Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek

Računarstvo usluga i analiza podataka

SEMINARSKI RAD

„Klasifikacija lijekova“

Ana-Marija Katić

Osijek, 2022.

Sadržaj

[1. Uvod 1](#_Toc95153855)

[2. Pregled problema 2](#_Toc95153856)

[2.1. Korišteni podaci 2](#_Toc95153857)

[2.2. Korišteni postupci strojnog učenja 4](#_Toc95153858)

[3. Opis programskog rješenja 5](#_Toc95153859)

[3.1. Model strojnog učenja 5](#_Toc95153860)

[3.2. Klijentska aplikacija 9](#_Toc95153861)

[4. Zaključak 12](#_Toc95153862)

[5. Poveznice i literatura 13](#_Toc95153863)

# Uvod

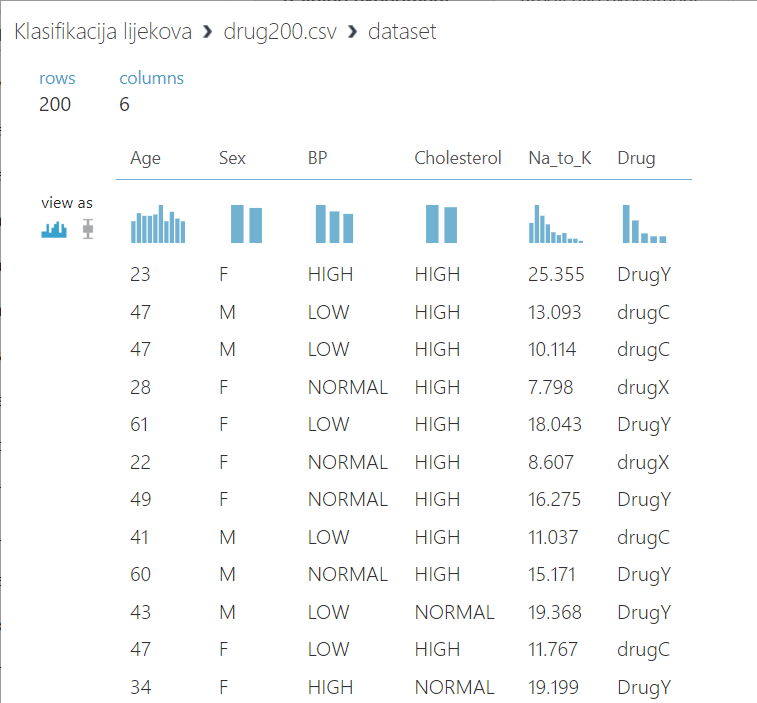
Ljudi su svakodnevno izloženi velikoj količini novih informacija i različitih podataka. Zbog toga se koriste računala i strojno učenje kako bi se donijele adekvatne odluke na temelju informacija koje su poznate. U medicini je donošenje ispravnih odluka prioritet, jer neispravne odluke mogu imati značajne negativne posljedice na kvalitetu života pojedinca. Različite bolesti imaju slične simptome, zbog čega liječnici trebaju znati kada se koji lijekovi koriste. U slučaju novih lijekova teško je znati kakve će nuspojave pojedinci imati, ali analizom podataka liječnici stječu nova znanja o djelovanju lijekova. Ovim projektom je napravljena klasifikacija lijekova na temelju poznati parametara pacijenata. Klasifikacije lijekova trebala bi olakšati liječnicima donošenje odluka koje će lijekove propisati pojedinim pacijentima. S obzirom da se propisivanjem krivog lijeka mogu izazvati znatne nepoželjne posljedice, u projektu se teži maksimalnoj točnosti i preciznosti pri predviđanju i kategoriziranju lijekova.

# Pregled problema

Kako bi se odradila klasifikacija lijekova prvo je bilo potrebno prikupiti podatke s parametrima na temelju kojih se određeni lijekovi propisuju. Nakon prikupljanja podataka bilo je potrebno proučiti i pripremiti podatke kako bi ih se bolje razumjelo i lakše koristilo prilikom strojnog učenja. Pomoću pripremljenih podataka napravljen je model s različitim algoritmima za višeklasnu klasifikaciju. On je istreniran i testiran na podacima te je odabiran najprikladniji algoritma koji je korišten u razvijenoj klijentskoj aplikaciji.

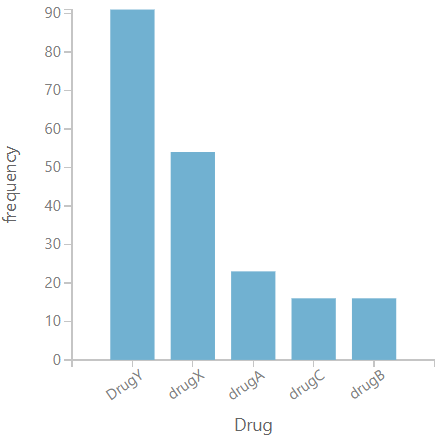
## Korišteni podaci

Korišteni podaci predstavljaju skup podataka *drug200* koji se sastoji od 200 različitih podataka koji sadrže informacije o određenim vrstama lijekova. Ovaj skup podataka preuzet je sa stranice *Kaggle* koja nudi veliku količinu objavljenih podataka od strane zajednice, a korišteni skup podataka objavio je Pratham Tripathi. Skup podataka je lako razumljiv i nema pojave izostavljenih vrijednosti. Skup podataka se sastoji od pet značajki na temelju kojih se određuje šesta značajka. Slika 2.1.1. prikazuje značajke skupa podataka gdje „Age“ značajka predstavlja godine pacijenata, „Sex“ spol pacijenata, „BP“ razinu krvnog tlaka, „Cholesterol“ razinu kolesterola u krvi, „Na\_to\_K“ omjer natrija i kalija u krvi te „Drug“ koja predstavlja vrstu lijeka.



*Slika 2.1.1. Prikaz značajki skupa podataka drug200*

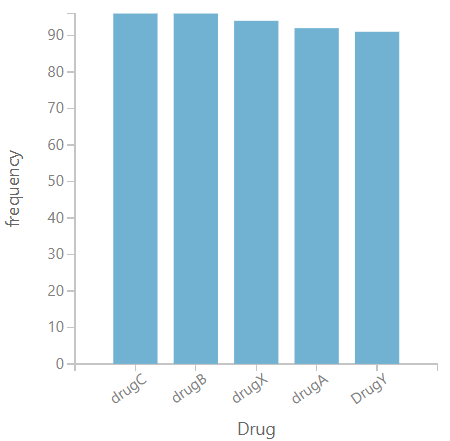
Prije rada s podacima, odnosno njihove obrade, podaci se trebaju provjeriti i pripremiti za korištenje. Prvo se provjerava distribucija podataka kako bi se provjerio broj uzoraka, odnosno frekvencija određenih podataka. Podaci za godine, spol, razinu krvnog tlaka i razinu kolesterola u krvi su uravnoteženi, odnosno imaju približan broj uzoraka za svaku kategorija. S druge strane podaci za omjer natrija i kalija u krvi i vrste lijekova su neuravnoteženi.



Slika 2.1.2. Graf prikazuje broj uzoraka određene vrste lijeka

Iz stupčastog grafa na slici 2.1.2. može se vidjeti da je broj uzoraka lijeka „DrugY“ puno veći od korištenja ostalih vrsta lijekova. To predstavlja problem kod klasifikacije jer uzoraka za jednu vrstu lijekova ima značajno više i njih je lakše predvidjeti i klasificirati u odnosu na ostale četiri vrste kod kojih će rezultati biti neprecizni. Za problem klasifikacije bilo bi idealno imati sličan broj uzoraka različitih vrsta lijekova. Za rješavanje problema neuravnoteženosti klasa, korišten je algoritam SMOTE (engl. *Synthetic Minority Oversampling Techinque*). Algoritma SMOTE obuhvaća preveliko uzorkovanje (engl *oversampling*) što podrazumijeva stvaranje novih uzoraka manjinske klase dupliciranjem ili stvaranjem sintetičkih podataka na temelju već postojećih. Određivanjem klase „DrugY“ kao većinske i ostalih slijedno kao manjinske, te primjenom algoritama SMOTE i k-najbližih susjeda, čija je vrijednost postavljena na tri, riješen je problem neuravnoteženosti klasa. Za svaku manjinsku klasu postavljen je prikladan postotak prevelikog uzorkovanja novih podataka, kako bi distribucija bila uravnotežena. Slika 2.1.3. prikazuje novu uravnoteženu distribuciju podataka klasa.

Nakon uravnoteženja klasa odabrane su sve značajke, iako značajka spol nema utjecaj na klasifikaciju lijeka. To je učinjeno kako bi je po potrebi mogao koristiti prošireni skup podataka u kojem vrsta lijeka ovisi o spolu. Na temelju tih podataka model za klasifikaciju vrsta lijekova će biti treniran i testiran.



Slika 2.1.3. Graf prikazuje broj uzoraka određene vrste lijeka nakon primjena SMOTE algoritma

## Korišteni postupci strojnog učenja

Ovaj projekt se temelji na klasifikaciji. Klasifikacija se uobičajeno sastoji od dvije faze, trening faze i testne faze. Nakon pripreme podataka za rad potrebno je odvojiti podatke na podatke za trening i podatke za testiranje. Korištena je podjela podataka 75% za treniranje i 25% za testiranje. Algoritam klasifikacije tijekom trening faze pokušava odrediti uzorak, odnosno funkciju na temelju koje u testnoj fazi klasifikator određuje kojoj klasi uzorak pripada.

Ne postoji najbolji klasifikator za sve skupove podataka već je potrebno usporediti više klasifikatora i utvrditi koji je najprecizniji i najtočniji za korišteni skup podataka. Zbog toga uspoređeno je više višeklasnih klasifikacijskih algoritama kako bi se mogao odabrati najoptimalniji za skup podataka *drug200*. Višeklasni algoritmi za klasifikaciju, čije su vrijednosti uspoređivane su šume odlučivanja, džungle odlučivanja, logistička regresija i neuronske mreže. Algoritam višeklasna šuma odlučivanja podržava podatke s različitim distribucijama, a radi tako da gradi više stabala odlučivanja i zatim glasa o najpopularnijoj izlaznoj klasi. Algoritam višeklasne džungle odlučivanja proširuje algoritam šuma odlučivanja te je precizniji. Algoritam višeklasne logistička regresije je primjerenija za probleme s dvije klase. Algoritam višeklasne neuronske mreže se koristi za ciljane značajke s više vrijednosti.

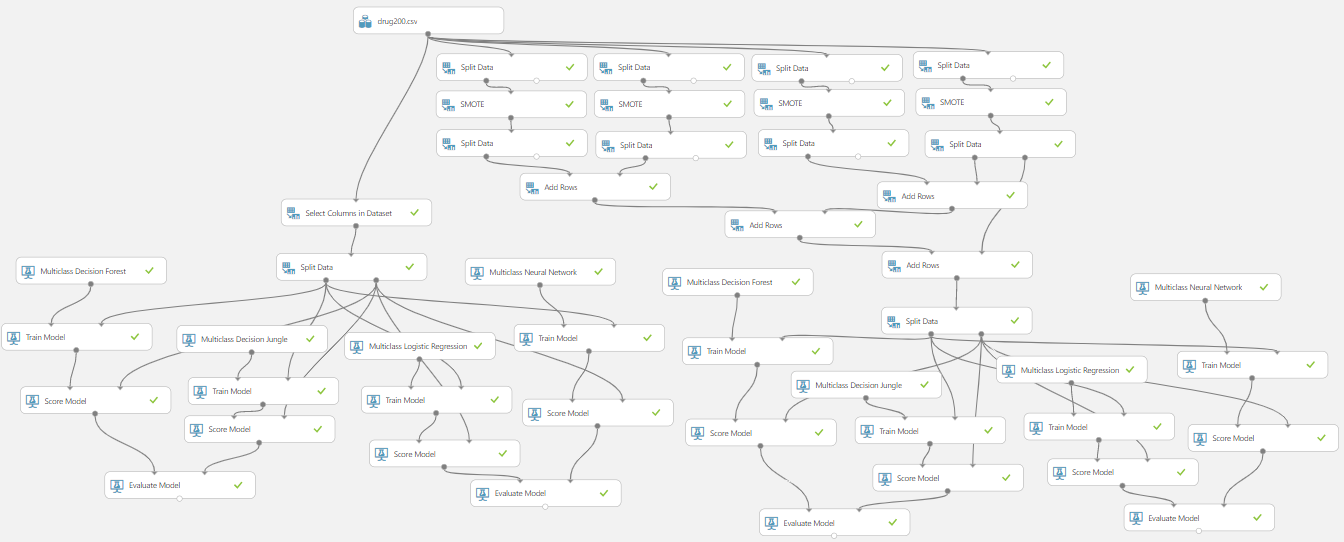
Nakon treniranja i testiranja kako bi se odabrao najprecizniji algoritam korištena je matrica zbunjenosti koja na jednostavan i razumljiv način prikazuje točnost i preciznost klasifikatora. Kod matrice zbunjenosti stupci predstavljaju stvarne vrijednosti, a redci predviđene vrijednosti uzoraka.

# Opis programskog rješenja

Model za klasifikaciju lijekova napravljen je u Azure Machine Learning Studiu gdje su uspoređeni različiti algoritmi višeklasne klasifikacije. Rezultate koje vrati web servis prikazuje se na izrađenoj C# desktop aplikaciji.

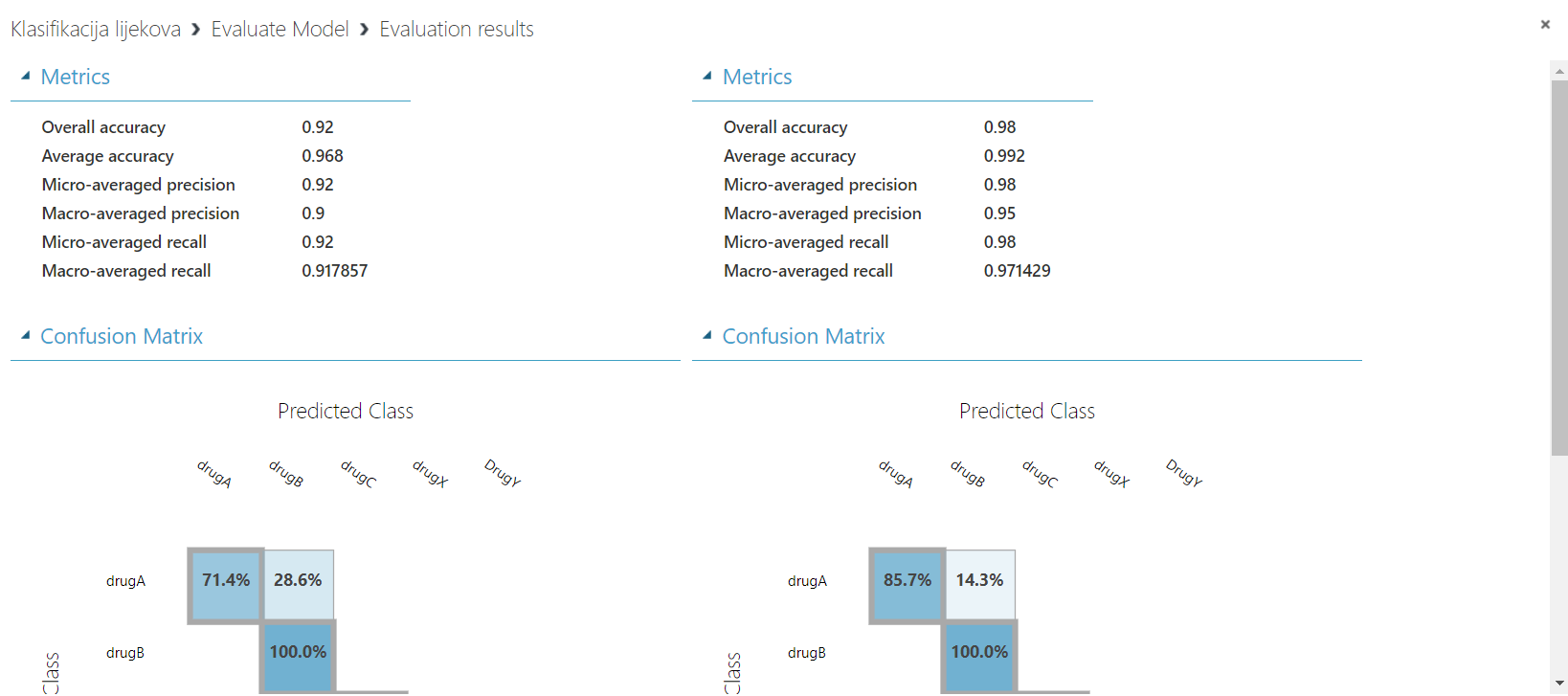
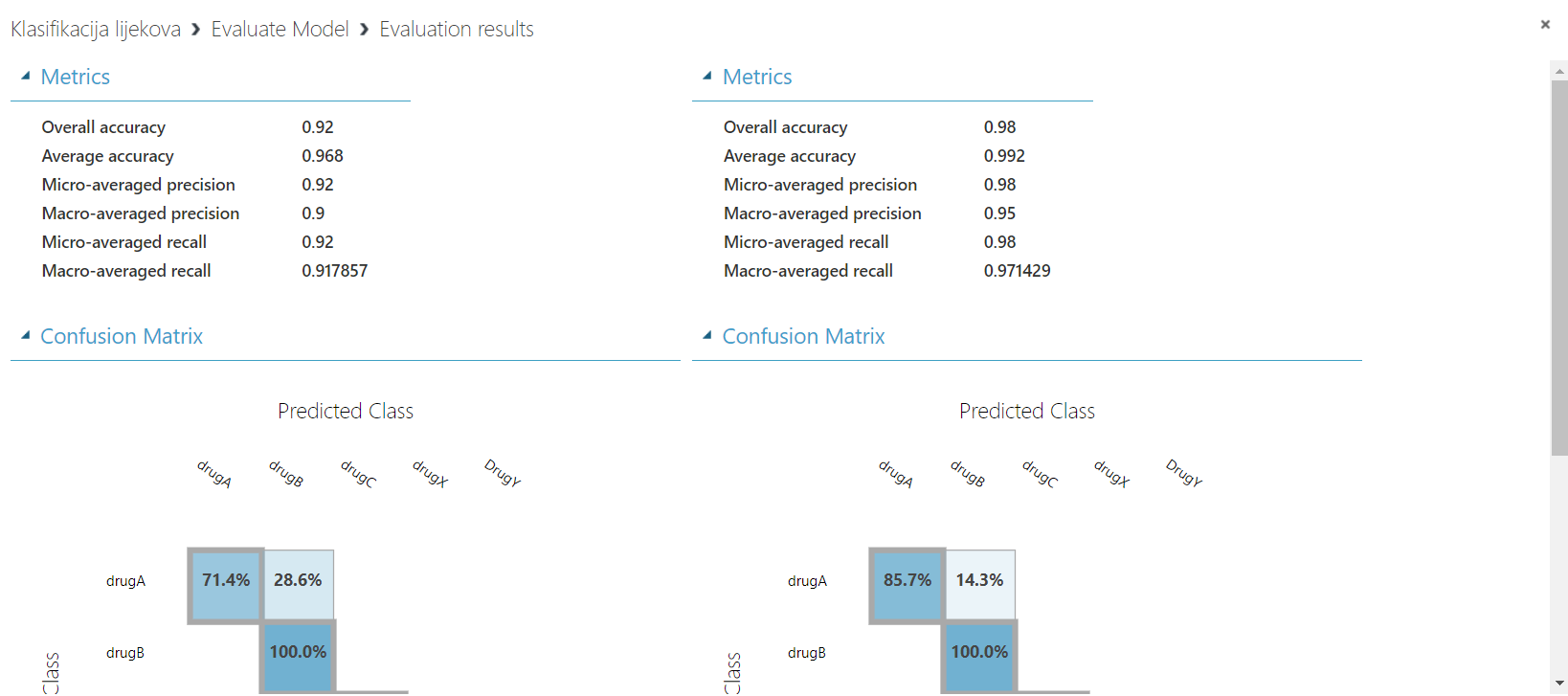
## Model strojnog učenja

U poglavlju „Korišteni postupci strojnog učenje“ sažeto je opisana izrada modela strojnog učenja, a ovdje će to biti detaljnije opisano. Model na slici 3.1.1. se sastoji od osam modela za klasifikaciju lijekova. Četiri modela s desne stran rade s uravnoteženim klasama, dok četiri modela s lijeve strane rade s neuravnoteženim ciljanim klasama.

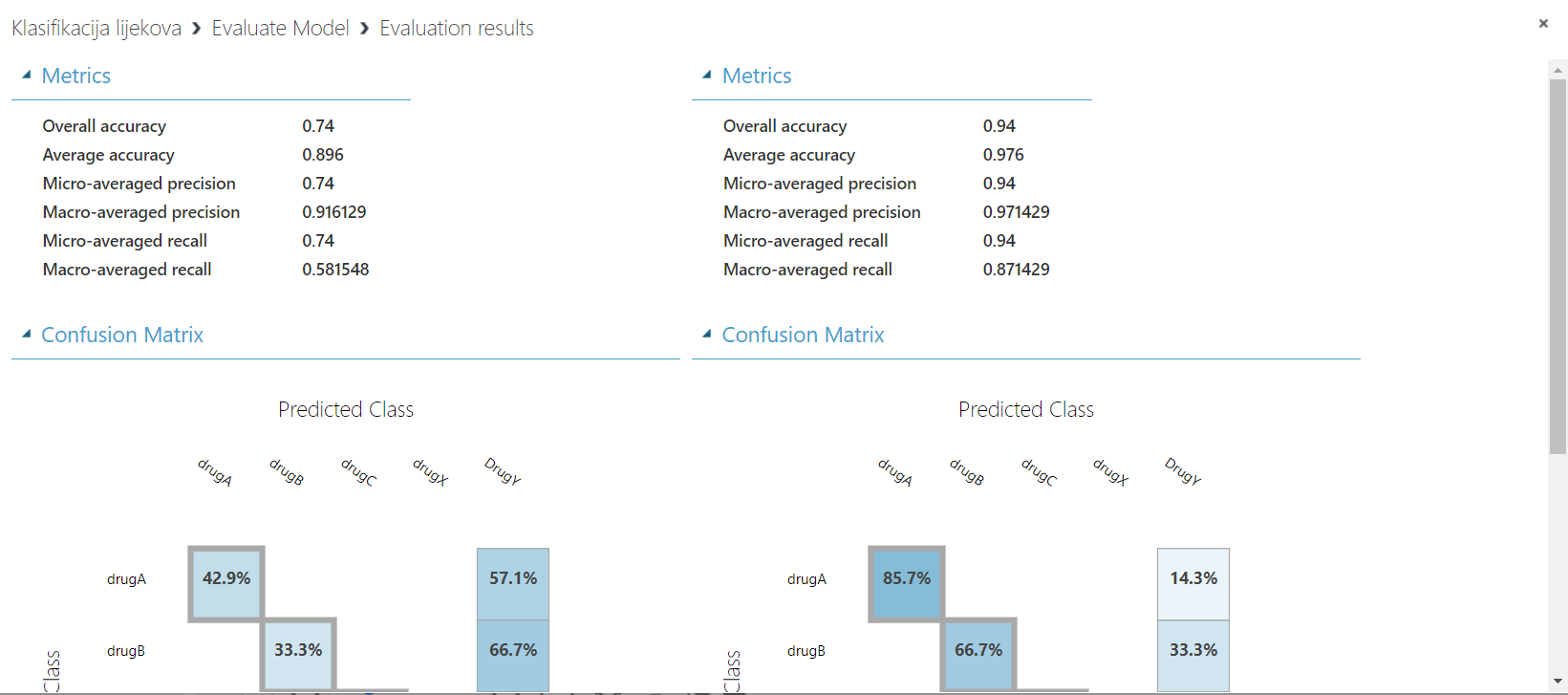
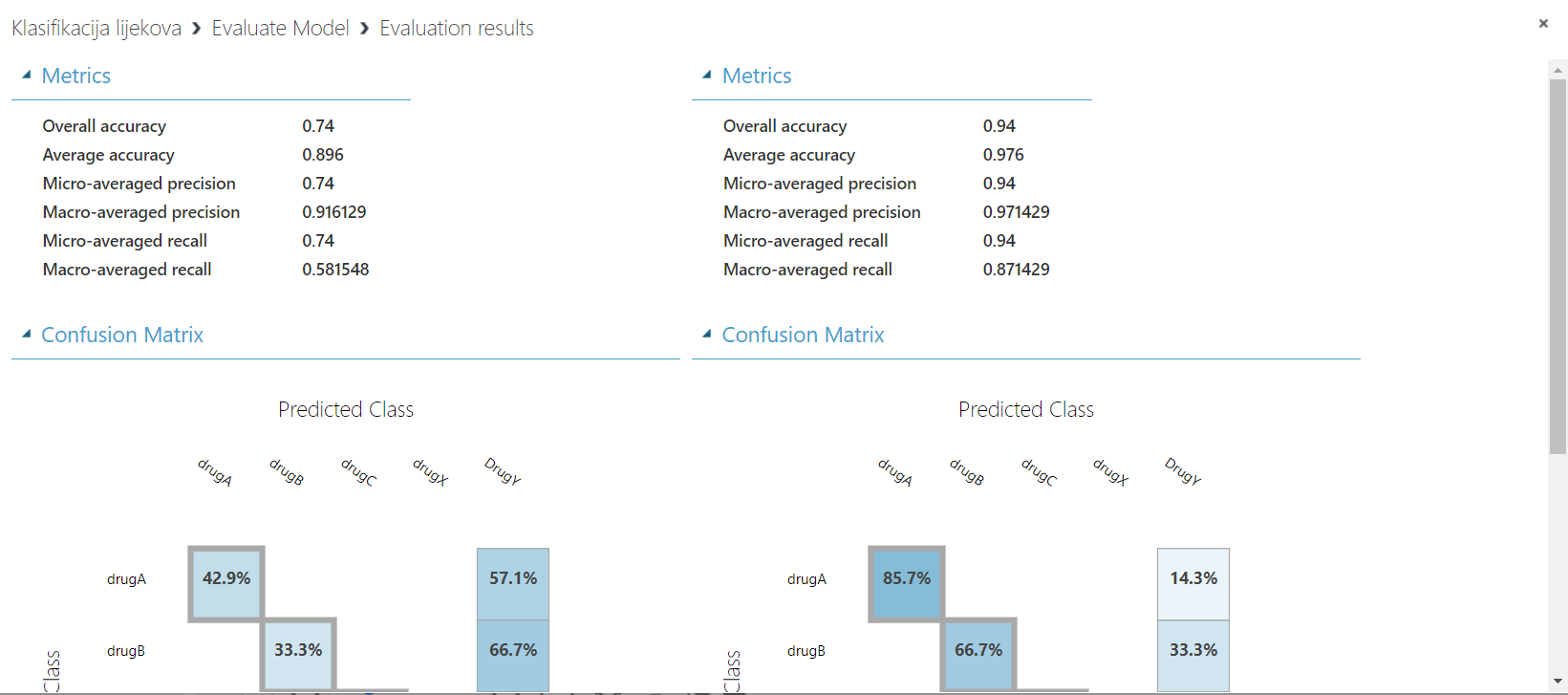


Slika 3.1.1. Model strojnog učenja

Nakon unosa podataka podaci su podijeljeni na 75% za treniranje i 25% za testiranje. Azure Machine Learning Studio podržava različite algoritme za strojno učenje od kojih su izabrani algoritmi višeklasne klasifikacije kao „Multiclass Decision Forest“, „Multiclass Decision Jungle“, „Multiclass Logistic Regression“ i „Multiclass Neural Network“. Rezultati algoritama su uspoređeni u nastavku.

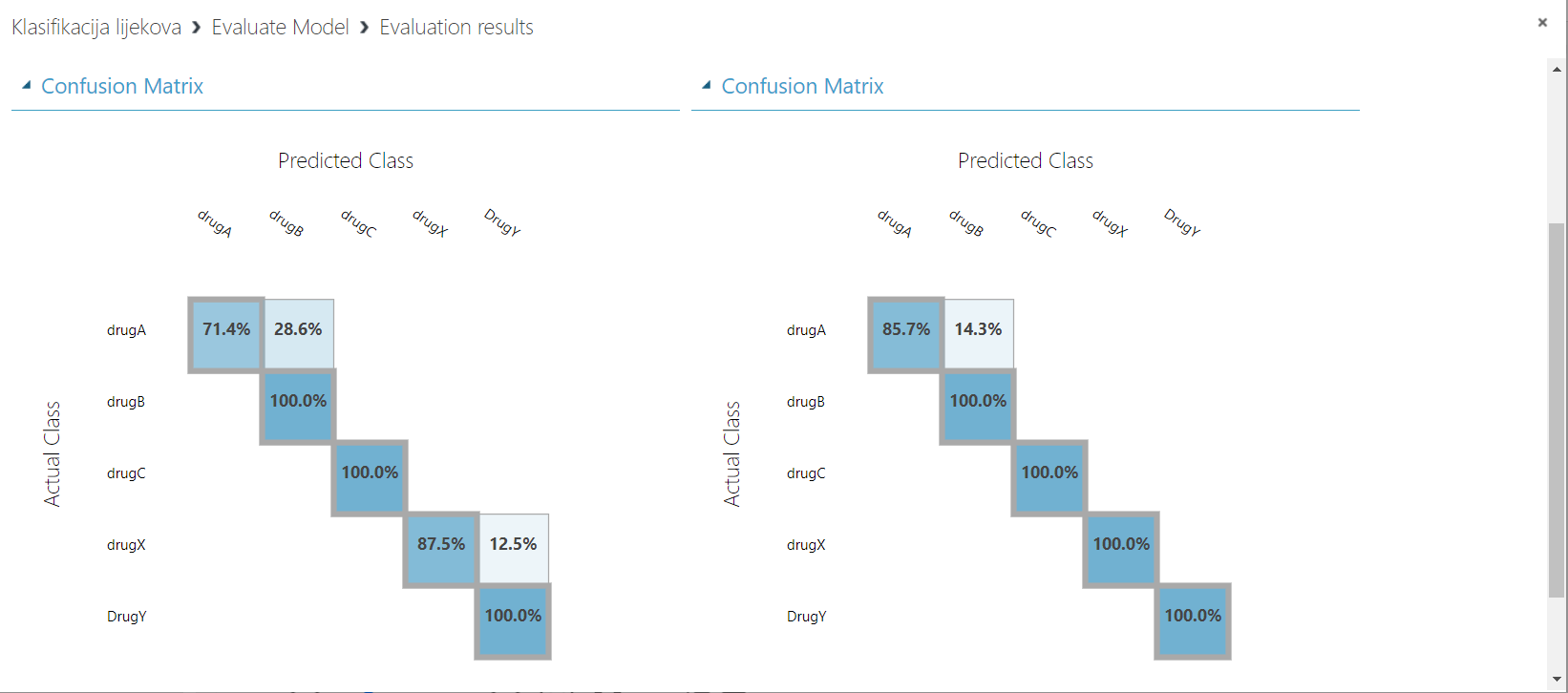
 

Slika 3.1.2. Preciznost za „Multiclass Decision Forest“ i „Multiclass Decision Jungle“ algoritme

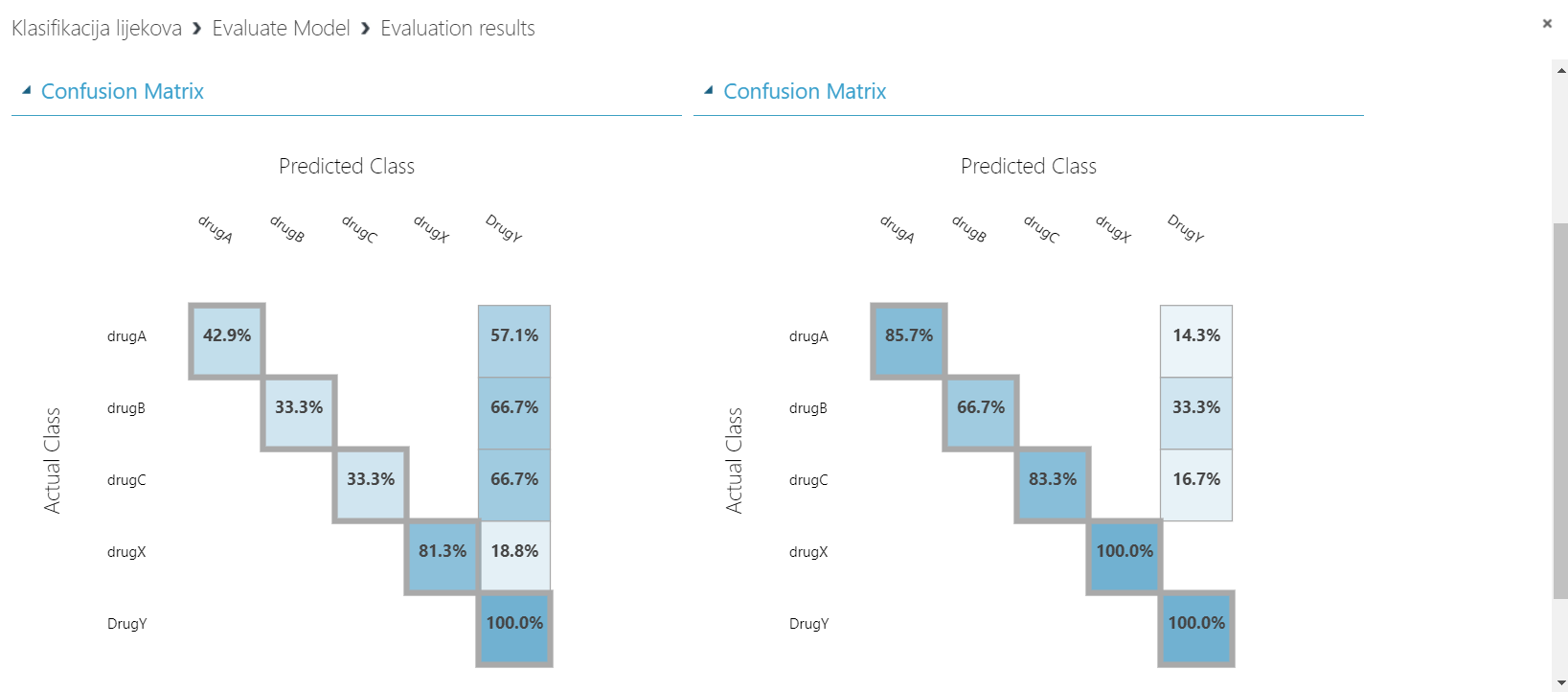
 

Slika 3.1.3. Preciznost za „Multiclass Logistic Regression“ i „Multiclass Neural Network“ algoritme

Uspoređujući podatke na slikama 3.1.2. i 3.1.3. može se zaključiti da algoritam „Multiclass Decision Jungle“ ima najveću preciznost, a algoritam „Multiclass Logistic Regression“ najmanju. Ti se rezultati mogu potvrditi pomoću matrica zbunjenosti na slikama 3.1.4. i 3.1.5.

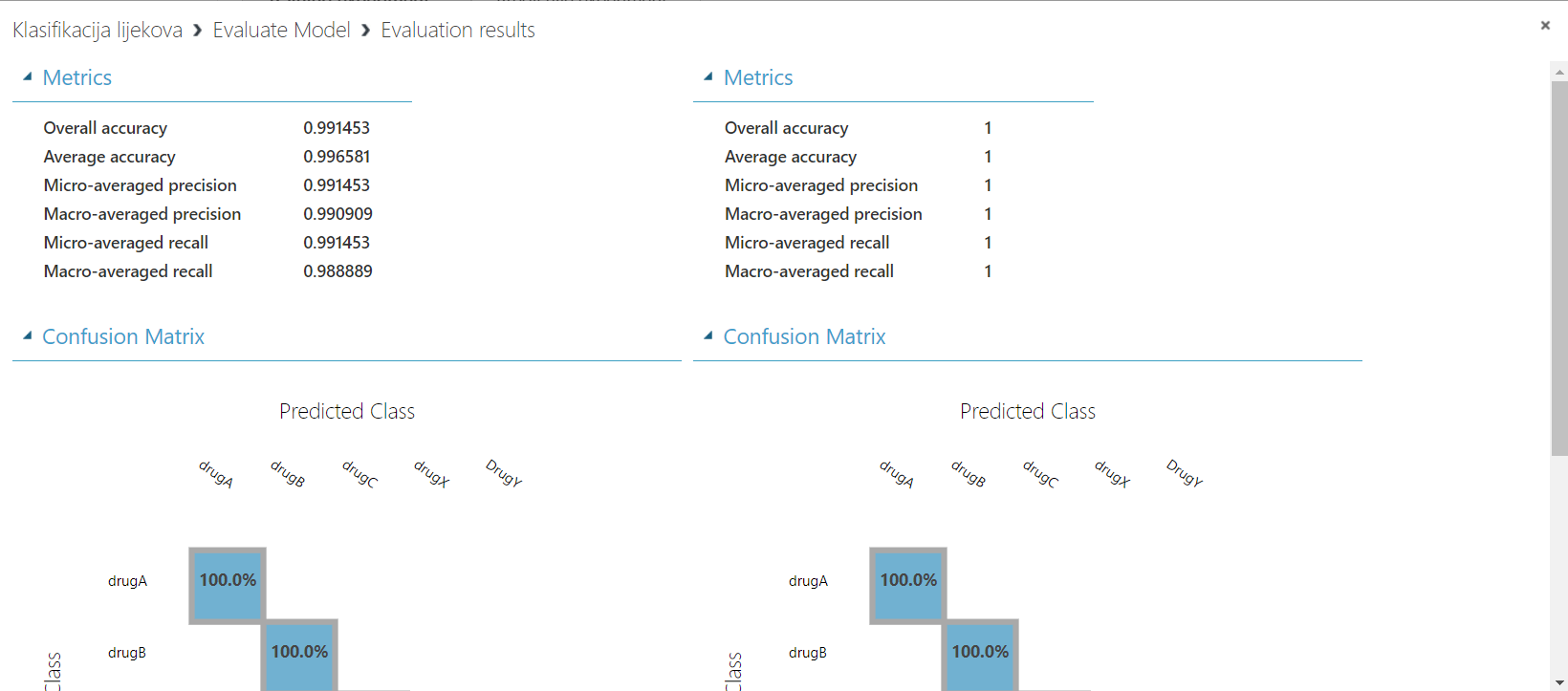
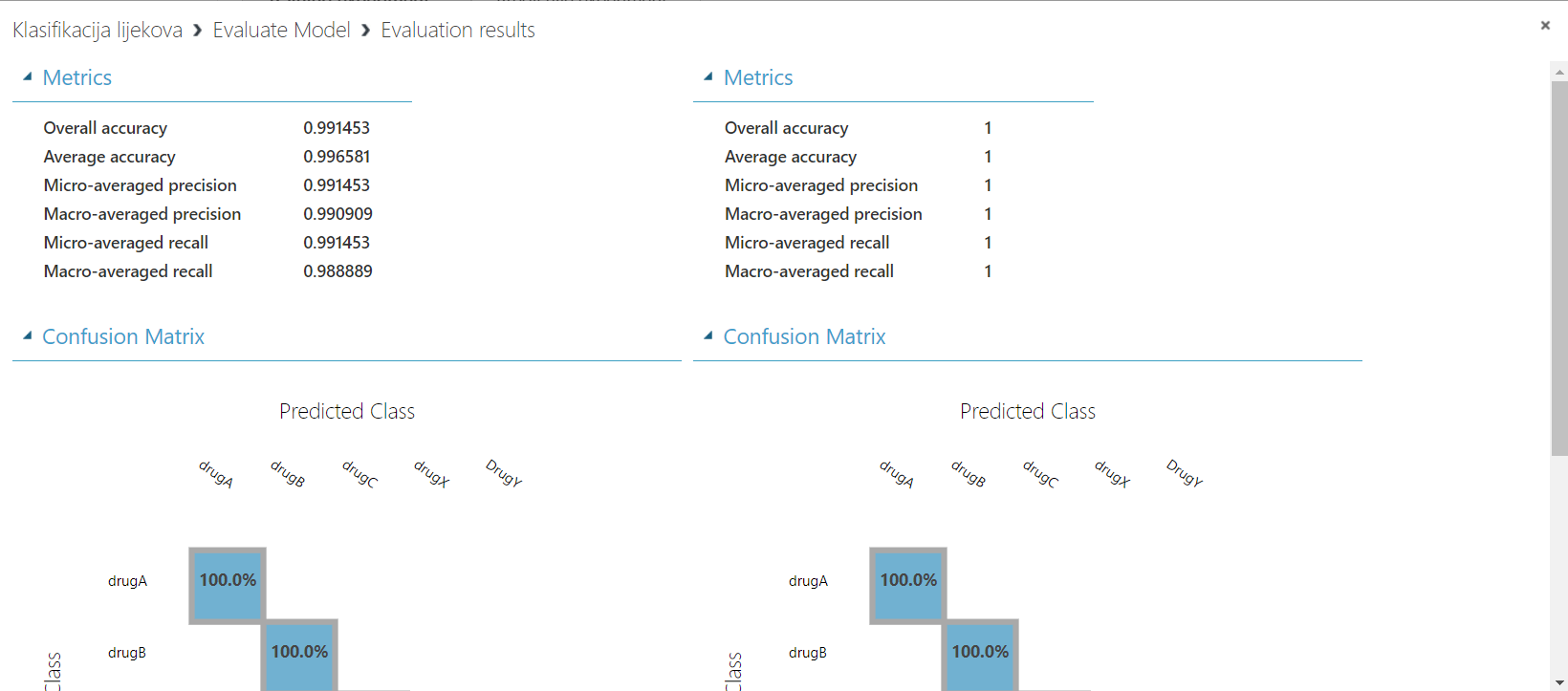


Slika 3.1.4. Matrica zbunjenosti za „Multiclass Decision Forest“ i „Multiclass Decision Jungle“ algoritme

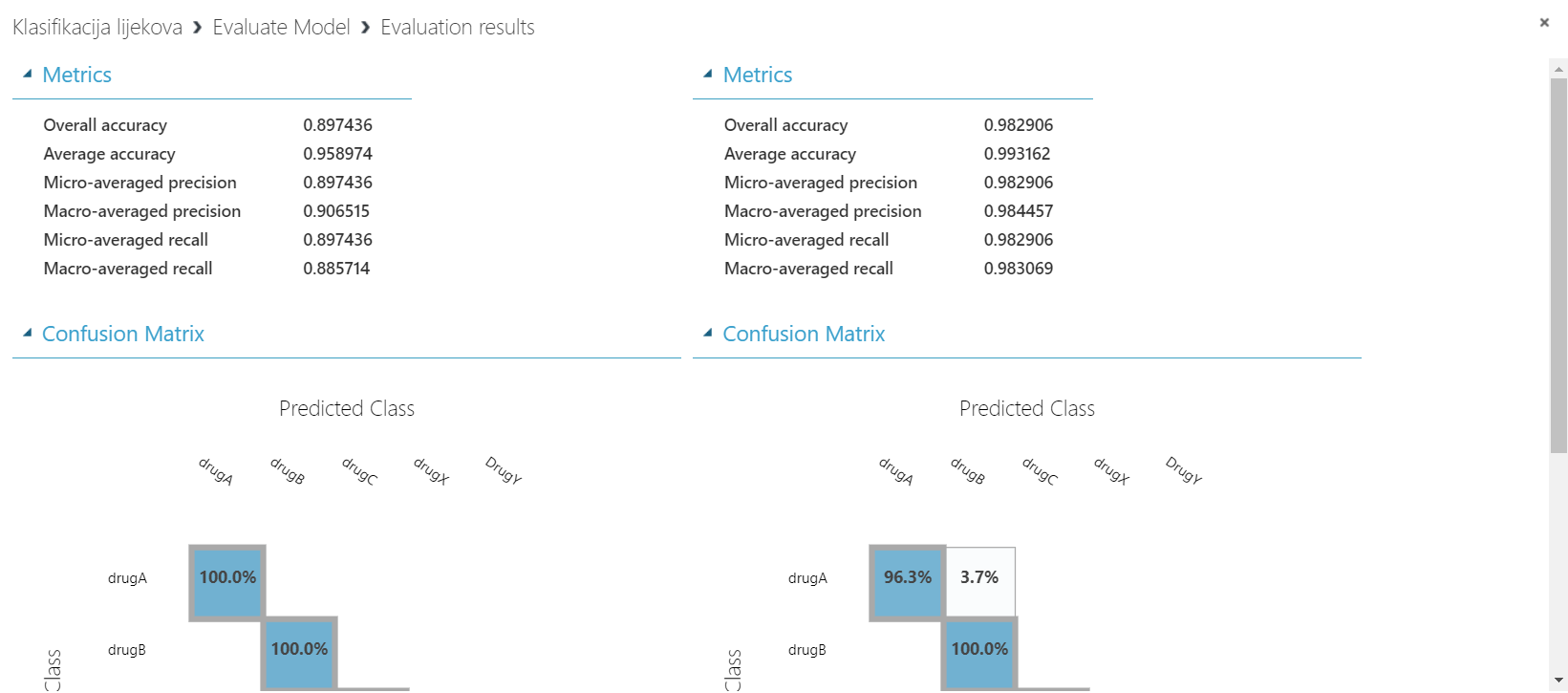
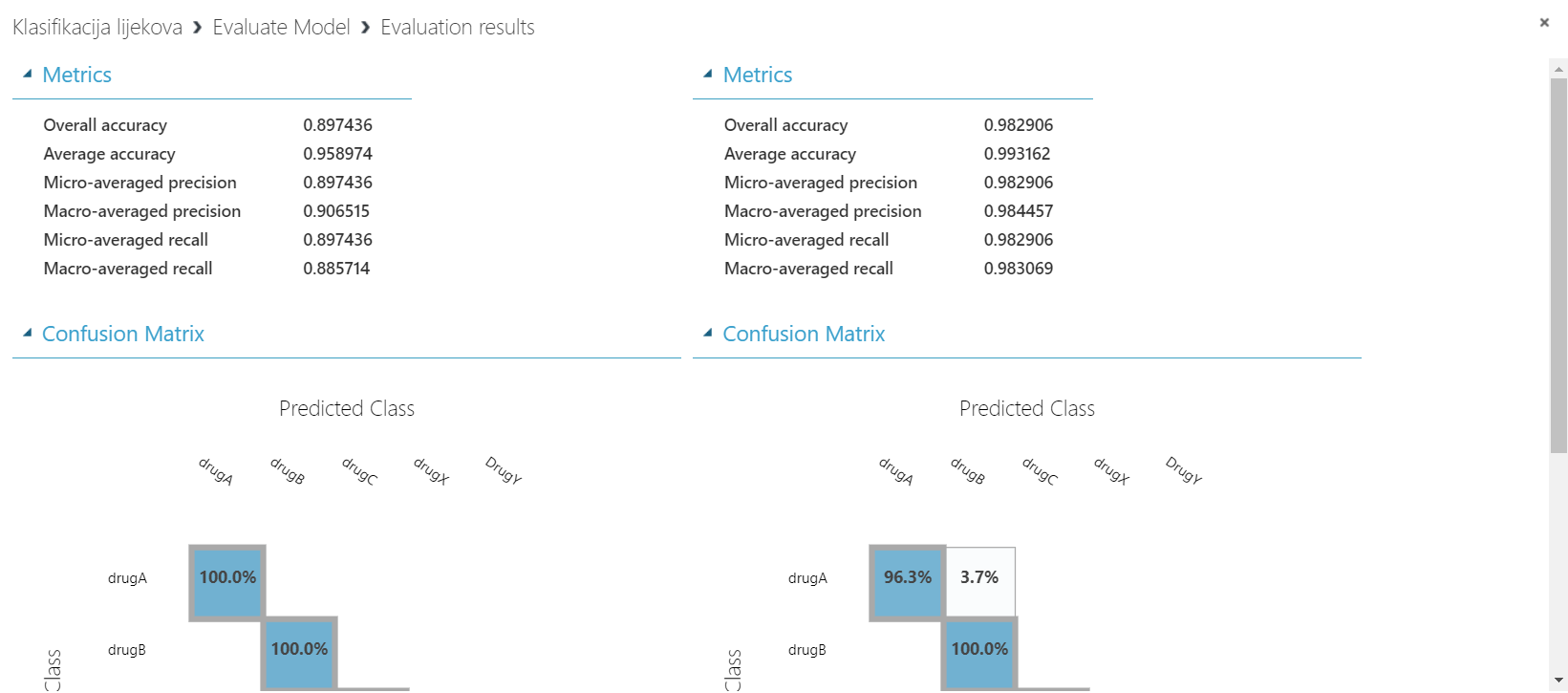


Slika 3.1.5. Matrica zbunjenosti za „Multiclass Logistic Regression“ i „Multiclass Neural Network“ algoritme

Iako je preciznost „Multiclass Decision Jungle“ algoritma velika, obradom podataka se pokušala dobiti još veća preciznost. Za modele s desne strane modela korišten je algoritam SMOTE kako bi distribucija uzoraka klasa vrste lijekova bila uravnotežena što je povećalo preciznost algoritama koja se može vidjeti na slikama 3.1.6. i 3.1.7.

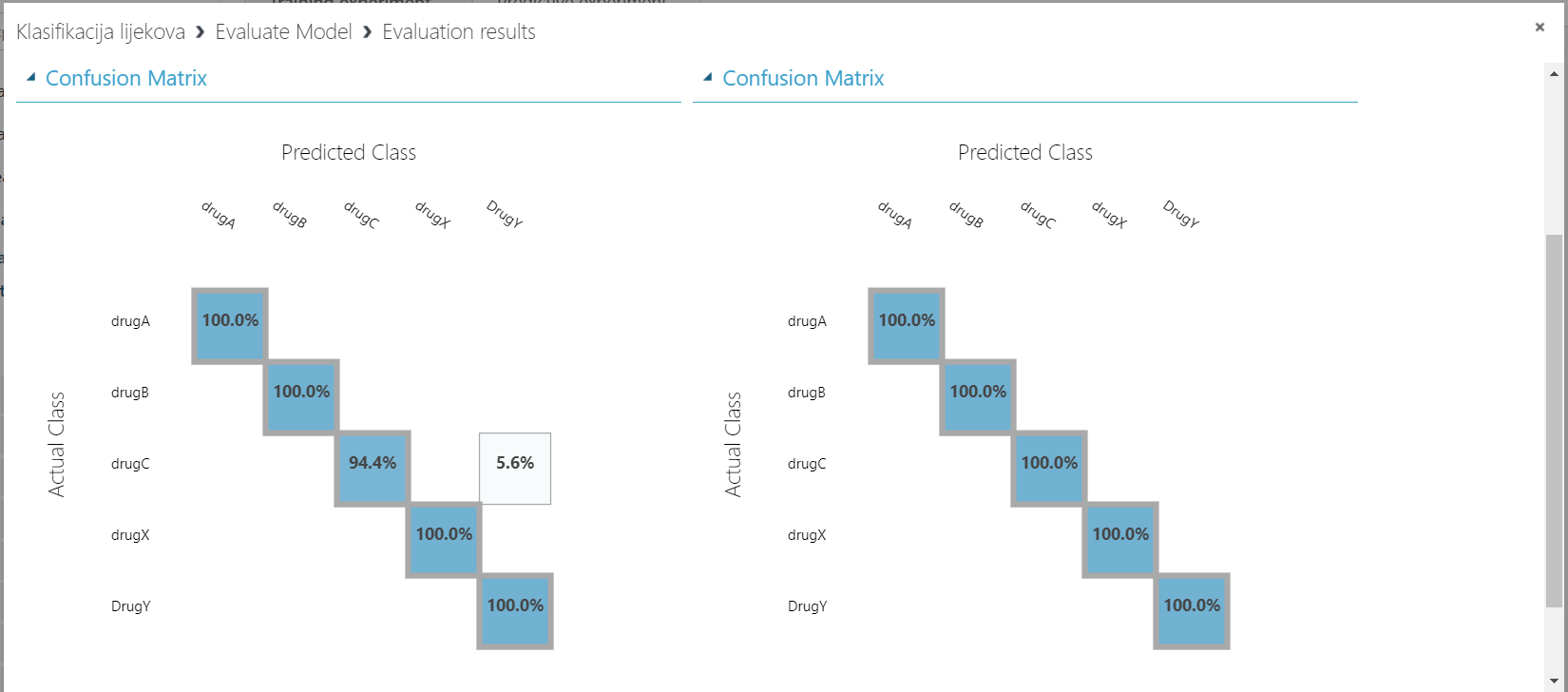
¸ 

Slika 3.1.6. Preciznost za „Multiclass Decision Forest“ i „Multiclass Decision Jungle“ algoritme s uravnoteženim klasama

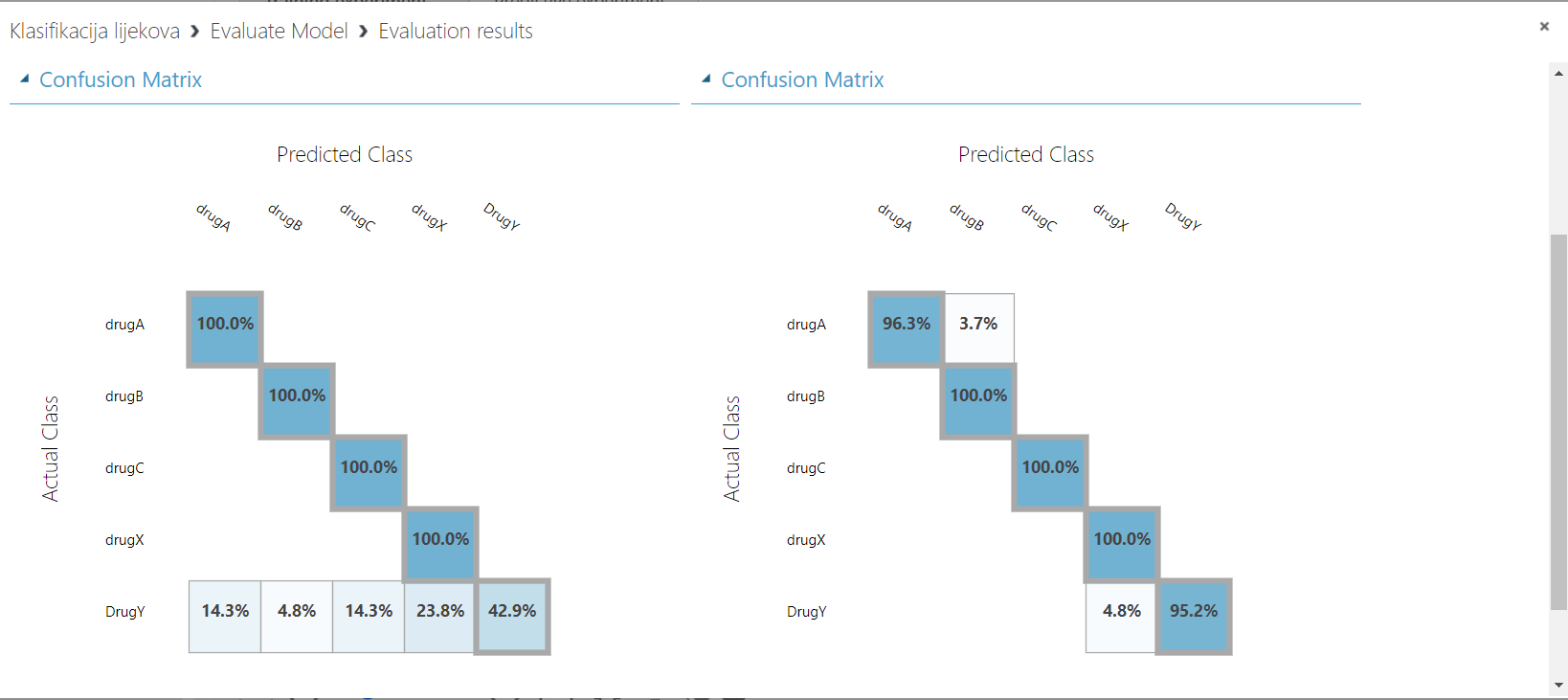
 

Slika 3.1.7. Preciznost za „Multiclass Logistic Regression“ i „Multiclass Neural Network“ algoritme s uravnoteženim klasama

Uspoređujući preciznosti desnih i lijevih modela može se zaključiti da svi desni modeli koji koriste uravnotežene klase, postižu veću preciznost. Najveću preciznost postiže „Multiclass Decision Jungle“ algoritam, a najmanju „Multiclass Logistic Regression“ algoritam, što je identično modelima na lijevoj strani modela. Rezultati se mogu potvrditi matricama zbunjenosti na slikama 3.1.8. i 3.1.9.

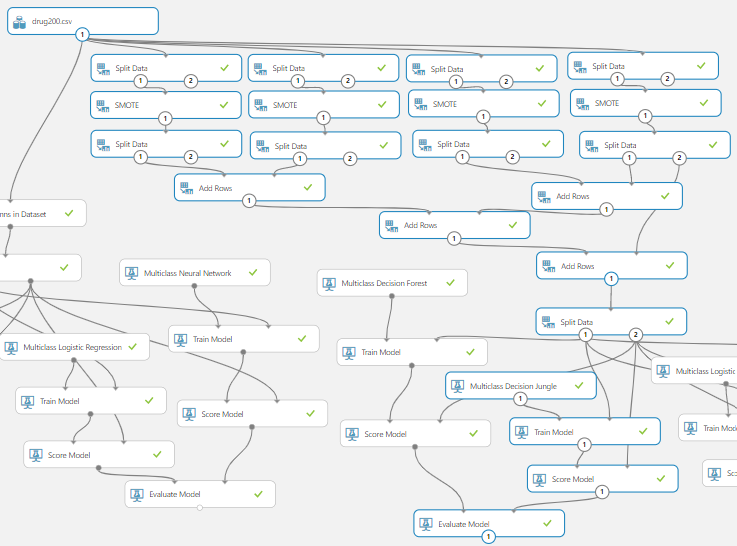


Slika 3.1.8. Matrica zbunjenosti za „Multiclass Decision Forest“ i „Multiclass Decision Jungle“ algoritme s uravnoteženim klasama



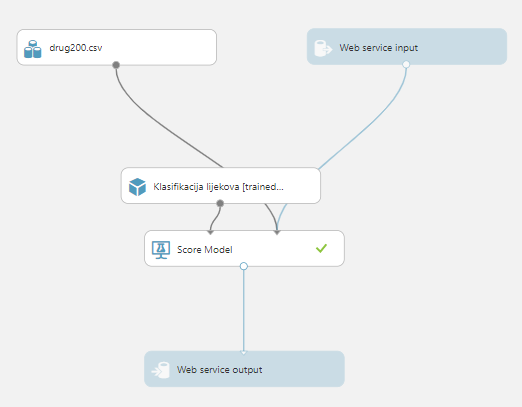
Slika 3.1.9. Matrica zbunjenosti za „Multiclass Logistic Regression“ i „Multiclass Neural Network“ algoritme s uravnoteženim klasama

Uspoređujući dobivene preciznosti i matrice zbunjenosti zaključuje se da je najoptimalniji algoritam za klasifikaciju lijekova za ovaj skup podataka „Multiclass Decision Jungle“ algoritam koji nadograđuje algoritam šume odlučivanja. Algoritam džungle odlučivanja sastoji se od skupa odluka usmjerenih acikličih grafova i dopušta spajanje grana stabala.



Slika 3.1.10. Shema modela odabranog za klasifikaciju lijekova

Model je prvo pretvoren u prediktivni model u kojem su uklonjene SMOTE komponente modela jer ih Azure Machine Learning Studio ne podržava za višepodatkovne ulaze u prediktivnom modelu. Nakon toga model je pretvoren u web servis kojem je Azure Machine Learning Studio dodijelio HTTP pristupne točke za pristup modelu s udaljene lokacije, a u ovom slučaju to je C# desktop aplikacija. Pomoću API-ja omogućeno je slanje podataka prediktivnom modelu te primanje odgovora, odnosno rezultata klasifikacije od modela.



Slika 3.1.11. Prediktivni model

## Klijentska aplikacija

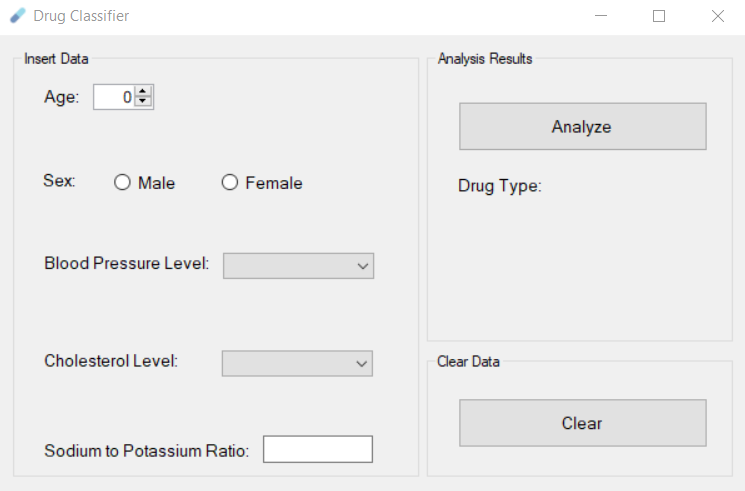
Klijentska desktop aplikacija je razvijena u Visual Studiu u C# jeziku. Glavni cilj aplikacije je prikaz vrste lijeka na temelju parametara određenog pacijenta. Izgled korisničke aplikacije je jednostavan i lako razumljiv kao što se može vidjeti na slici 3.2.1. Korisnik treba unijeti podatke kao što su godine, spol, razinu krvnog tlaka, razinu kolesterola u krvi i omjer natrija i kalija u krvi i pritiskom na tipku „Analyze“ ispod teksta „Drug Type“ dobiva se rezultat klasifikacije, odnosno vrsta lijeka. Kako bi se izvršila klasifikacija nije potrebno unijeti sve podatke, ali se preciznost povećava s količinom unesenih podataka. Klikom na tipku „Clear“ uklanjaju se uneseni podaci i rezultat ako je prije toga pritisnuta tipka „Analyze“. Na taj način korisnik ne mora brisati svaki podatak zasebno prije unosa novih podataka. Za primanje i slanje odgovora s Azure Machine Learning Studia korišten je automatski generirani kod pri objavi web servisa. Promjene u kodu su napravljene za unos i ispis podataka. Pri unosu podataka ne traži se vrijednost vrste lijeka, već se ona ispisuje nakon komunikacije s Azure servisom. Odgovor koji se dobije nalazi se u JSON obliku te kako bi ga rastavili i uzeli samo potrebne podatke, u ovom slučaju samo rezultat klasifikacije, korišten je Newtonsoftov Json framework. Kod za rastavljanje JSON-a se nalazi u nastavku.

var returnedObject = JsonConvert.DeserializeObject<RootObject>(result);

string drugType = returnedObject.Results.output1.value.Values[0][11];

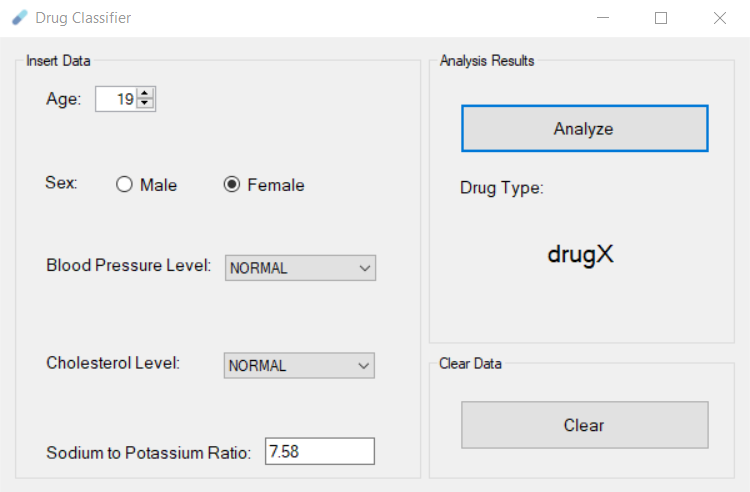
labelDrugType.Text = drugType;

Kod 3.2.1. Rastavljanje JSON formata i ispis vrste lijeka



Slika 3.2.1. Izgled korisničke aplikacije

Unos godina je ograniče na unos cijelog broja, spol na muški i ženski, razina krvnog tlaka na vrijednosti niska, normalna i visoka, a razina kolesterola u krvi na normalnu i visoku. Nakon unosa podataka i pritiska tipke „Analyze“ pokreće se *request* i *response* dio koda za komunikaciju i prikazuje se vrsta lijeka kao što se može vidjeti na slici 3.2.2.



Slika 3.2.2. Rezultat analize i klasifikacije podataka

Funkcionalnost uklanjanja svih podataka nakon pritiska tipke „Clear“ izvedena je prema kodu 3.2.2. Funkcija čišćenja uklanja sve unesene parametre neovisno o broju unesenih parametara, a također uklanja i rezultat analize ako je prije toga izvršena klasifikacija.

private void buttonClear\_Click(object sender, EventArgs e)

{

numericUpDownAge.Value = 0;

radioButtonFemale.Checked = false;

radioButtonMale.Checked = false;

comboBoxBP.SelectedIndex = -1;

comboBoxCholesterol.SelectedIndex = -1;

textBoxNatoK.Clear();

labelDrugType.Text = String.Empty;

}

Kod 3.2.2. Funkcionalnost čišćenja

Za desktop aplikaciju napravljena je ikona u GIMP-u koja se može vidjeti na slici 3.2.3. Ikona predstavlja kapsulu s dvije nijanse plave boje.



Slika 3.2.3. Ikona korisničke desktop aplikacije

# Zaključak

Zbog značajnih posljedica neispravnih odluka u medicini se teži donošenju ispravnih odluka. Tako se i pri rješavanju problema klasifikacije težilo što većoj preciznosti. Problem klasifikacije vrste lijekova je uspješno riješen i postignuta je preciznost od 100%. Korištenjem algoritma džungle odlučivanja za višeklasnu klasifikaciju izrađen je model koji je pretvoren u web servis i pomoću kojega je razvijena korisnička aplikacija. Korisnička desktop aplikacija je razvijena da bude lako razumljiva, jednostavna za korištenje i brza, te olakša liječnicima donošenje odluka i potvrdu njihove odluke u odabiru vrste lijeka za određenog pacijenta. Pri unosu podataka u aplikaciju nije potrebno unijeti sve parametre, ali s povećanjem količine podataka povećava se i točnost rezultata. Korisnička aplikacija zadovoljava potrebne funkcionalnosti, ali bi se mogla nadograditi i proširiti novim funkcionalnostima poput spremanja prethodno unesenih parametara i dobivenih rezultata te njihovog prikaza. Rješavanjem ovoga problema klasifikacije, prošlo se kroz različite algoritme klasifikacije i bolje se upoznalo sa strojnim učenjem.

# Poveznice i literatura

Programskom je rješenju moguće pristupiti preko:

|  |
| --- |
| [Programsko rješenja na GitHubu](https://github.com/Ana-MarijaKatic/Klasifikacija_lijekova) |
| [ML model (predictive)](https://galhttps:/gallery.cortanaintelligence.com/Experiment/Klasifikacija-lijekova-Predictive-Explery.cortanaintelligence.com/Experiment/Klasifikacija-lijekova) |
| [ML model](https://gallery.cortanaintelligence.com/Experiment/Klasifikacija-lijekova) |
| [Korišteni skup podataka drug200](https://www.kaggle.com/prathamtripathi/drug-classification) |