PRAKTIČNA PRIMENA ALGORITAMA MAŠINSKOG UČENJA U OKVIRU PREDMETA ALGORITMI VEŠTAČKE INTELIGENCIJE

Ninoslava Tihi[1] Srđan Popov[2] Miloš Todorov[3] Maja Gaborov[4] Stefan Popović[5]

Rezime: Algoritmi mašinskog učenja (ML) čine temelj savremenih sistema veštačke inteligencije. Njihova primena omogućava automatizaciju analize podataka, identifikaciju obrazaca i donošenje odluka na osnovu podataka. U ovom radu biće demonstrirana praktična primena osnovnih algoritama mašinskog učenja kroz rad sa studentima druge godine master programa – Informacione tehnologije na Visokoj tehničkoj školi strukovnih studija u Novom Sadu, u okviru predmeta Algoritmi veštačke inteligencije. Pored teorijskih osnova, studentima su kroz praktične zadatke predstavljeni algoritmi poput KNN, Naivni Bajes, SVM i Random Forest, zajedno s njihovom implementacijom u programskom jeziku R koristeći pakete caret i randomForest. Rad uključuje poređenje performansi različitih algoritama na stvarnim skupovima podataka, čime se studentima omogućava sveobuhvatno iskustvo u razvoju i evaluaciji modela veštačke inteligencije.

Ključne reči: Veštačka inteligencija, Mašinsko učenje, R, caret, randomForest, klasifikacija, predikcija

PRACTICAL APPLICATION OF MACHINE LEARNING ALGORITHMS IN THE ARTIFICIAL INTELLIGENCE ALGORITHMS COURSE

Abstract: Machine learning (ML) algorithms form the foundation of modern artificial intelligence systems. Their application enables automated data analysis, pattern identification, and data-driven decision-making. This paper demonstrates the practical application of fundamental machine learning algorithms through student work in the second year of the Master's program in Information Technologies at the Higher Technical School of Professional Studies in Novi Sad, within the course *Artificial Intelligence algorithms.* In addition to theoretical foundations, students were introduced to algorithms such as KNN, Naive Bayes, SVM, and Random Forest through hands-on assignments, along with their implementation in the R programming language using the *caret* and *randomForest* packages. The paper includes a comparison of the performance of various algorithms on real-world datasets, providing students with a comprehensive experience in the development and evaluation of artificial intelligence models.

Key words: Artificial intelligence, Machine learning, R, caret, randomForest, classification, prediction

1. UVOD

Veštačka inteligencija (AI) i mašinsko učenje (ML) predstavljaju najdinamičnije oblasti u savremenim informacionim tehnologijama, sa primenom koja obuhvata gotovo sve sektore privrede i društva – od medicine, preko finansija, do obrazovanja i robotike. U oblasti medicine na primer, razvoj algoritama mašinskog učenja, posebno dubokog učenja (Deep Learning), značajno je unapredio analizu medicinskih slika, omogućavajući precizniju dijagnostiku i automatizaciju složenih zadataka u kliničkoj praksi. Dva istaknuta rada u ovoj oblasti ilustruju

[1] Mr, Visoka tehnička škola strukovnih studija u Novom Sadu, Školska 1, Novi Sad, e-mail: tihi@vtsns.edu.rs
[2] Dr, Fakultet tehničkih nauka, Trg Dositeja Obradovića 6, Novi Sad, e-mail: srdjanpopov@uns.ac.rs
[3] Doc. Dr, Fakultet za matematiku i računarske nauke, Alfa BK Univerzitet, Palmira Toljatija 3, Novi Beograd, e- mail: milos.todorov@alfa.edu.rs
[4] MSc, Tehnički fakultet „Mihajlo Pupin“, Univerzitet u Novom Sadu, Đure Đakovića bb, Zrenjanin e-mail: maja.gaborov@tfzr.rs
[5] MSc, Fakultet za matematiku i računarske nauke, Alfa BK Univerzitet, Palmira Toljatija 3, Novi Beograd, e- mail: stefan.popovic@alfa.edu.rs

primenu dubokih konvolutivnih neuronskih mreža u različitim medicinskim kontekstima. Rajpurkar [1] je u svom radu prikazao kako model dubokog učenja može dostići i premašiti performanse radiologa u detekciji upale pluća na rendgenskim snimcima grudnog koša. Koristio je javni skup podataka ChestX-ray14 u kom se nalaze više od 112 000 rendgenskih snimaka grudnog koša od oko 30 000 pacijenata. Dijagnoze su automatski označavane pomoću NLP tehnika (Natural Language Processing) na osnovu radioloških izveštaja.Korišćena je unapred istrenirana DenseNet-121 arhitektura (prethodno trenirana na ImageNet-u) koja je zatim podešena za klasifikaciju rendgenskih snimaka. DenseNet (Densely Connected Convolutionl Network) koristi guste veze između slojeva što omogućava bolji protok informacija i efikasniji rad sa ograničenim podacima Glavni fokus je bio na detekciji pneumonije iako je model sposoban da klasifikuje svih 14 patologija. Model je treniran tako da klasifikuje slike kao pozitivne ili negativne na osnovu prisustva pneumonije. Nakon toga je testiran na posebnom skupu slika i upoređen sa četiri licencirana radiologa. CheXNet je pokazao performanse jednake ili bolje od prosečnog učinka tih radiologa u detekciji pneumonije. Za interpretaciju modela korišćena je metoda Grad-CAM (Gradient-weighted Class Activation Mapping), koja omogućava vizualizaciju oblasti slike koje su najviše uticale na odluku modela. Olaf Ronenber i njegovi saradnici [2] su razvili i testirali U-Net arhitekturu- specijalizovanu konvolutivnu neuronsku mrežu za segmentaciju biomedicinskih slika posebno kada su dostupni ograničeni skupovi podataka. Koristili su ISBI Cell Tracking Challenge skup podataka u kojem su nalaze podaci o mikroskopskim slikama ćelija. Zadatak je bio da se izvrši segmentacija ćelija – tj. precizno izdvajanje kontura ćelija u slikama, piksel po piksel. Svaka slika je imala mape segmentacije (ground truth maske) koje označavaju koji pikseli pripadaju kojoj ćeliji. Model je testiran na više skupova u okviru ISBI izazova, i postigao vrlo visoku tačnost u zadatku segmentacije ćelija. U-Net se pokazao kao izuzetno efikasan i generalizabilan, i danas je osnova za mnoge napredne modele u medicinskoj segmentaciji (npr. 3D U-Net, Attention U-Net, nnU-Net).

Razvoj savremenih AI sistema zasniva se na sofisticiranim algoritmima mašinskog učenja koji omogućavaju sistemima da samostalno uče iz podataka, bez eksplicitnog programiranja pravila ponašanja. Ovi algoritmi omogućavaju identifikaciju obrazaca, donošenje odluka i predviđanje budućih događaja na osnovu istorijskih podataka [3].

Uvođenje ovih koncepata u obrazovni sistem, naročito na visokoškolskom nivou, od suštinskog je značaja za osposobljavanje studenata za rad u savremenom IT okruženju. Integracijom praktičnih primera i konkretnih problema u nastavu, studenti ne samo da usvajaju teorijsko znanje, već stiču i praktične veštine neophodne za samostalnu izradu i evaluaciju inteligentnih sistema.

U okviru predmeta Algoritmi veštačke inteligencije, koji se izvodi na drugoj godini master studija na Visokoj tehničkoj školi strukovnih studija u Novom Sadu, studenti se upoznaju sa osnovnim algoritmima nadgledanog učenja kao što su K najbližih suseda (KNN), Naivni Bajes, SVM (podrška vektor mašina) i Random Forest. Poseban akcenat stavljen je na implementaciju ovih algoritama u programskom jeziku R, uz upotrebu paketa caret i randomForest, koji omogućavaju jednostavan i standardizovan pristup treniranju i evaluaciji modela mašinskog učenja.

Pored savladavanja osnovnih algoritama, studenti uče i o postupcima pripreme podataka, podeli na trening i test skupove, unakrsnoj validaciji i interpretaciji metrika uspešnosti kao što su tačnost, preciznost, odziv i F1 mera. Kroz analizu stvarnih skupova podataka i poređenje performansi različitih algoritama, studenti stiču dubinsko razumevanje praktične primene veštačke inteligencije u rešavanju konkretnih problema.

2. TEORIJSKI OKVIR MAŠINSKOG UČENJA

Mašinsko učenje (ML) je disciplina veštačke inteligencije koja se fokusira na razvoj algoritama i modela koji omogućavaju računarima da uče iz podataka i donose odluke bez

eksplicitnog programiranja [4]. Njegova upotreba postaje sve učestalija u sektorima poput medicine, finansija, bezbednosti, proizvodnje i obrazovanja, gde se obimni podaci mogu iskoristiti za donošenje preciznih i brzih zaključaka.

U osnovi, mašinsko učenje se može klasifikovati u tri glavne kategorije [5]:

1. **Nadzirano učenje** (**Supervised Learning**) - Podrazumeva obučavanje modela na označenim podacima, gde su ulazne vrednosti i odgovarajuće izlazne (ciljne) vrednosti poznate. Cilj modela je da uspostavi relaciju između ulaza i izlaza kako bi mogao da predviđa izlaze za nove, nepoznate podatke. Klasični primeri algoritama nadziranog učenja su:
   - K-najbližih suseda (K-Nearest Neighbours – KNN)
   - Naivni Bajes (Naive Bayes)
   - Mašina vektora potpore (Support Vector Machines – SVM)
   - Nasumične šume (Random Forest)

2. **Nenadzirano učenje** (**Unsupervised Learning)** - Primjenjuje se kada podaci nisu označeni, tj. ne postoje prethodno definirane izlazne vrijednosti. Cilj modela je identifikacija skrivenih obrazaca ili struktura u podacima. Uobičajene metode obuhvataju grupisanje (Clustering) i redukciju dimenzionalnosti, kao što je analiza glavnih komponenti (PCA analiza).

3. **Učenje pojačanjem** (**Reinforcement Learning**) - Usmereno na agente koji stiču sposobnost donošenja optimalnih odluka u dinamičnom okruženju na osnovu sistema nagrađivanja. Ova tehnika se primenjuje u oblastima kao što su robotika, automatizovani sistemi kontrole i razvoj inteligentnih agenata (npr. u video igricama).

Ovaj rad se fokusira na nadzirano učenje, s obzirom na njegovu široku primenu u klasifikacionim i regresionim zadacima, koji su najzastupljeniji u obrazovnim kontekstima. Svaki algoritam primenjen u ovom radu poseduje svoje prednosti i nedostatke, što studentima omogućava da kroz eksperimentisanje i komparaciju steknu dublje razumevanje koncepata.

Poseban značaj pridaje se evaluaciji modela, koja se realizuje korišćenjem metrika kao što su tačnost (accuracy), preciznost (precision), odziv (recall) i F1 mera. Osim toga, ključna komponenta nastave je razumevanje procesa deljenja podataka na trening i test skupove (Data partitioning), kao i unakrsna validacija (Cross validation), koja omogućava objektivniju evaluaciju performansi modela [6,7].

Primena ovih teorijskih osnova kroz praktične zadatke u R programskom jeziku dodatno osnažuje sposobnost studenata da modeluju i rešavaju stvarne probleme korišćenjem alata koji su široko primenjeni u praksi i istraživanju.

3. PRAKTIČAN RAD SA STUDENTIMA

3.1. Radno okruženje

U okviru predmeta Algoritmi veštačke inteligencije, praktičan rad sa studentima osmišljen je tako da obuhvati sve korake u razvoju jednog prediktivnog modela, počev od obrade podataka, preko izbora algoritma, do evaluacije performansi. Nastava se odvija u računarskom laboratorijskom okruženju, gde studenti koriste programski jezik R (verzija 4.3.3) uz razvojno okruženje kao što je RStudio (verzija 2024.12.1+ 563) zajedno sa bibliotekama kao što su caret, ggplot2, itd. Pomoću njih, studenti analiziraju javno dostupne skupove podataka i primenjuju različite algoritme.

Kroz praktične zadatke, studenti imaju priliku da primene sledeće korake u rešavanju problema klasifikacije koristeći već ugrađen skup podataka kao što je Iris.

Svi zadaci su osmišljeni tako da ohrabre eksperimentisanje i samostalno rešavanje problema. Studentima se sugeriše da promene hiperparametre modela (npr. broj stabala u Random Forest-u ili broj suseda u KNN-u), i da koriste unakrsnu validaciju kako bi dodatno poboljšali tačnost modela.

3.2. Klasifikacioni problemi

U ovom delu je dato nekoliko praktičnih zadataka u R jeziku koji se bave učitavanjem skupa podataka, podelom podataka na trening i test skupove, treniranjem različitih klasifikacionih modela i na kraju evaluacijom performansi i vizualizacijom rezultata [8-10].

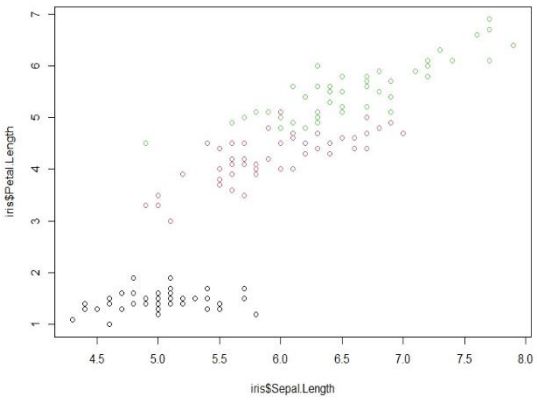
Zadatak 1: Učitavanje i osnovna analiza podataka

Studenti treba da nauče kako da učitaju skup podataka koristeći funkcije read.csv() i data() iz base i datasets paketa, da izvrše osnovnu analizu i vizualizaciju strukture skupa.

```
> # Učitavanje Iris skupa podataka
> data("iris")
>
> # Pregled skupa podataka Iris u tabličnom formatu
> View(iris)
>
> # Prikaz statističkih informacija o Iris skupu podataka
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa     :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
>
>
>
>
> # Kreiranje raspršenog dijagrama (scatterplot) između dve numeričke promenljive iz Iris skupa podataka
> plot(iris$Sepal.Length, iris$Petal.Length, col = iris$Species)
```

Slika 1. R kod za učitavanje i osnovnu analizu Iris skupa podataka

Pomoću funkcije plot() dobija se grafički prikaz dve promenljive iz skupa – Sepal.Length i Petal.Length. Prva promenljiva predstavlja dužinu čašičnog lista (sepal length) i njene vrednosti su nacrtane na x-osi dok druga predstavlja dužinu latastog lista (petal length) koje su nacrtane na y-osi. Boja tačaka zavisi od vrste cveta a za svaku vrstu se koristi različita boja. Pomoću funkcije View() se podaci iz skupa mogu prikazati u tabelarnom formatu. Na slici 2a je prikazan dijagram raštrkanosti za dve promenljive (Sepal Length i Petal length) iz Iris skupa podataka a na slici 2b je prikazan deo Iris skupa podataka u tabelarnom formatu.



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa

Slika 2a) Dijagram raštrkanosti (scatterplot) za dve promenljive iz Iris skupa, b) Tabelarni prikaz jednog dela Iris skupa podataka

Zadatak 2: Podela podataka na trening i test skup

Jedan od prvih koraka prilikom modeliranja je particionisanje podataka na trening i test skup. Trening skup se koristi za građenje modela a test za validaciju modela. Studenti koriste funkciju *createdataPartition()* iz caret paketa kako bi izvršili particionisanje.

```r
# učitavanje biblioteke caret
library(caret)

# Postavljanje početne vrednosti za generator slučajnih brojeva
set.seed(123)

# Kreiranje indeksa redova koji će se koristiti za trening skup,
# Podela podataka (70% trening, 30% test)

index <- createDataPartition(iris$Species, p = 0.7, list = FALSE)

# Kreiranje trening skupa
train_data <- iris[index, ]
# Kreiranje test skupa
test_data <- iris[-index, ]
```

Slika 3. R kod za particionisanje Iris skupa podataka

Na slici 3 je prikazan R kod za particionisanje Iris skupa podataka. Nakon učitavanja biblioteke *caret* potrebno je postaviti početnu vrednost za generator slučajnih brojeva u R-u. To u stvari znači da će svi „slučajni“ rezultati biti isti svaki put kada se kod pokrene jer se koristi isti „*seed*“ broj. Na ovaj način se dobija reproduktivnost rezultata a bez *set.seed()* funkcije rezultati bi se razlikovali svaki put. To posebno može biti korisno kada je u istraživanjima i testiranju modela važno da drugi mogu ponoviti eksperiment i dobiti iste rezultate.

Zadatak 3: Treniranje klasifikacionih modela

Studentima su predstavljeni osnovni algoritmi mašinskog učenja kao što su KNN, Naivni Bajes, SVM i Random Forest i svi modeli se treniraju pomoću funkcije *train()* iz paketa *caret*. Na slici 4 je prikazan R kod za instaliranje potrebnih biblioteka i njihovo učitavanje a nakon toga se pravi kontrolni objekat za funkciju train(9 kojom se definiše kako se trenira model, tj. kako se radi validacija tokom treniranja. Konkretno, u ovom datom primeru se koristi metoda unakrsne validacije (cross-validation) što se definiše postavljanjem parametra method da ima vrednost „cv“ a parametar number da bude 5 što znači da se ceo trening skup deli na 5 podskupova. Podaci se u stvari dele na 5 jednakih delova a model se trenira 5 puta i svaki put se koriste 4 dela za treniranje a 1 deo za validaciju. Nakon podešavanja parametara za unakrsnu validaciju, grade se KNN, Naive Bayes, Random Forest i SVM modeli.

```r
#instaliraj potrebne biblioteke
install.packages(c("caret", "e1071", "randomForest", "klaR"))

# učitaj biblioteke
library(caret)
library(e1071)        # za SVM i Naive Bayes
library(randomForest)
library(klaR)         # dodatna podrška za Naive Bayes

# Postavi kontrolu treniranja
control <- trainControl(method = "cv", number = 5)

# 1. Gradenje modela k-Nearest Neighbors (kNN)
model_knn <- train(Species ~ ., data = train_data, method = "knn", trControl = control)


# 2. Gradenje Naive Bayes modela
model_nb <- train(Species ~ ., data = train_data, method = "nb", trControl = control)


# 3. Gradenje Random Forest modela
model_rf <- train(Species ~ ., data = train_data, method = "rf", trControl = control)


# 4. Gradenje Support Vector Machine modela (SVM, radial kernel)
model_svm <- train(Species ~ ., data = train_data, method = "svmRadial", trControl = control)
```

Slika 4. R kod za treniranje klasifikacionih modela

Zadatak 4: Evaluacija modela

Nakon izgradnje modela, vrši se predikcija a zatim se prikazuje matrica konfuzije (confusion Matrix) koja vrši poređenje predikcija i stvarnih vrednosti. Na kraju studenti vrše evaluaciju upoređujući učinak svakog modela i kroz uporedne rezultate diskutuju o tačnosti, preciznosti, odzivu i F1 meri za svaki algoritam. Na slici 5 je prikazan R kod za evaluaciju svih izgrađenih modela.

```r
# Predikcija i evaluacija KNN modela
pred_knn <- predict(model_knn, test_data)
cat("KNN:\n")
print(confusionMatrix(pred_knn, test_data$Species))

# Predikcija i evaluacija Naive Bayes modela
pred_nb <- predict(model_nb, test_data)
cat("\nNaive Bayes:\n")
print(confusionMatrix(pred_nb, test_data$Species))

# Predikcija i evaluacija Random Forest modela
pred_rf <- predict(model_rf, test_data)
cat("\nRandom Forest:\n")
print(confusionMatrix(pred_rf, test_data$Species))

# Predikcija i evaluacija SVM modela
pred_svm <- predict(model_svm, test_data)
cat("\nSVM:\n")
print(confusionMatrix(pred_svm, test_data$Species))
```

Slika 5. R kod za evaluaciju modela

Na osnovu koda sa prethodne slike dobijeni su rezultati modela koji su prikazani u tabeli 1. Poređenjem rezultata više modela na istom skupu podataka, studenti su razvili razumevanje o tome kako odabrati odgovarajući model.

Algoritam	Tačnost	Preciznost	Odziv	F1 mera
KNN	0.9556	0.9555	0.9555	0.9555
Naive Bayes	0.8889	0.8899	0.8889	0.8899
SVM	0.9111	0.9111	0.9111	0.9111
Random Forest	0.8889	0.8899	0.8889	0.8888

Tabela 1 – Rezultati za sva četiri modela

Evaluacija KNN modela: Iz tabele se može videti da je tačnost za KNN model 0.9556 što znači da je model tačan u 95.56% slučajeva. Na osnovu tačnosti modela može se zaključiti da je većina predikcija bila pravilna. Ovaj model ima takođe i veoma veliku preciznost od 0.9555 što znači da kada je model predvideo neku klasu ta predikcija je u 95.55% bila ispravna. Na osnovu odziva od 95.55%, ovaj model je pokazao da je veoma dobar u pronalaženju svih pravih pozitivnih uzoraka što potvrđuje da je uspešno detektovao većinu stvarnih pozitivnih klasa. F1 skor od 0.9555 je odličan, jer predstavlja harmonijsku sredinu između preciznosti i odziva. Može se zaključiti da je KNN odličan model sa visokom tačnošću i izuzetno dobrim rezultatima u preciznosti, odzivu i F1 meri.

Evaluacija Naive Bayes modela: Naive Bayes ima nešto nižu tačnost od 88.89% u odnosu na KNN model, što znači da je skloniji pravljenju grešaka ali se može reći da je i dalje veoma dobar. Preciznost ovog modela od 0.8899 znači da kada je model predvideo neku klasu, 88,99% je to bila tačna predikcija. Što se tiče odziva, ovaj model je propustio neki broj pozitivnih uzoraka jer odziv iznosi 88,89%. F1 skor je takođe visok (88,99%) što znači da je model izbalansiran u pogledu preciznosti i odziva. Zaključak je da je ovaj model vrlo dobar, ali svakako ima prostora za poboljšanje u odnosu na KNN model.

Evaluacija SVM modela: SVM model ima tačnost od 91,11% što je niže od KNN ali je više od Naive Bayes modela. Model je uspeo da predvidi klase u 91,11% slučajeva. SVM model je prilično efikasan u detekciji stvarnih pozitivnih uzoraka sa odzivom od 91,11%. a F1 skor od takođe 91,11% je dobar ali je ipak manji od KNN što pokazuje da nije u potpunosti uravnotežen kao KNN. Može se zaključiti da ovaj model ima solidnu tačnost i dobar F1 skor u poređenju sa

KNN i Naive Bayes-om ali ima malo niže rezultate što znači da ni trebalo da se testira sa različitim parametrima kako bi se poboljšala preciznost i odziv.

Evaluacija Random Forest modela: Random Forest model ima tačnost od 0.8889 kao i Naive Bayes model. U 88,99% slučajeva je uspeo da predvidi klasu a odziv mu je takođe 88,89% što potvrđuje da nije baš dobar u pronalaženju stvarnih pozitivnih uzoraka. F1 skor od 88,88% znači da ovaj model nije baš uravnotežen. Zaključak je da je Random Forest dobar model i njegova tačnost i F1 skor su veoma solidni ali nije bolji od KNN i SVM modela u preciznosti i odzivu.

Na osnovu evaluacije svih modela generalni zaključak je da KNN najbolji model jer je postigao najbolje rezultate sa najvišom tačnošću, preciznošću, odzivom i F1 skorom. Odličan je u prepoznavanju svih klasa, posebno Setosa klase. Naive Bayes i Random Forest su postigli slične rezultate iako nisu imali visok F1 skor kao KNN i SMV modeli. Random Forest je takođe dao solidne rezultate ali je njegov F1 skor za klase Versicolor i Virginica bio niži nego kod drugih modela.

Zadatak 5: Vizualizacija rezultata

Kao završni deo, studenti dobijaju zadatak da vizualizuju rezultate korišćenjem ggplot2, lattice ili drugih ugrađenih plotting funkcija kako bi grafički prikazali učinak modela. Na slici 6 je prikazan R kod za vizualizaciju komparacije modela po tačnosti.

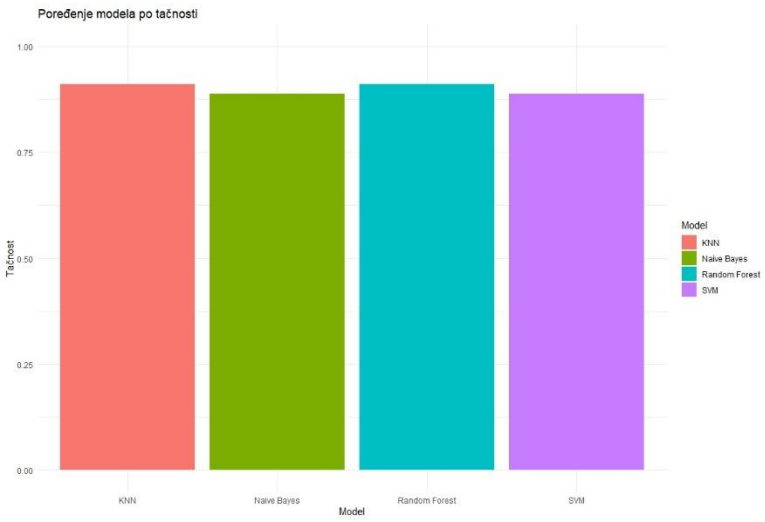
```
# Kompajliraj tačnost svih modela
results <- data.frame(
  Model = c("KNN","Naive Bayes", "Random Forest", "SVM"),
  Accuracy = c(
    knn_conf_matrix$overall["Accuracy"],
    nb_conf_matrix$overall["Accuracy"],
    svm_conf_matrix$overall["Accuracy"],
    rf_conf_matrix$overall["Accuracy"]

  )
)

# vizualizuj komparaciju modela po tačnosti
ggplot(results, aes(x = Model, y = Accuracy, fill = Model)) +
  geom_bar(stat = "identity") +
  ylim(0, 1) +
  labs(
    title = "Poređenje modela po tačnosti",
    y = "Tačnost",
    x = "Model"
  ) +
  theme_minimal()
```

Slika 6. R kod za vizualizaciju rezultata

Nakon pokretanja koda sa slike 6, dobija se grafik poređenja svih modela po tačnosti koji je prikazan na slici 7.



Slika 7. Grafik poređenja svih modela po tačnosti

4. ZAKLJUČAK

U ovom radu, kroz rad sa realnim podacima i primenu osnovnih algoritama mašinskog učenja u okviru predmeta „Algoritmi veštačke inteligencije“, studenti su stekli praktična znanja i iskustva iz oblasti mašinskog učenja i temeljno razumevanje ključnih tehnika iz oblasti veštačke inteligencije. Fokusiranjem na popularne algoritme poput KNN, Naivnog Bajesa, SVM-a i Random Forest-a, studentima je omogućeno da ne samo nauče teoriju, već i da kroz praktične zadatke razvijaju veštine implementacije i evaluacije modela. Koristeći programski jezik R i pakete kao što su caret i randomForest, studenti su imali priliku da primene naučene algoritme na stvarnim skupovima podataka i dobiju dublje uvid u njihove prednosti i ograničenja u različitim kontekstima. Poređenjem performansi različitih algoritama, studenti su se upoznali sa važnostima odabira odgovarajućih metoda u zavisnosti od specifičnih karakteristika podataka i ciljeva predikcije ili klasifikacije. Ovaj praktični pristup im omogućava da se pripreme za rad u industriji, gde su sposobnost implementacije i evaluacije modela, kao i razumevanje kako različiti algoritmi funkcionišu u stvarnim situacijama, ključni za rešavanje složenih problema.

Dalja istraživanja mogu se fokusirati na unapređivanje trenutnih metoda, istraživanje hibridnih pristupa ili proširivanje korišćenja naprednijih tehnika, kao što su duboko učenje, za još preciznije i robusnije modele u raznim industrijskim aplikacijama. Integracija sa velikim podacima, poboljšanje brzine treniranja modela i interpretabilnost rezultata ostaju izazovi koji će oblikovati buduće primene mašinskog učenja u oblasti veštačke inteligencije.

Ovaj rad pruža studentima solidnu osnovu za dalje istraživanje i primenu mašinskog učenja u realnim uslovima, čime doprinosi njihovom razvoju kao stručnjaka u oblasti veštačke inteligencije a ovako koncipirana nastava doprinosi njihovoj boljoj pripremljenosti za izazove iz realnog sveta i omogućava lakšu primenu teorijskih znanja u praksi.

5. LITERATURA

[1] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., & Ng, A. Y. (2017*). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. ArXiv*. DOI: https://doi.org/10.48550/arXiv.1711.05225

[2] Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. DOI: https://doi.org/10.48550/arXiv.1505.04597

[3] Kuhn, M., & Johnson, K.: *Applied Predictive Modeling*. Springer. New York, 2013, ISBN: 978-1-4614-6849-3.

[4] James, G., Witten, D., Hastie, T., & Tibshirani, R.: *An Introduction to Statistical Learning: with Applications in R*, Springer, 2013.

[5] Lantz, B.: *Machine Learning with R: Expert techniques for predictive modeling*, Third edition. Packt Publishing, 2019, ISBN-13: 9781788295864.

[6] Torgo, L.: *Data Mining with R: Learning with Case Studies*, Second Edition. Chapman and Hall/CRC, 2017, ISBN-13: 978-1482234893.

[7] Wickham, H., Cetinkaya-Rundel, M., & Grolemund, G.: *R for Data Science*, Second edition. O'Reilly Media, 2023, ISBN-13: 978-1492097402

[8] https://www.r-bloggers.com/2015/03/machine-learning-in-r-for-beginners/, April, 2025.

[9] https://www.r-bloggers.com/2023/08/visualizing-categorical-data-in-r-a-guide-with-engaging-charts-using-the-iris-dataset/, April 2025.

[10] „Iris Data Set“ – UCI Machine Learning Repository https://archive.ics.uci.edu/ml/datasets/iris, April 2025.