeda**	**Logistička regresija**	**Nasumična šuma**	**Naive Bayes**
DARPA	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.83 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.95 ± 0.01 (-0.00)
KDD99	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (+0.01)	**1.00 ± 0.00** (+0.01)	0.81 ± 0.01 (-0.17)
NSL-KDD	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (+0.02)	0.99 ± 0.01 (-0.00)	0.51 ± 0.01 (-0.37)
KYOTO	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	0.97 ± 0.01 (-0.01)	0.99 ± 0.01 (-0.00)	0.99 ± 0.01 (-0.00)
Malware	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.03)
DREBIN	0.97 ± 0.01 (-0.01)	**0.98 ± 0.01** (-0.01)	0.97 ± 0.01 (-0.01)	0.96 ± 0.01 (-0.01)	0.79 ± 0.03 (-0.03)
ISCXIDS2012	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.98 ± 0.01 (-0.01)
CICIDS2017	**1.00 ± 0.00** (+0.01)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)	**1.00 ± 0.00** (+0.01)	0.99 ± 0.01 (-0.00)
DS2OS	**1.00 ± 0.00** (-0.00)	**1.00 ± 0.00** (+0.00)	0.99 ± 0.01 (-0.01)	**1.00 ± 0.00** (+0.01)	0.87 ± 0.02 (-0.10)
IMPACT	**0.87 ± 0.01** (+0.01)	0.86 ± 0.01 (-0.01)	0.18 ± 0.01 (-0.03)	0.50 ± 0.01 (-0.01)	0.12 ± 0.01 (-0.05)
UNSW-NB15	**1.00 ± 0.00** (+0.02)	**1.00 ± 0.00** (+0.03)	0.95 ± 0.01 (+0.03)	**1.00 ± 0.00** (+0.04)	0.98 ± 0.01 (+0.12)
CIC-DDOS2019	**1.00 ± 0.00** (-0.00)	0.96 ± 0.01 (-0.01)	0.98 ± 0.01 (+0.01)	0.97 ± 0.01 (-0.00)	0.87 ± 0.01 (-0.00)



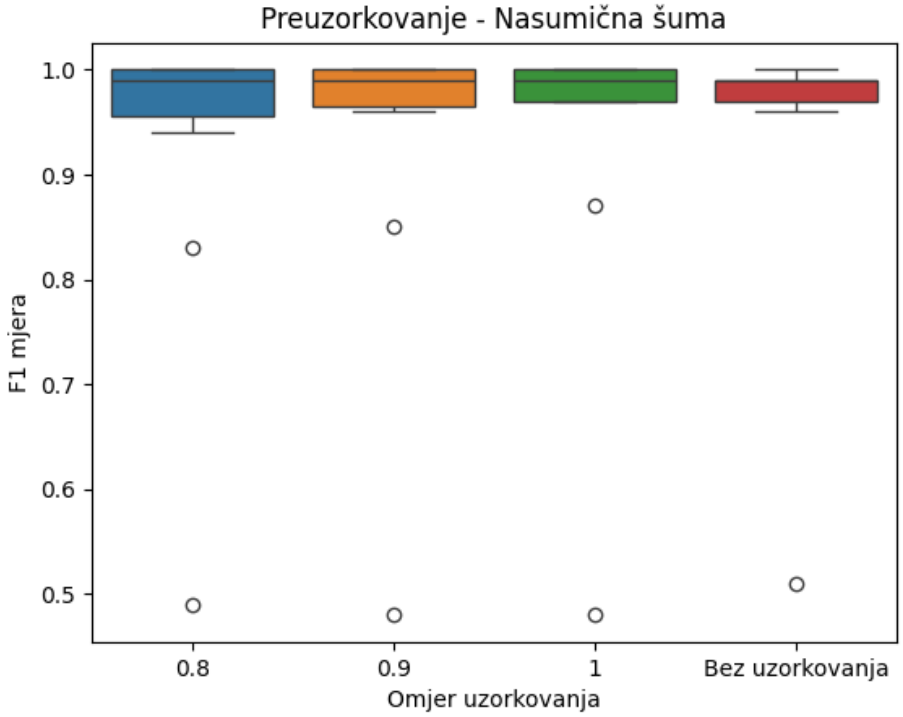
Sl. 4.14. Dijagram pravokutnika vrijednosti F1 mjere za algoritam stablo odluke pri različitom omjeru uzorkovanja



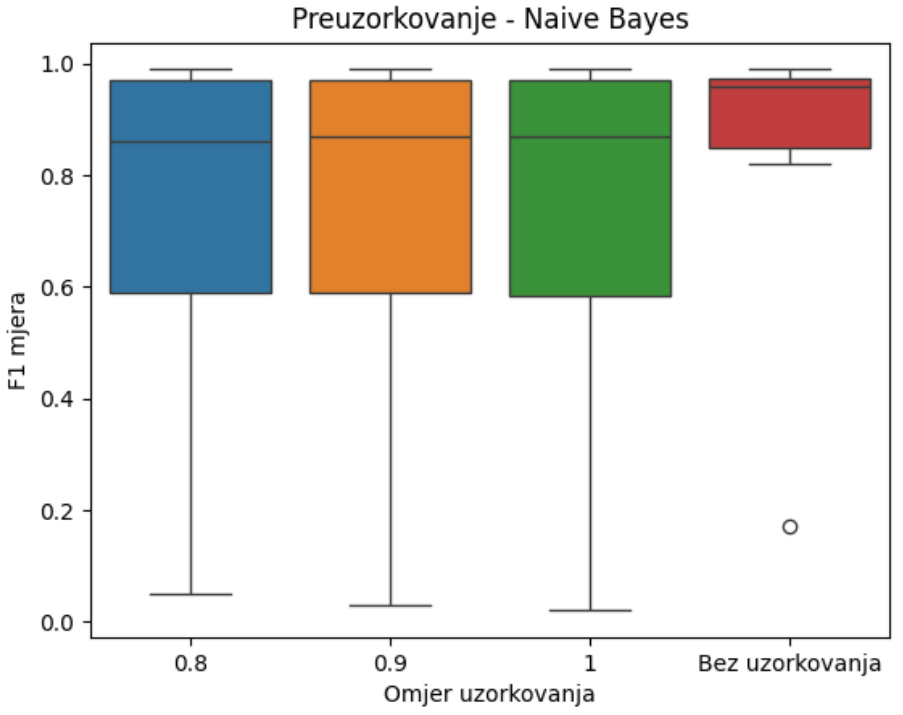
Sl. 4.15. Dijagram pravokutnika vrijednosti F1 mjere za algoritam *k*-najbližih susjeda pri različitom omjeru uzorkovanja



Sl. 4.16. Dijagram pravokutnika vrijednosti F1 mjere za algoritam logistička regresija pri različitom omjeru uzorkovanja



Sl. 4.17. Dijagram pravokutnika vrijednosti F1 mjere za algoritam nasumična šuma pri različitom omjeru uzorkovanja



Sl. 4.18. Dijagram pravokutnika vrijednosti F1 mjere za algoritam Naive Bayes pri različitom omjeru uzorkovanja

5. ZAKLJUČAK

Razvojem tehnologije omogućen je napredak u svim poljima ljudskog života te nema dana kada se ne koristimo nekim oblikom računalnog uređaja ili Internetom. Kako se nove tehnologije koriste za napredak isto tako postoje i oni koji ih zloupotrebljavaju. Broj različitih prijetnji sve je veći i one postaju sve naprednije, a tradicionalni oblici zaštite u obliku sustava za otkrivanje upada i programi za zaštitu od virusa sve teže mogu pratiti njihov razvoj.

Za tu svrhu sve se više počinje koristiti strojno učenje u području kibernetičke sigurnosti kako bi se unaprijedila obrana. Tehnika klasifikacije nadziranog strojnog učenja pogotovo je korisna jer omogućuje treniranje modela koji će na temelju podataka o postojećim napadima moći otkriti nove vrste napada s kojima se još nisu susreli.

Sam proces kreiranja modela sastoji se od nekoliko koraka. Prvo je važno tehnikama predobrade osigurati kvalitetu podataka koji čine temelj strojnog učenja. Odabirom algoritma prelazi se na idući korak treniranja modela te nakon što je model istreniran upotrebom mjera kao što su točnost, F1 mjera, TPR i TNR vrednuju se njegove performanse.

Svi korišteni algoritmi ostvarili su zadovoljavajuće rezultate na gotovo svakom skupu podataka. Na skupovima DREBIN i IMPACT ostvarene su nešto lošije performanse kod algoritme Naive Bayes i logistička regresija. Od svih algoritama, najbolje rezultate dao je algoritam stablo odluke. Gotovo u svim situacijama postigao je najveću točnost klasifikacije, čak i bez upotrebe tehnika predobrade. Algoritmi k-najbližih susjeda i nasumična šuma također ostvaruju vrlo dobre performanse, ali ipak nisu jednako konstanti kao algoritam stablo odluke. Naive Bayes algoritam kod najvećeg broja skupova podataka ostvaruje najlošije performanse. Ipak, kod ovog algoritma vidljiva su najveće poboljšanje u performansama nakon predobrade skupova podataka. Od svih tehnika odabira značajki najveće poboljšanje performansi postignuto je upotrebom SFS omotača, iako se kod svih tehnika ostvaruju u većini slučajeva ostvaruju bolji ili podjednako dobar rezultat smanjenjem broja značajki. Kod tehnika nasumičnog uzorkovanja boljom se pokazala tehnika preuzorkovanja.

Strojno učenje ima široku primjernu u kibernetičkoj sigurnosti te se sve više koristi za prepoznavanje malicioznih napada. Unatoč tome, potrebno je nastaviti razvijati nove i poboljšane modele kako bi se ostalo u koraku sa sve većim brojem modernih oblika napada. Veliki problem koji onemogućuje brži razvoj boljih modela je nedostatak kvalitetnih podataka. U domeni prepoznavanja malicioznih napada većina dostupnih skupova podataka je neuravnotežena ili zastarjela. Poboljšavanjem njihove kvalitete omogućit će se treniranje modela sa većom moći generalizacije.

LITERATURA

[1] R. Sharp, An Introduction to Malware. 2007.

[2] M. Ahsan, K. E. Nygard, R. Gomes, M. M. Chowdhury, N. Rifat, J. F. Conolly, Cybersecurity Threats and Their Mitigation Approaches Using Machine Learning – A Review, Journal of Cybersecurity and Privacy, sv. 2, str. (527 – 555), srpanj 2022.

[3] A. P. Namanya, A. Cullen, I. U. Awan, J. P. Disso, The World of Malware: An Overview, IEEE 6th International Conference on Future Internet of Things and Cloud, str. (420 – 427), 2018.

[4] R. Tahir, A study on malware and malware detection techniques, International Journal of Education and Management Engineering, sv. 8, br. 2, str. (20 – 30), ožujak 2018.

[5] D. Lupton, Panic computing: The viral metaphor and computer technology, Cultural Studies, sv. 8, br. 3, str. (556 – 568), kolovoz 2006.

[6] A. R. Muntode, S. S. Parwe, An Overview on Phishing- its types and Countermeasures, International Journal of Engineering Research & Technology, sv. 8, br. 12, str. (545 – 548), prosinac 2019.

[7] R. Sangita, A. K. Singh, A. S. Sairam, A Novel Approach to Prevent SQL Injection Attack Using URL Filter, International Journal of Innovation, Management and Technology, sv. 3, br. 5, str. (499 – 502), listopad 2012.

[8] A. Bendovschi, Cyber-Attacks – Trends, Patterns and Security Countermeasures, 7th International Conference on Financial Criminology, sv. 8, str. (24 – 31), Oxford, 2015

[9] A. M. Abuzaid, M. M. Saudi, B. M. Tiab, Z. H. Abdullah, An Efficient Trojan Horse Classification (ETC), International Journal of Computer Science, sv. 10, br. 3, str. (96 – 104), ožujak 2013.

[10] K. Shaukat, S. Luo, V. Varadharajan, I. A. Hameed, S. Chen, D. Liu, J. Li, Performance Comparison and Current Challenges of Using Machine Learning Techniques in Cybersecurity, Energies, sv. 13, br. 10, str. (1 – 27), svibanj 2020.

[11] J. B. Fraley, J. Cannady, The Promise of Machine Learning in Cybersecurity, Conference SoutheastCon, 2017.

[12] L. D`hooge, T. Wauters, B. Volckaert, F. De Turck, In - depth Comparative Evaluation of Supervised Machine Learning Approaches for Detection of Cybersecurity Threats, Proceedings of

the 4th International Conference on Internet of Things, Big Data and Security, str. (125 – 136), Heraklion, 2019.

[13] H. Debar, An Introduction to Intrusion-Detection Systems, 2009.

[14] C-A. D. Tsiakos, C. Chalkias, Use of Machine Learning and Remote Sensing Techniques for Shoreline Monitoring: A Review of Recent Literature, Appl. Sci., sv. 13, br. 5, str. (1 – 20), ožujak 2023.

[15] J. M. Torres, C. I. Comesana, P. J. Garcia – Nieto, Review: machine learning techniques applied to cybersecurity, International Journal of Machine Learning and Cybernetics, sv. 10, str. (2823 – 2836), 2019.

[16] M. Dudjak, Učenje iz neuravnoteženih podataka unaprijeđenim postupcima za odabir značajki, preuzorkovanje i izgradnju radijalnih neuronskih mreža, Sveučilište J. J. Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek, Doktorska disertacija, 2022.

[17] K. P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press, London, 2012.

[18] A. Handa, A. Sharma, S. K. Shukla, Machine learning in cybersecurity: A review, WIREs Data Mining Knowl Discov., sv. 9, br. 4, str. (1 – 7), siječanj 2019.

[19] M. Dash, H. Liu, Feature selection for classification, Intelligent data analysis, sv. 1, br. 3, str. (131 – 156), ožujak 1997.

[20] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, A. Ng, Cybersecurity data science: an overview from machine learning perspective, Journal of Big Data, sv. 7, br. 41, str. (1 -29), lipanj 2020.

[21] A. F. Alnuaimi, T. H. Albadawi, An overview of machine learning classification techniques, BIO Web of Conferences, sv. 97, str. (1 – 24), ožujak 2020.

[22] M. Aghaabbasi, A. Mujahid, M. Jasinski, Z. Leonowicz, T. Novak ,On Hyperparameter Optimization of Machine Learning Methods Using a Bayesian Optimization Algorithm to Predict Work Travel Mode Choice, IEEE Access, sv. 11, str. (19762 - 19774), veljača 2023.

[23] B. Mahesh, Machine Learning Algorithms - A Review, International Journal of Science and Research, sv. 9, br. 1, str. (381 – 386), siječanj 2020.

[24] J. Miao, L. Niu, A Survey on Feature Selection, Procedia computer science, sv. 91, str. (919 - 926), siječanj 2016.

[25] B. Venkatesh, J. Anuradha, A Review of Feature Selection and Its Methods, Cybernetics and Information Technologies, sv. 19, br. 1, str. (2 – 25), ožujak 2019.

[26] R. Kovahi, G. H. John, Wrappers for feature subset selection, Artificial intelligence, sv. 97, br. 1-2, str. (273 – 324), prosinac 1997.

[27] Y. B. Wah, N. Ibrahim, H. A. Hamid, S. Abdul-Rahman, S. Fong, Feature Selection Methods: Case of Filter and Wrapper Approaches for Maximising Classification Accuracy, Pertanika Journal of Science & Technology, sv. 26, br. 1, str. (330 - 339), rujan 2017.

[28] B. Krawczyk, Learning from imbalanced data: open challenges and future directions, Progress in artificial intelligence, sv. 5, br. 4, str (221 – 232), travanj 2016.

[29] H. Wang, A. Singhal, P. Liu, Tackling imbalanced data in cybersecurity with transfer learning: a case with ROP payload detection, Cybersecurity, sv. 6, br. 1, str (1 – 15), siječanj 2023.

[30] A., Goldbloom, Kaggle [online], San Francisco, 2010., dostupno na: https://www.kaggle.com/ [16.8.2024.]

[31] D., Aha, UCI Machine Learning Repository, California, 1987., dostupno na: https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html [16.8.2024.]

[32] University of New Brunswick, dostupno na: https://www.unb.ca/cic/datasets/ids.html [19.8.2024.]

[33] UNSW Sydney, dostupno na: https://research.unsw.edu.au/projects/unsw-nb15-dataset [21.8.2024.]

SAŽETAK

Cilj ovog rada bio je odrediti i primijeniti tok učenja odabranih algoritama klasifikacije iz skupova podataka koji opisuju različite vrste malicioznih napada i definirati procedure unutar tog toka. Opisani su maliciozni napadi koji predstavljaju prijetnju te se pruža uvid u tradicionalne načine obrane. Osim toga, objašnjen je pojam strojnog učenja i zadatak klasifikacije kao posebne vrste nadziranog strojnog učenja. Pregledom literature, prikazani su problemi u kojima je korišteno strojno učenje za izgradnju sustava za obranu od malicioznih napada. Analizirani su klasifikacijski algoritmi korišteni u radu te su predstavljene tehnike za odabir značajki i rad s neuravnoteženim skupovima podataka. Procijenjena je efikasnost metoda za odabir značajki i tehnika uzorkovanja te je analiziran njihov doprinos poboljšanju modela. Prilikom klasifikacije malicioznih napada najbolje rezultate ostvario je algoritam stablo odluke. Nasuprot njemu, najlošije se pokazao algoritam Naive Bayes. Upotrebom tehnika za odabir značajki modeli ostvaruju jednako dobre ili bolje performanse smanjenjem broja značajki. SFS omotač je jedna od tehnika koja se ističe zbog doprinosa u poboljšanju modela. Smanjenjem stupnja neuravnoteženosti skupova podataka performanse rastu upotrebom tehnike nasumičnog preuzorkovanja.

Ključne riječi: klasifikacija, maliciozni napadi, neuravnoteženi skup podataka, odabir značajki, strojno učenje

ABSTRACT

Title: Malicious attack detection using machine learning

The goal of this work was to determine and apply the learning flow of selected classification algorithms from data sets describing different types of malicious attacks and to define the procedures within that flow. Malicious attacks are described, and traditional defenses against them are provided. In addition, the concept of machine learning and the task of classification as a special type of supervised machine learning are explained. By reviewing the literature, problems in which machine learning was used to build a system to defend against malicious attacks were presented. The classification algorithms used in the work were analyzed and the techniques for features selection and handeling imbalanced data sets were presented. The effectiveness of feature selection methods and sampling techniques was evaluated and their contribution to model improvement was analyzed. When classifying malicious attacks, the decision tree algorithm achieved the best results. In contrast, the Naive Bayes algorithm performed the worst. By using feature selection techniques, models achieve equal or better performance by reducing the number of features. The SFS wrapper is one of the techniques that stands out for its contribution to model improvement. By reducing the imbalance ratio of data sets, performance increases using the random oversampling technique.

Keywords: classification, feature selection, imbalanced data set, machine learning, malicious attack

PRILOZI

P.4.1. Tablica hiperparametara algoritma stablo odluke za svaki skup podataka

Skup podataka	Kriterij za podjelu	Maksimalna dubina	Strategija za podjelu
DARPA	entropija	20	najbolji
KDD99	entropija	15	nasumično
NSL-KDD	entropija	15	najbolji
KYOTO	entropija	20	najbolji
Malware	gini	5	najbolji
DREBIN	gini	20	nasumično
ISCXIDS2012	entropija	5	nasumično
CICIDS2017	gini	20	nasumično
DS2OS	gini	15	nasumično
IMPACT	entropija	20	najbolji
UNSW-NB15	entropija	20	najbolji
CIC-DDOS2019	entropija	20	najbolji

P.4.2. Tablica hiperparametra algoritma *k*-najbližih susjeda za svaki skup podataka

Skup podataka	Algoritam za pretraživanje susjeda	Broj susjeda	Težina susjeda
DARPA	automatski	5	udaljenost
KDD99	automatski	2	udaljenost
NSL-KDD	automatski	2	udaljenost
KYOTO	automatski	4	udaljenost
Malware	automatski	5	udaljenost
DREBIN	automatski	2	udaljenost
ISCXIDS2012	automatski	2	udaljenost
CICIDS2017	automatski	2	udaljenost
DS2OS	automatski	2	udaljenost
IMPACT	automatski	5	udaljenost
UNSW-NB15	automatski	4	udaljenost
CIC-DDOS2019	automatski	5	udaljenost

P.4.3. Tablica hiperparametra algoritma logistička regresija za svaki skup podataka

Skup podataka	C	Prilagodba presjeka	Kazna	Optimizacijski algoritam
DARPA	2	Da	l2	lbfgs
KDD99	2	Ne	l2	lbfgs
NSL-KDD	2	Da	l2	lbfgs
KYOTO	1	Da	l2	lbfgs
Malware	2	Da	l2	lbfgs
DREBIN	2	Ne	l2	lbfgs
ISCXIDS2012	2	Da	l2	lbfgs
CICIDS2017	2	Ne	l2	lbfgs
DS2OS	2	Da	l2	lbfgs
IMPACT	2	Da	l2	lbfgs
UNSW-NB15	2	Da	l2	lbfgs
CIC-DDOS2019	2	Ne	l2	lbfgs

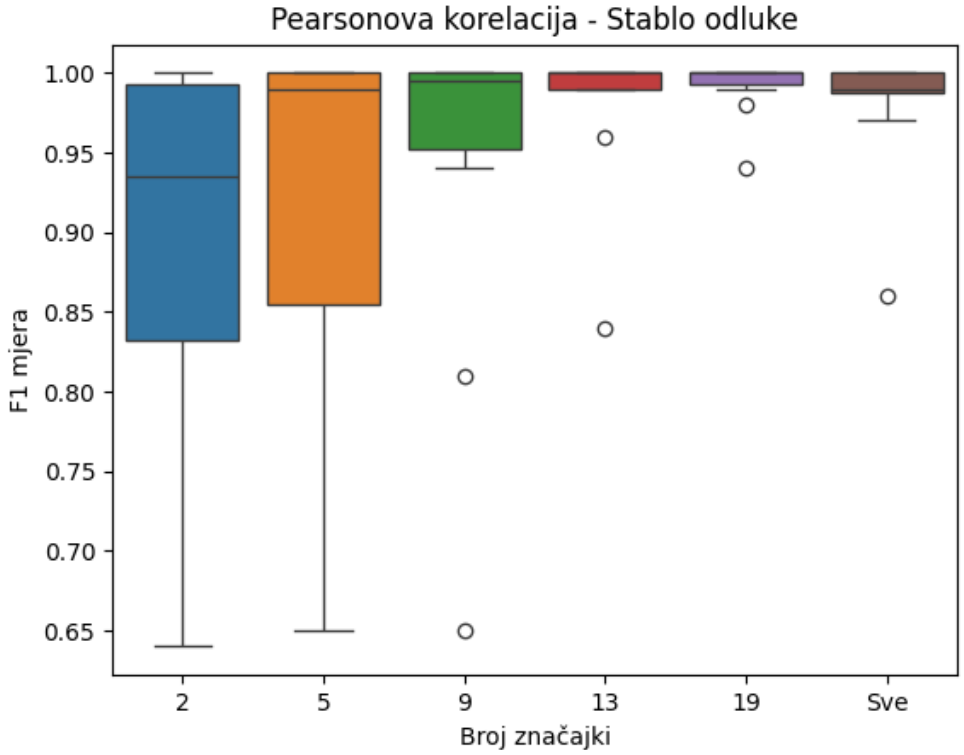
P.4.4. Tablica hiperparametra algoritma nasumična šuma za svaki skup podataka

Skup podataka	Maksimalna dubina	Minimalni broj uzoraka u listu	Broj stabala
DARPA	10	1	200
KDD99	10	1	200
NSL-KDD	10	2	100
KYOTO	10	1	200
Malware	5	1	150
DREBIN	10	1	100
ISCXIDS2012	5	1	200
CICIDS2017	10	1	200
DS2OS	10	1	200
IMPACT	10	1	100
UNSW-NB15	10	1	100
CIC-DDOS2019	10	1	200

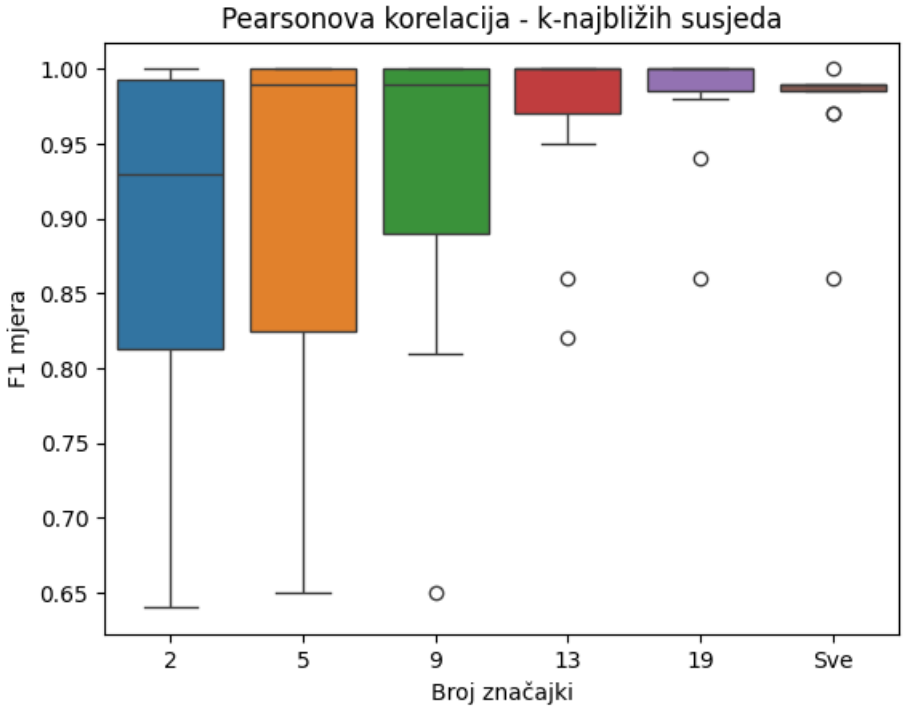
P.4.5. Tablica hiperparametra algoritma Naive Bayes za svaki skup podataka

Skup podataka	**Izjednačavanje varijance**
DARPA	0.1
KDD99	1
NSL-KDD	1
KYOTO	0.0001
Malware	0.1
DREBIN	0.01
ISCXIDS2012	1
CICIDS2017	0.0000001
DS2OS	1
IMPACT	1
UNSW-NB15	0.000000001
CIC-DDOS2019	0.0001

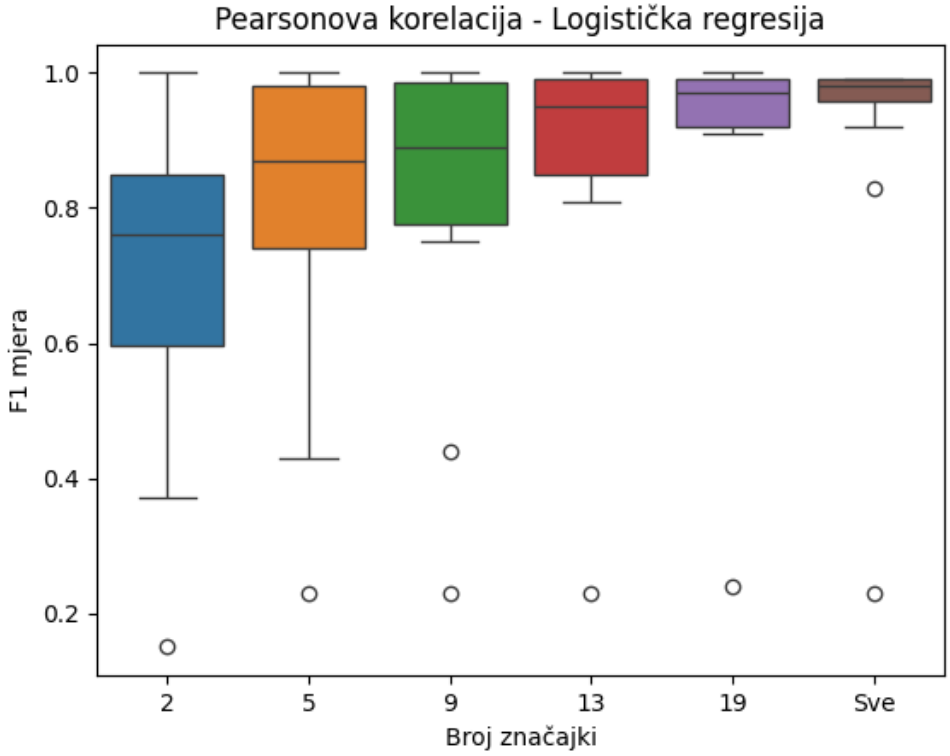
P.4.6. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma stablo odluke pri različitom broju značajki odabranih filtrom



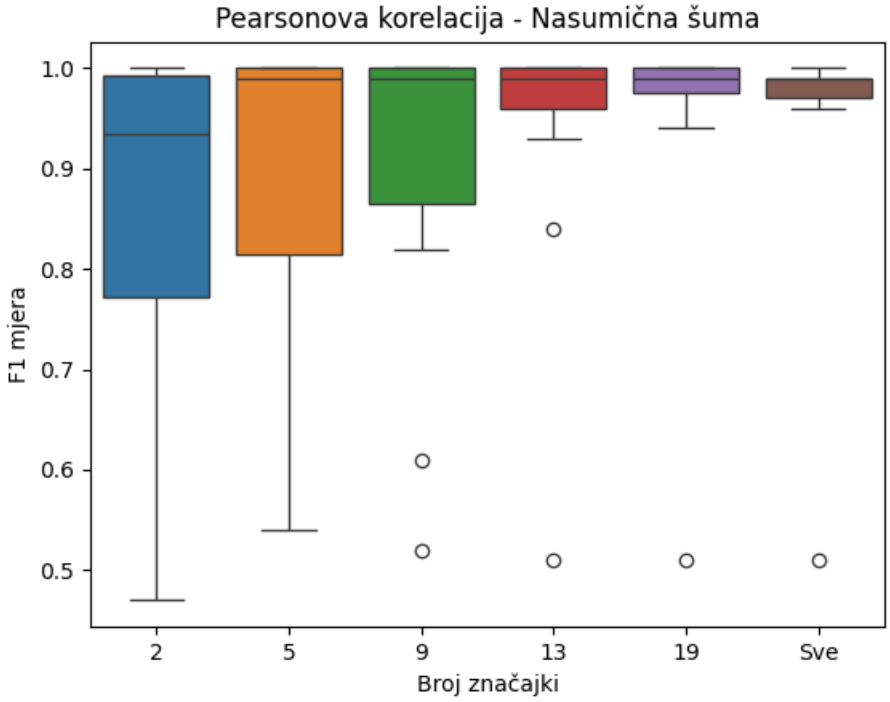
P.4.7. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma *k*-najbližih susjeda pri različitom broju značajki odabranih filtrom



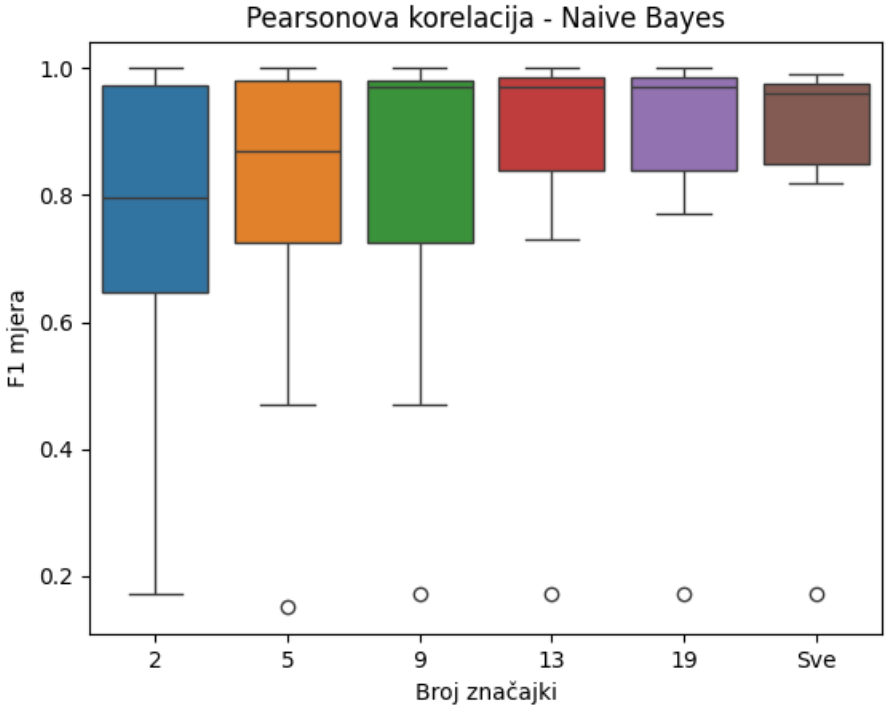
P.4.8. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma logistička regresija pri različitom broju značajki odabranih filtrom



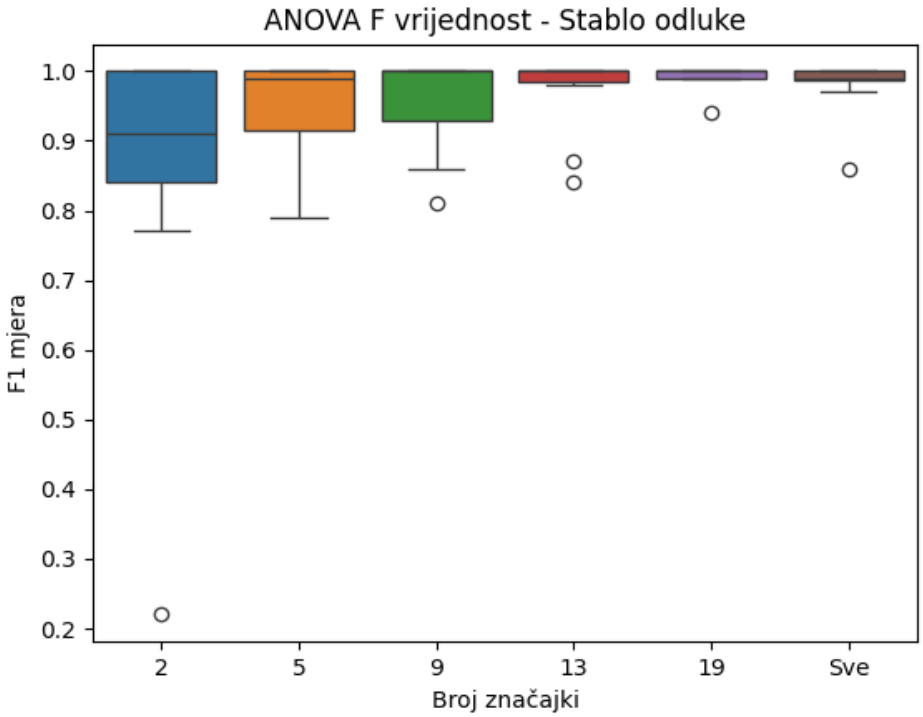
P.4.9. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma nasumična šuma pri različitom broju značajki odabranih filtrom



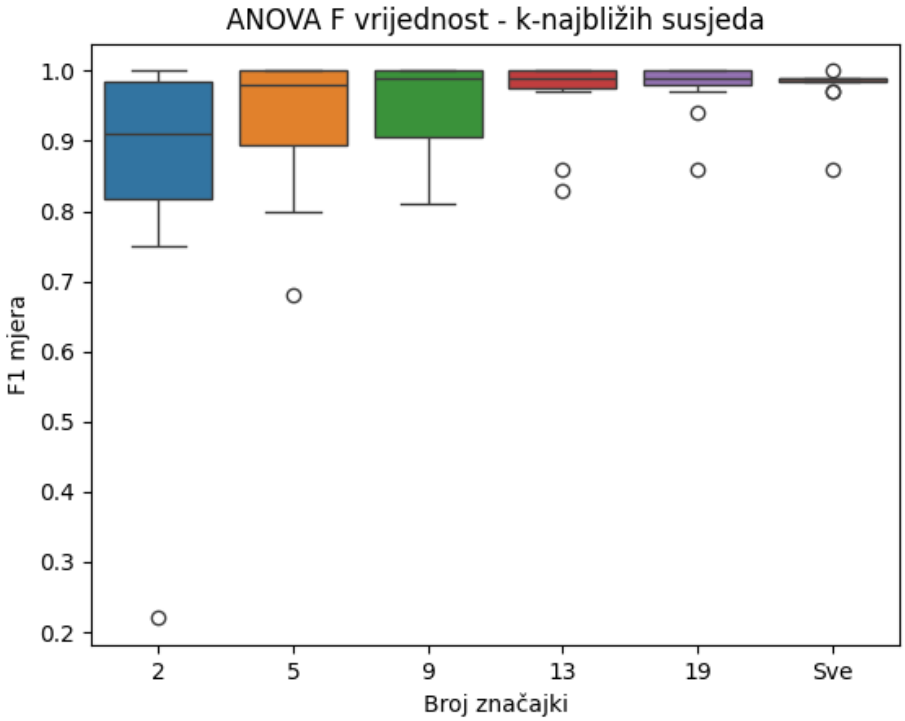
P.4.10. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma Naive Bayes pri različitom broju značajki odabranih filtrom



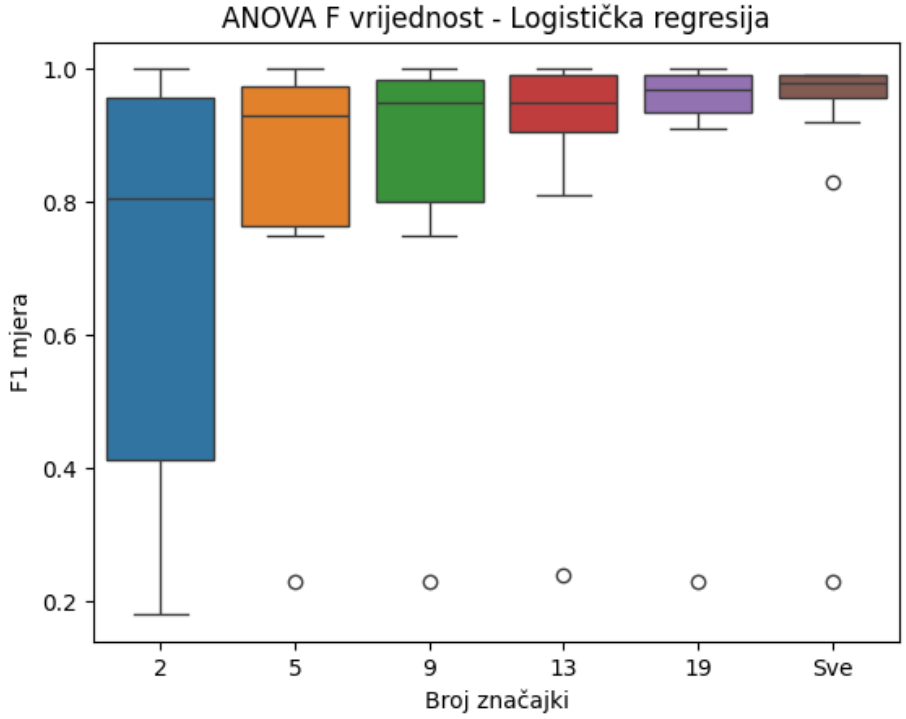
P.4.11. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma stabla odluke pri različitom broju značajki odabranih filtrom



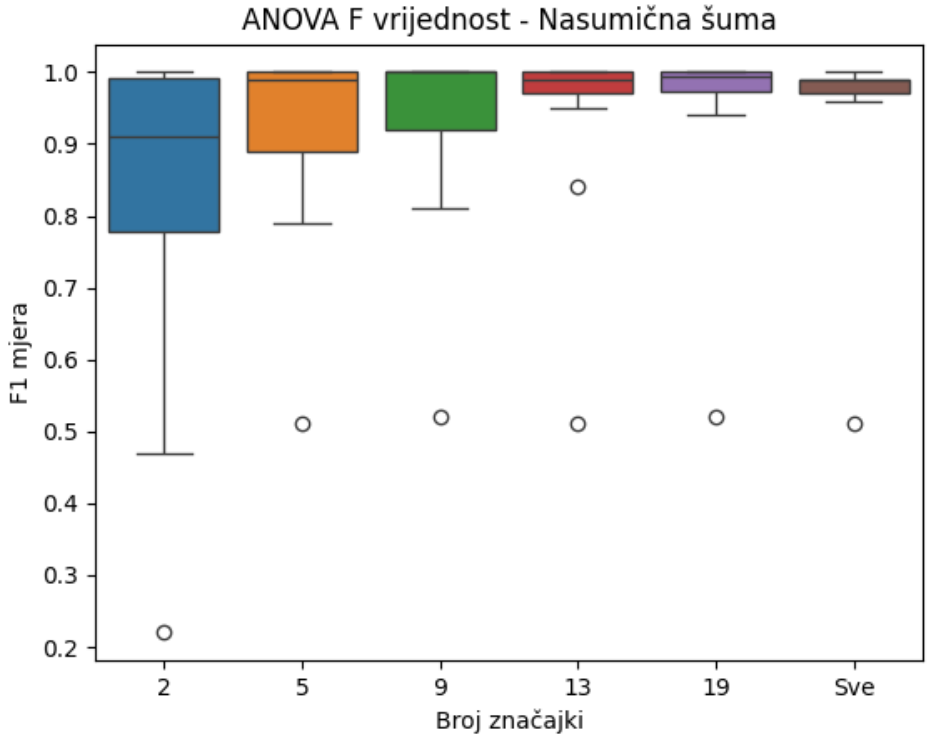
P.4.12. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma *k*-najbližih susjeda pri različitom broju značajki odabranih filtrom



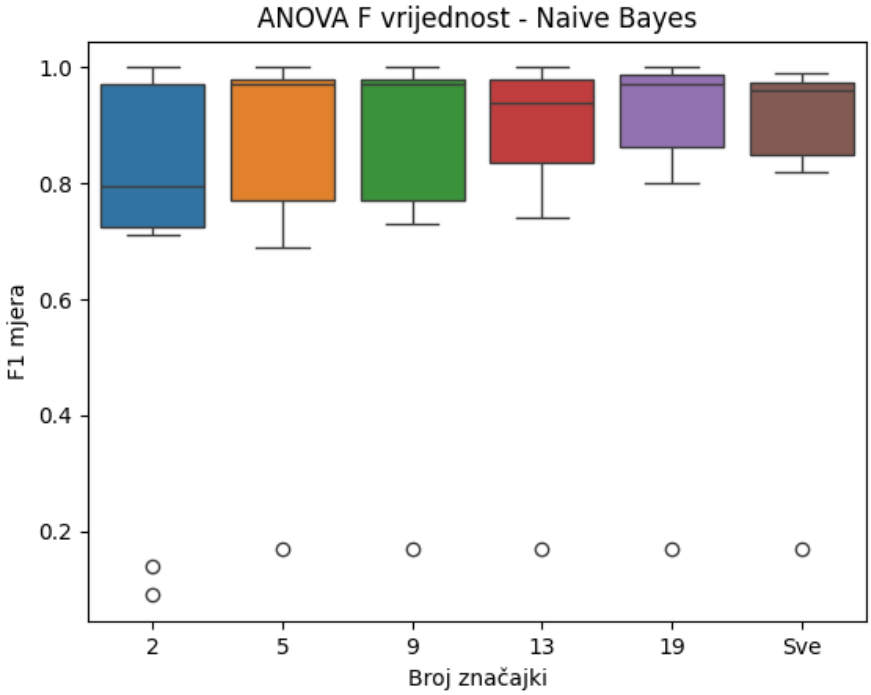
P.4.13. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma logistička regresija pri različitom broju značajki odabranih filtrom



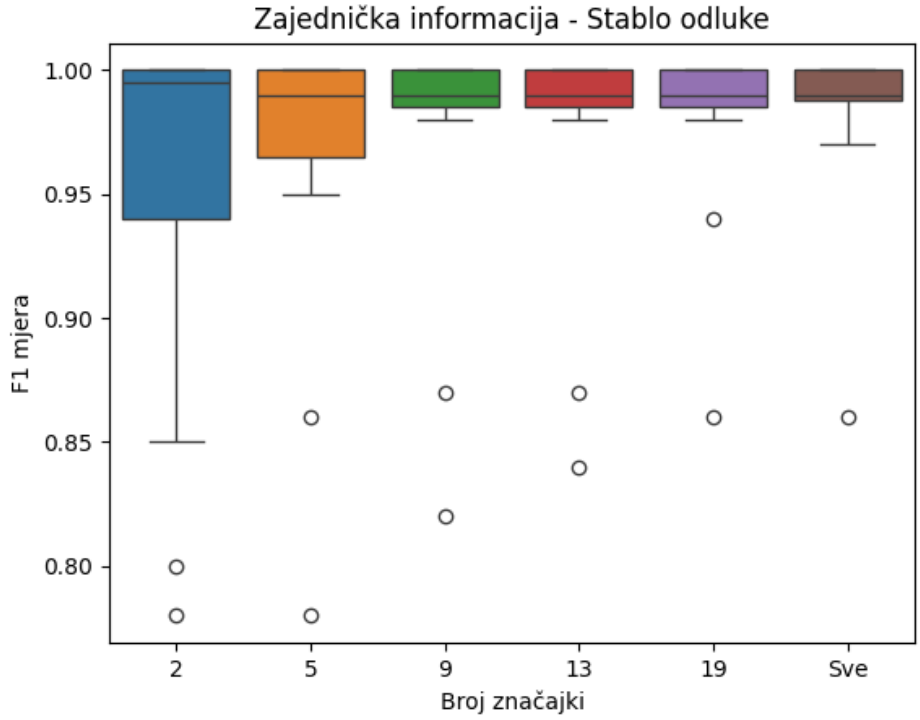
P.4.14. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma nasumična šuma pri različitom broju značajki odabranih filtrom



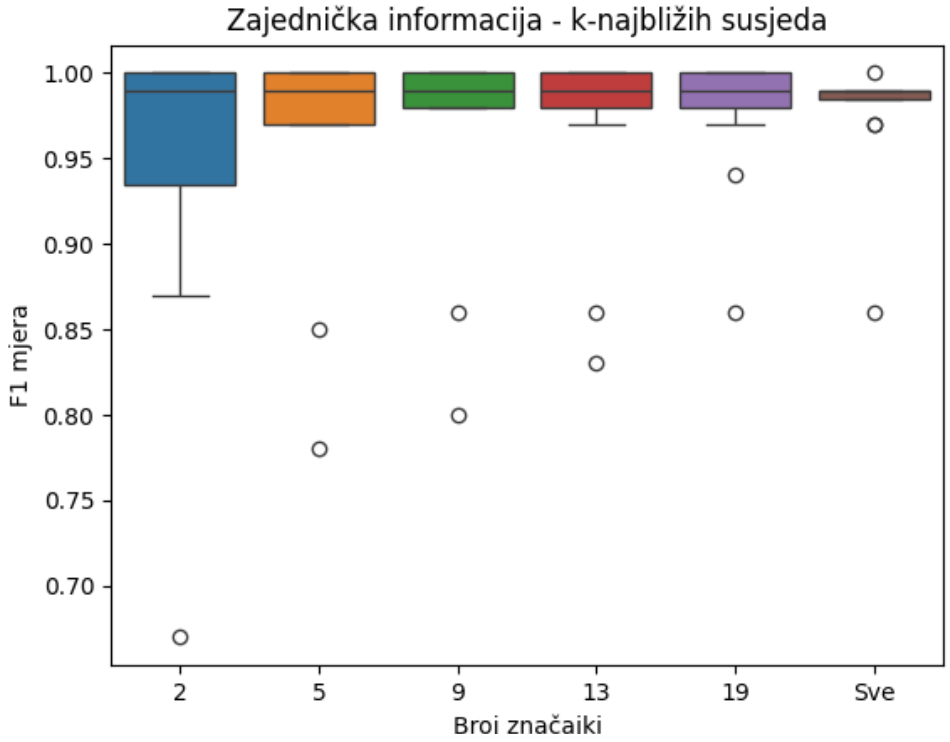
P.4.15. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma Naive Bayes pri različitom broju značajki odabranih filtrom



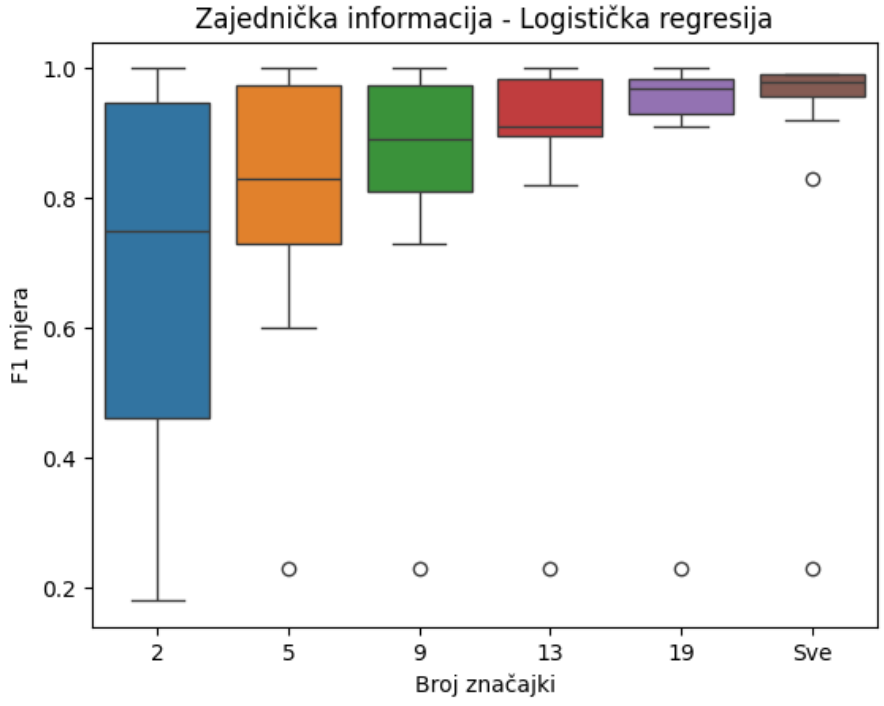
P.4.16. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma stablo odluke pri različitom broju značajki odabranih filtrom



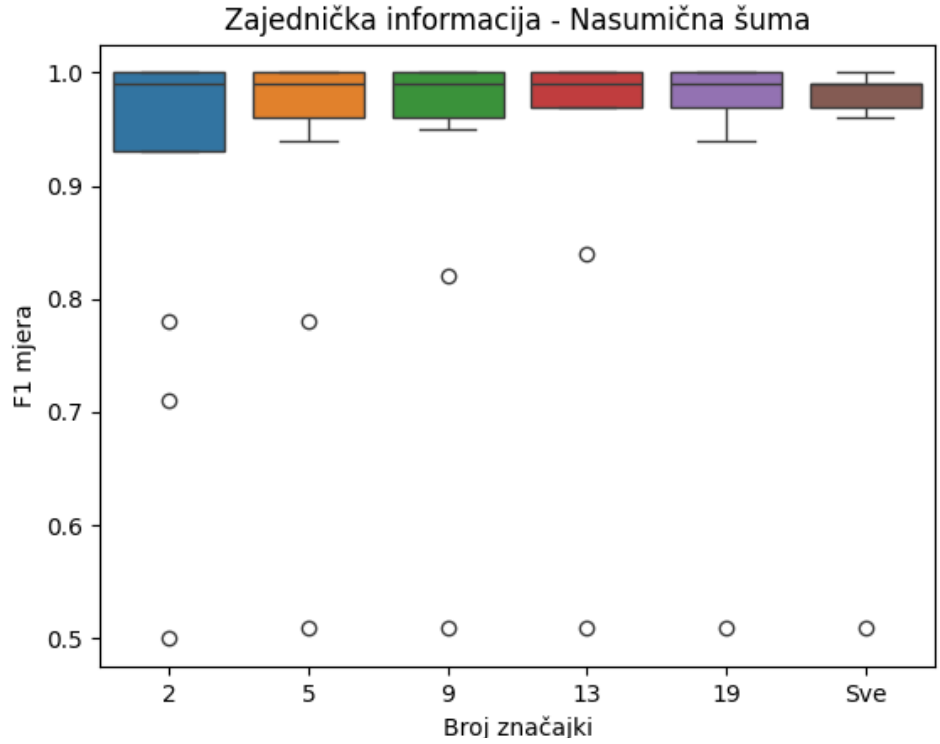
P.4.17. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma *k*-najbližih susjeda pri različitom broju značajki odabranih filtrom



P.4.18. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma logistička regresija pri različitom broju značajki odabranih filtrom



P.4.19. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma nasumična šuma pri različitom broju značajki odabranih filtrom



P.4.20. Slika dijagrama pravokutnika vrijednosti F1 mjere algoritma Naive Bayes pri različitom broju značajki odabranih filtrom