SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1983

MODELI PROCJENE RIZIKA ZA KONZUMACIJU PSIHOAKTIVNIH SUPSTANCI TEMELJENI NA OSOBINAMA LIČNOSTI I METODAMA STROJNOG UČENJA

Jana Gazdek

Zagreb, lipanj 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1983

MODELI PROCJENE RIZIKA ZA KONZUMACIJU PSIHOAKTIVNIH SUPSTANCI TEMELJENI NA OSOBINAMA LIČNOSTI I METODAMA STROJNOG UČENJA

Jana Gazdek

Zagreb, lipanj 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 3. ožujka 2025.

ZAVRŠNI ZADATAK br. 1983

Pristupnica:	**Jana Gazdek (0036537526)**
Studij:	Elektrotehnika i informacijska tehnologija i Računarstvo
Modul:	Računarstvo
Mentor:	izv. prof. dr. sc. Alan Jović

Zadatak: **Modeli procjene rizika za konzumaciju psihoaktivnih supstanci temeljeni na osobinama ličnosti i metodama strojnog učenja**

Opis zadatka:

Problem procjene rizika pojedinca za konzumaciju i zlouporabu psihoaktivnih supstanci (droga) vrlo je važan zbog značajnih posljedica na okolinu. Iz literature je poznato da postoje razni čimbenici povezani s početnom uporabom psihoaktivnih supstanci, uključujući psihološke, socijalne, okolišne, ekonomske i individualne. Ovi čimbenici su također povezani s nizom osobina ličnosti, koje se mogu u značajnoj mjeri odrediti korištenjem odgovarajućih psiholoških testova. Cilj ovog završnog rada je izrada više modela rizika za konzumaciju psihoaktivnih supstanci temeljenih na osobinama ličnosti i metodama strojnog učenja. Kao skup za učenje modela rizika potrebno je upotrijebiti relevantni i slobodno dostupni skup podataka https://archive.ics.uci.edu/dataset/373/drug+consumption+quantified. U radu je potrebno teorijski opisati metode strojnog učenja koje se mogu upotrijebiti za izradu visokotočnih modela rizika (npr. stroj s potpornim vektorima, slučajna šuma, XGBoost) kao i metode za izgradnju interpretabilnih metoda rizika (npr. stablo odluke, induktivna pravila). Nakon toga, potrebno je pripremiti podatke te naučiti modele rizika za svaku vrstu psihoaktivne supstance dostupne u skupu podataka. Modele je potrebno međusobno usporediti na izdvojenom testnom skupu u smislu evaluacijskih mjera strojnog učenja. Komentirati rezultate. Implementaciju je potrebno napraviti u programskom jeziku po vlastitom izboru, a za strojno učenje modela može se u slučaju nedostatka vlastitih sklopovskih resursa koristiti dostupna web rješenja (npr. Google Colab).

Rok za predaju rada: 23. lipnja 2025.

Zahvaljujem mentoru, izv. prof. dr. sc. Alanu Joviću, na pomoći pri izradi završnog rada

Sadržaj

1. Uvod

Upotreba psihoaktivnih supstanci često je povezana s nizom čimbenika koji utječu na ponašanje i odluke pojedinca. Iako su socijalni i demografski uvjeti važni za razumijevanje ovih obrazaca, utjecaj osobina ličnosti sve se više prepoznaje kao važan faktor u modeliranju rizika za razvoj ovisnosti. Razumijevanje utjecaja različitih psiholoških karakteristika na sklonost konzumaciji psihoaktivnih tvari omogućuje precizniju identifikaciju rizičnih skupina te pravovremeno provođenje preventivnih mjera [1].

Cilj ovog rada je istražiti može li se na temelju osobina ličnosti pojedinca procijeniti sklonost konzumaciji psihoaktivnih supstanci. U tu svrhu, koristit će se različite metode strojnog učenja, uključujući logističku regresiju, stablo odluke, slučajnu šumu (engl. *Random Forest*), *XGBoost* i stroj s potpornim vektorima (engl. *Support Vector Machine*). Usporedba modela provest će se na temelju njihovih performansi i sposobnosti generalizacije na testnim podacima. Osim toga, analizirat će se i značaj pojedinih osobina ličnosti u predikciji rizika, s ciljem boljeg razumijevanja psiholoških faktora potencijalno rizičnih pojedinaca.

2. Opis podataka i osnovni pojmovi

U ovom poglavlju predstavljeni su osnovni pojmovi relevantni za razumijevanje problema, opis korištenog skupa podataka te način na koji su podaci pripremljeni za analizu.

2.1. Osnovni pojmovi

- **Procjena rizika**: U kliničkom kontekstu, predstavlja sustavan proces identifikacije i evaluacije razine opasnosti kod pojedinaca koji pokazuju znakove upotrebe psihoaktivnih supstanci [2].
- **Psihoaktivne supstance**: Kemijske tvari koje mijenjaju moždanu funkciju, što rezultira privremenom promjenom percepcije, raspoloženja, svijesti ili ponašanja [3].
- **Osobine ličnosti** : Pretpostavljene, uglavnom trajne značajke čovjekove ličnosti koje jednu ličnost razlikuju od druge i na temelju kojih je moguće donekle predvidjeti ponašanje pojedinaca [4].

2.2. Skup podataka Drug Consumption

Kao skup za učenje modela korišten je javno dostupan skup podataka Drug Consumption (Quantified) [5]. Skup se sastoji od jedne csv datoteke koja sadrži demografske značajke, osobine ličnosti te informacije o konzumaciji 18 različitih psihoaktivnih supstanci za 1885 ispitanika.

2.2.1. Osobine ličnosti

Svaki je ispitanik ispunio tri upitnika usmjerena na procjenu osobina ličnosti: **NEO-FFI-R** (mjerenje pet glavnih osobina ličnosti), **BIS-11** (procjena impulzivnosti) te **ImpSS** (mjerenje sklonosti traženju uzbuđenja i impulzivnosti). Rezultati prikupljeni za svakog ispitanika predstavljeni su sljedećim pokazateljima:

- **nscore**: Neuroticizam
- **escore**: Ekstraverzija
- **oscore**: Otvorenost prema iskustvu
- **ascore**: Ugodnost
- **cscore**: Savjesnost
- **impulsive**: Impulzivnost
- **ss**: Traženje uzbuđenja

2.2.2. Psihoaktivne supstance

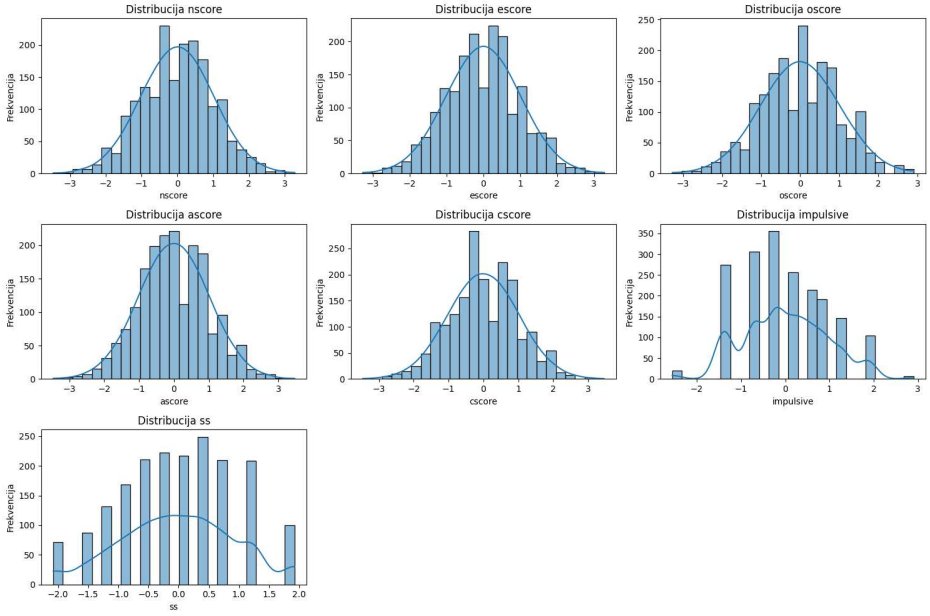
Svaki je ispitanik za 18 različitih, legalnih i ilegalnih, vrsta psihoaktivnih supstanci označio vremenski period zadnjeg korištenja. Psihoaktivne supstance su redom: *alkohol, amfetamin, amil nitrit, benzodiazepin, kofein, kanabis, čokolada, kokain, crack, MDMA, heroin, ketamin, legal highs, LSD, metamfetamin, gljive, nikotin i zloupraba hlapljivih tvari (VSA)*. Periodi korištenja bili su izraženi na sljedeći način:

- **CL0**: Nikad
- **CL1**: Prije više od desetljeća
- **CL2**: U zadnjem desetljeću
- **CL3**: U zadnjih godinu dana
- **CL4**: U zadnjih mjesec dana
- **CL5**: U zadnjih tjedan dana

- **CL6**: U protekla 24 sata

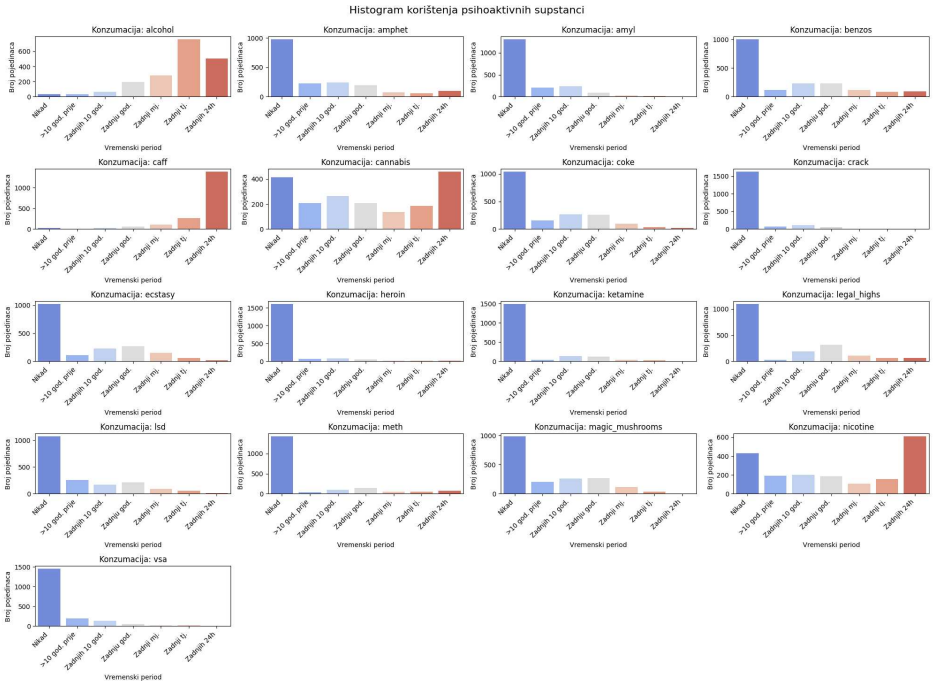
2.3. Obrada podataka

Podaci u izvornom skupu bili su već standardizirani i u formatu prikladnom za analizu i izgradnju modela, stoga nisu bile potrebne veće preinake. Slika 2.1. prikazuje distribuciju svake osobine ličnosti. Vidljivo je da svaka osobina prati približno normalnu razdiobu, što je očekivano za ovakvu vrstu podataka. Ipak, provedene su manje prilagodbe kako bi se osigurala dosljednost i kvaliteta podataka.



Slika 2.1. Distribucija osobina ličnosti

U skupu se nalazila i izmišljena droga pod nazivom *Semeron*, koja je služila za identifikaciju neiskrenih ispitanika. Svi ispitanici koji su naveli konzumaciju te tvari uklonjeni su iz analize. Također je iz skupa uklonjena i supstanca *čokolada*, budući da njezini učinci nisu usporedivi s učincima ostalih psihoaktivnih supstanci te se smatraju analitički nevažnima u kontekstu ovog istraživanja. Na slici 2.2. prikazana je konzumacija svake psihoaktivne supstance prema navedenim periodima korištenja (CL0 - CL6). Ovdje je jasno vidljiva neuravnoteženost klasa između redovitih i povremenih korisnika kod većine supstanci, osim kod alkohola, kofeina, kanabisa i nikotina, gdje je broj redovitih korisnika veći. To je također očekivano s obzirom da je riječ o široko rasprostranjenim i društveno prihvatljivijim psihoaktivnim supstancama.



Slika 2.2. Distribucija konzumacije psihoaktivnih supstanci

Kako bi se razlikovali rizični korisnici od prosječnih, u kontekstu konzumacije psihoaktivnih supstanci, definiran je pokazatelj ukupnog rizika (engl. *total risk factor*) za svakog ispitanika. Za svaku supstancu koju je ispitanik konzumirao, dodijeljena je određena količina bodova ovisno o vremenu posljednje konzumacije. Ukupan rizik pojedinog ispitanika dobiven je zbrajanjem bodova za sve supstance. Korišten je sljedeći sustav bodovanja:

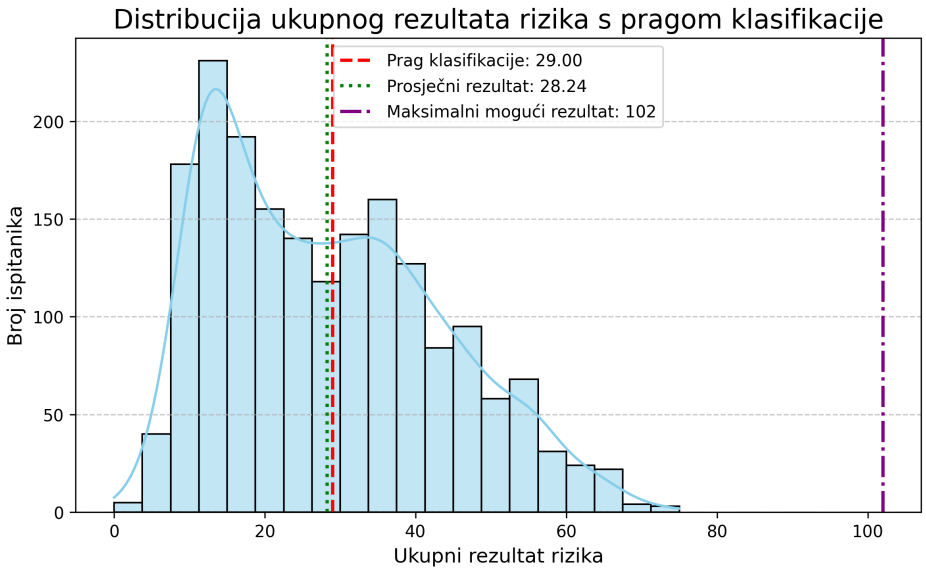
- CL6 (konzumacija unutar posljednjih 24 sata) – 6 bodova
- CL5 (unutar posljednjih tjedan dana) – 5 bodova
- CL4 (unutar posljednjih mjesec dana) – 4 boda
- CL3 (unutar posljednjih godinu dana) – 3 boda
- CL2 (prije više od godinu dana) – 2 boda
- CL1 (samo jednom u životu) – 1 bod
- CL0 (nikada nije konzumirao) – 0 bodova

Zatim je izračunat ukupan rizik svih ispitanika. Na temelju prosječne vrijednosti

ukupnog rizika svih ispitanika određena je granična vrijednost za kategorizaciju.

$$\text{risk threshold} = \text{mean}(\text{round}(\text{total risk})) + 1 \tag{2.1}$$

Ispitanici čiji rezultat prelazi ovu granicu smatrani su rizičnim korisnicima (slika 2.3.).



Slika 2.3. Distribucija rizične konzumacije

3. Opis i implementacija algoritama strojnog učenja

U ovom radu primjenjuje se nadzirano strojno učenje za procjenu rizika konzumacije psihoaktivnih supstanci. Nadzirano strojno učenje je pristup u kojem se modeli uče na skupu podataka koji sadrži i ulazne značajke i pripadajuće ciljne vrijednosti. Cilj takvog učenja je otkriti obrasce i odnose između ulaznih podataka i cilja, kako bi model mogao donositi točne predikcije na novim, dosad neviđenim podacima [6].

Korišteno je ukupno pet algoritama strojnog učenja, tri algoritma za izgradnju prediktivnih modela, te dva usmjerena na izgradnju interpretabilnih modela, kako bi se osim točnosti omogućilo i bolje razumijevanje odluka modela.

3.1. Priprema podataka za modeliranje

Ciljna varijabla definirana je binarno, na temelju prosječne vrijednosti ukupnog rizika, kako je prethodno opisano. Ulazne značajke korištene u modelima odnose se isključivo na osobine ličnosti ispitanika.

Kako bi se podaci podijelili na skup za učenje i skup za testiranje, korištena je funkcija `train_test_split` iz biblioteke `scikit-learn`. Za testiranje je izdvojeno 20% podataka (`test_size=0.2`) (slika 3.1.).

Parametar `random_state` postavljen je na vrijednost 42 kako bi se osigurao isti nasumični odabir podataka za učenje i testiranje pri svakom pokretanju koda.

```
y_class = (df['total_risk_score'] >= CLASSIFICATION_THRESHOLD_VALUE).astype(int)
X = df[feature_cols_personality]
X_train, X_test, y_train, y_test = train_test_split( *arrays: X, y_class, test_size=0.2, random_state=42, stratify=y_class)
```

Slika 3.1. Podjela podataka na skup za učenje i skup za testiranje

Parametar `stratify` postavljen je na ciljne vrijednosti (`y_class`) kako bi se očuvala proporcija klasa (rizični i nerizični korisnici) u oba skupa.

3.2. Odabir hiperparametara

Hiperparametri u strojnome učenju predstavljaju postavke modela koje se ne uče iz podataka, već ih definira korisnik prije procesa učenja modela [7].

Odabir prikladnih hiperparametara ključan je za postizanje dobre generalizacije modela, odnosno ravnoteže između preciznosti s učenim podacima i sposobnosti predviđanja na novim podacima. Postoji nekoliko pristupa za pretraživanje prostora hiperparametara, među kojima su najčešći ***Grid Search*** i ***Randomized Search***.

U ovom radu korišten je *RandomizedSearchCV*, metoda koja umjesto iscrpnog isprobavanja svih mogućih kombinacija, nasumično odabire fiksirani broj kombinacija hiperparametara iz unaprijed definiranih raspona. Ova metoda često značajno skraćuje vrijeme pretraživanja, a pritom može dovesti do jednako dobrih ili boljih rezultata u usporedbi s potpunom pretragom [8].

Metoda `optimize_model_random_search` služi za optimizaciju hiperparametara modela pomoću metode *RandomizedSearchCV* iz paketa `scikit-learn` (slika 3.2.).

Ulazni argumenti su sljedeći: `pipeline`, koji definira model i pripremu podataka, `param_distributions` koji predstavlja skup parametara i njihovih raspona koji se istražuju, te skupove za učenje `X_train` i `y_train`.

Unutar funkcije koristi se `StratifiedKFold` s `5` presjeka (`n_splits=5`), koji dijeli podatke na podskupove pri čemu se čuva proporcionalna zastupljenost klasa (stratifikacija). Postavljeni su parametri `shuffle=True` radi nasumičnog miješanja podataka prije podjele te `random_state=42` za reproducibilnost.

Kao metrika koristi se F1 mjera (`scoring='f1'`) koja osigurava ravnotežu između osjetljivosti (engl. *recall*) i točnosti (engl. *precision*) modela. Nakon učenja, funkcija vraća najbolji pronađeni model (`best_model`) i pripadajuće hiperparametre (`best_params`).

```python
def optimize_model_random_search(pipeline, param_distributions, X_train, y_train):
    kf = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)

    random_search = RandomizedSearchCV(
        pipeline,
        param_distributions=param_distributions,
        n_iter=100,
        cv=kf,
        scoring='f1',
        random_state=42
    )

    random_search.fit(X_train, y_train)

    best_model = random_search.best_estimator_
    best_params = random_search.best_params_

    return best_model, best_params
```

Slika 3.2. Funkcija za odabir hiperparametara

3.2.1. SMOTE (Synthetic Minority Oversampling Technique)

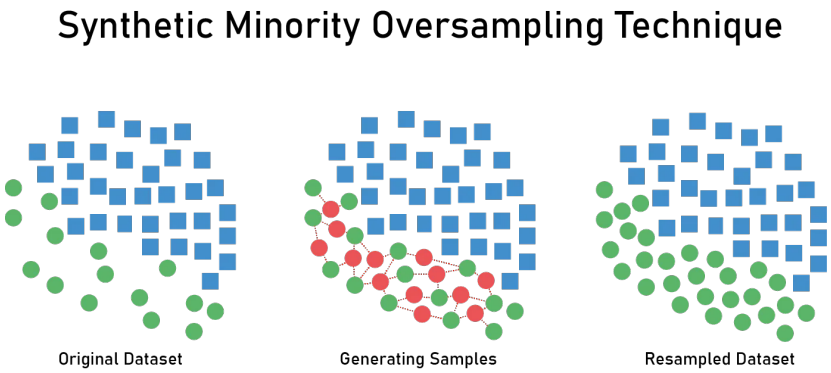
Kako bi se poboljšala učinkovitost modela i smanjila pristranost prema većinskoj klasi, u ovom radu korištena je tehnika **SMOTE** . Riječ je o metodi koja umjetno povećava broj uzoraka manjinske klase tako da generira nove primjere na temelju postojećih. Time se postiže ravnoteža između klasa, što pomaže modelu da bolje generalizira i ne favorizira dominantnu klasu, osobito kod neravnotežnih skupova podataka [9] (slika 3.3.).

Sintetski primjer generira se interpolacijom između postojećeg uzorka manjinske klase x_i i jednog od njegovih najbližih susjeda x_j iz iste klase, prema sljedećoj formuli:

$$x_{\text{new}} = x_i + \delta \cdot (x_j - x_i) \tag{3.1}$$

gdje je δ slučajna vrijednost u rasponu $[0, 1]$. Time se novi podatkovni primjer x_{new} nalazi negdje na linijskom segmentu između x_i i x_j, što omogućuje stvaranje varijacija unutar manjinske klase bez dupliciranja postojećih uzoraka.

Primjena metode SMOTE važna je jer pomaže u smanjenju prenaučenosti (engl. *overfitting*), koja se može javiti kada je model previše prilagođen uzorcima većinske klase.



Slika 3.3. Vizualizacija SMOTE metode [10]

3.3. Interpretabilni modeli

Interpretabilni modeli su modeli strojnog učenja čije je donošenje odluka transparentno i razumljivo ljudima. Takvi su modeli posebno korisni u područjima, poput psihologije ili medicine, gdje se uz točnost traži i tumačenje rezultata [11]. Za izgradnju interpretabilnih modela korišteni su algoritmi stablo odluke i logistička regresija.

3.3.1. Stablo odluke

Stablo odluke (engl. *Decision Tree*) je algoritam strojnog učenja koji se koristi za klasifikacijske i regresijske zadatke. Model predikciju temelji na strukturiranju odluka u obliku stabla, gdje svaki unutarnji čvor predstavlja test nad nekom značajkom, grane predstavljaju ishod tog testa, a listovi konačne klasifikacije [12].

Iz modela dobivenog stablom odluke moguće je generirati niz **ako-onda pravila** koja jasno opisuju odluke na svakom čvoru stabla. Ta pravila omogućuju jednostavno praćenje puta donošenja odluke i razumijevanje uvjeta pod kojima model klasificira podatke u određenu klasu.

Algoritam postupno dijeli podatke na manje skupove tako da na svakom koraku bira značajku i prag koji najbolje razlikuju klase. Najčešće korišteni kriteriji za odabir podjele su **Gini indeks** i **entropija**:

- Jednadžba za Gini indeks:

$$Gini(t) = 1 - \sum_{i=1}^{C} p_i^2 \tag{3.2}$$

  gdje je p_i udio uzoraka klase i u čvoru t, a C broj klasa.

```
pipeline_dt_smote = Pipeline([
    ('smote', SMOTE(random_state=42)),
    ('decision_tree', DecisionTreeClassifier(random_state=42))
])
param_distributions_dt = {
    'decision_tree__max_depth': [3, 5, 7, 10, 15],
    'decision_tree__min_samples_split': [2, 5, 10, 20],
    'decision_tree__min_samples_leaf': [1, 3, 5, 10],
    'decision_tree__criterion': ['gini', 'entropy']
}

best_dt_model, best_dt_params = optimize_model_random_search(pipeline=pipeline_dt_smote,
                                                             param_distributions=param_distributions_dt,
                                                             X_train=X_train, y_train=y_train)

dt_eval_results = evaluate_model(model=best_dt_model, X_test=X_test, y_test=y_test, model_name="Decision Tree")
```

Slika 3.4. Implementacija algoritma stablo odluke

- Jednadžba za entropiju:

$$H(t) = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{3.3}$$

Prednosti algoritma:

- Intuitivan i lako interpretabilan model.
- Može raditi s numeričkim i kategorijskim podacima.

Nedostaci algoritma:

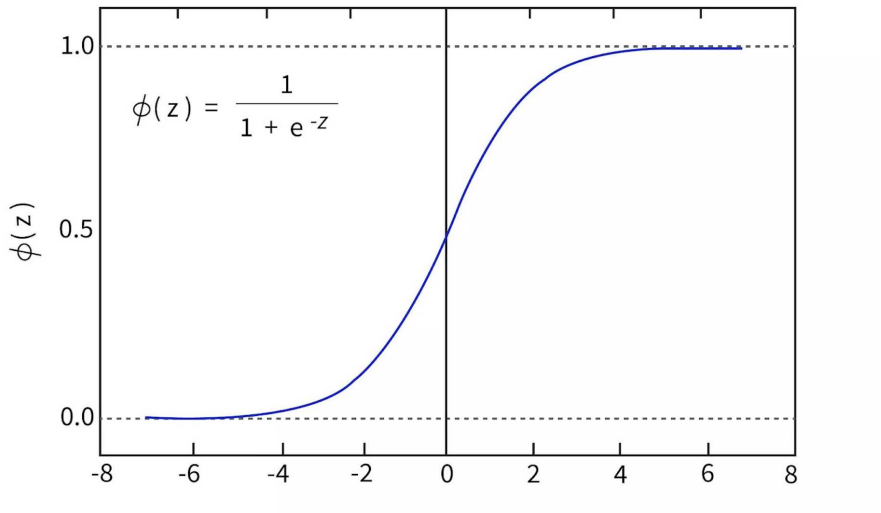
- Sklon prenaučenosti, osobito kod velikih stabala.
- Osjetljiv na male promjene u podacima.

Implementacija

U implementaciji stabla odluke korišten je model `DecisionTreeClassifier` iz paketa *scikit-learn*, u kombinaciji s metodom SMOTE i cjevovodom (`Pipeline`), koji su prethodno objašnjeni. Optimiraju se odabrani hiperparametri modela:

- `max_depth`: maksimalna dubina stabla,
- `min_samples_split`: minimalan broj uzoraka za podjelu čvora,
- `min_samples_leaf`: minimalan broj uzoraka u listu,
- `criterion`: *gini, entropy*

Na kraju se model s najboljim hiperparametrima evaluira na testnom skupu podataka



Slika 3.5. Graf sigmoidne funkcije

kako je vidljivo na slici 3.4.

3.3.2. Logistička regresija

Logistička regresija (engl. *Logistic regression*) je popularan algoritam za binarnu klasifikaciju koji modelira vjerojatnost da ulazni primjer pripada određenoj klasi. Za razliku od linearne regresije, koja daje kontinuiranu vrijednost, logistička regresija koristi sigmoidnu funkciju kako bi ograničila izlaz između 0 i 1. Sigmoidna funkcija prikazana je na slici 3.5. te u obliku izraza jednadžbe:

$$\sigma(z) = \frac{1}{1 + e^{-z}}, \quad \text{gdje je } z = \mathbf{w}^T\mathbf{x} + b \tag{3.4}$$

U izrazu 3.5 prikazana je jednadžba za računanje logističke regresije.

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \sum_{i=1}^{p} \beta_i x_i \tag{3.5}$$

Koeficijenti modela (β_i), predstavljaju težine koje pokazuju koliko pojedina značajka doprinosi povećanju ili smanjenju vjerojatnosti pripadnosti određenoj klasi. Zbog toga možemo smatrati model dobiven logističkom regresijom interpretabilnim jer omogućuje direktno razumijevanje utjecaja svake značajke na konačnu odluku modela.

```
pipeline_lr_smote = Pipeline([
    ('smote', SMOTE(random_state=42)),
    ('classifier', LogisticRegression(random_state=42))
])

param_distributions_lr = {
    'classifier__C': loguniform(0.01, 100),
    'classifier__solver': ['liblinear', 'saga'],
    'classifier__penalty': ['l1', 'l2']
}

best_lr_model, best_lr_params = optimize_model_random_search(pipeline=pipeline_lr_smote,
                                                             param_distributions=param_distributions_lr,
                                                             X_train=X_train, y_train=y_train)

lr_eval_results = evaluate_model(model=best_lr_model, X_test=X_test,
                                 y_test=y_test, model_name="Logistic Regression")
```

Slika 3.6. Implementacija logističke regresije

Prednosti algoritma:

- Jednostavna i brza implementacija.
- Dobro interpretabilni koeficijenti koji omogućuju analizu utjecaja pojedine značajke.

Nedostaci algoritma:

- Pretpostavlja linearnu separabilnost podataka.
- Lošija učinkovitost na kompleksnim i nelinearnim skupovima podataka.

Implementacija

U implementaciji je korišten algoritam `LogisticRegression` iz biblioteke *scikit-learn*.

Optimiziraju se odabrani hiperparametri modela:

- `C`: jačina regularizacije,
- `solver`: algoritam optimizacije modela,
- `penalty`: vrsta regularizacije za sprečavanje prenaučenosti

Nakon odabira najboljih hiperparametara, model se uči i evaluira na testnom skupu pomoću funkcije `evaluate_model` kao što je prikazano na slici 3.6.

3.4. Prediktivni modeli

Prediktivni modeli strojnog učenja primarno se koriste u situacijama kada je cilj postići što veću točnost u predviđanju ishoda. Zbog njihove složenosti, ovi modeli su često manje interpretabilni te se najčešće primjenjuju kada su dostupne velike količine podataka, a točnost predikcije ima veću važnost od razumljivosti modela. Za izgradnju prediktivnih modela u ovom radu korišteni su sljedeći algoritmi: stroj s potpornim vektorima (SVM), slučajna šuma i XGBoost.

3.4.1. Stroj s potpornim vektorima

Stroj s potpornim vektorima (engl. *Support Vector Machine*, SVM) je nadzirani algoritam strojnog učenja koji se koristi za klasifikacijske i regresijske zadatke. Ideja algoritma je pronaći optimalnu hiperravninu koja najbolje razdvaja klase u prostoru značajki. U kontekstu binarne klasifikacije, cilj je maksimizirati marginu između najbližih točaka svake klase, poznatih kao potporni vektori (slika 3.7.) [13].

U slučaju linearnog razdvajanja, hiperravnina se definira jednadžbom:

$$\mathbf{w}^T\mathbf{x} + b = 0 \tag{3.6}$$

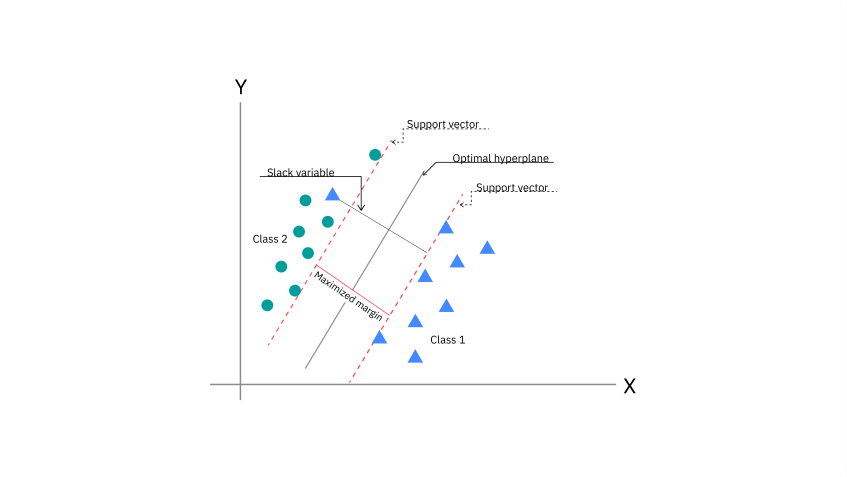
gdje je $\mathbf{w}$ normalni vektor na hiperravninu, a b slobodni član. Za nelinearno razdvojive podatke, koristi se jezgreni trik kojim se podaci preslikavaju u prostor više dimenzije u kojem su linearno razdvojivi.

Prednosti algoritma:

- Robustan je na prenaučenost, posebno pri korištenju regularizacije i kernela.
- Učinkovit u visokodimenzionalnim prostorima.

Nedostaci algoritma:

- Sporiji na velikim skupovima podataka.
- Teže interpretabilan u odnosu na jednostavnije modele poput logističke regresije.



Slika 3.7. Vizualizacija stroja s potpornim vektorima [13]

```
pipeline_svc_smote = Pipeline([
    ('smote', SMOTE(random_state=42)),
    ('svc', SVC(random_state=42, probability=True, class_weight='balanced'))
])

param_distributions_svc = {
    'svc__C': loguniform(0.01, 100),
    'svc__kernel': ['linear', 'rbf', 'poly', 'sigmoid'],
    'svc__gamma': loguniform(0.001, 1),
    'svc__degree': [2, 3, 4]
}

best_svc_model, best_svc_params = optimize_model_random_search(pipeline=pipeline_svc_smote,
                                                               param_distributions=param_distributions_svc,
                                                               X_train=X_train, y_train=y_train)

svc_eval_results = evaluate_model(model=best_svc_model, X_test=X_test,
                                  y_test=y_test, model_name="Support Vector Machine")
```

Slika 3.8. Implementacija stroja s potpornim vektorima

Implementacija

U provedbi se koristi klasa `SVC` iz biblioteke *scikit-learn*, uz podršku za izračun vjerojatnosti klasifikacije i balansiranje klasa.

Parametri koji se optimiraju uključuju:

- `C`: regulacijski parametar za kompromis između maksimalne margine i pogreške klasifikacije,
- `kernel`: tip jezgre (linearna, RBF, polinomna, sigmoid),
- `gamma`: dodatni parametar jezgre,
- `degree`: stupanj polinoma za polinomnu jezgru.

Optimizacija hiperparametara provodi se funkcijom `optimize_model_random_search`, nakon čega se model evaluira na testnim podacima kako je prikazano na slici 3.8.

3.4.2. Slučajna šuma

Slučajna šuma vrsta je algoritma strojnog učenja zasnovanog na ansamblima. To znači da se ne koristi samo jedan model, već se kombiniraju rezultati više jednostavnijih modela za donošenje odluke. Slučajna šuma gradi velik broj stabala odluke, a svako stablo uči se na blago drugačijem skupu podataka te koristi blago drugačiji skup značajki (tehnika *feature bagging*). Završna predikcija donosi se većinskim glasanjem (engl. *majority voting*) svih stabala.

Feature bagging ili nasumičnost odabira značajki, generira nasumični podskup značajki, što osigurava nisku korelaciju među stablima. To je ključna razlika između stabala odluke i slučajne šume. Dok stabla odluke razmatraju sve moguće podjele svih značajki, slučajna šuma odabire samo podskup tih značajki [14].

Svako stablo daje svoju predikciju, a ona klasa koju predloži najviše stabala postaje konačan rezultat modela [14].

Prednosti:

- Visoka točnost i robusnost na prenaučenost u odnosu na pojedinačna stabla.
- Učinkovit za velike skupove podataka i visoku dimenzionalnost.

Nedostatci:

- Teže interpretabilan model zbog velikog broja stabala.
- Potrošnja memorije i vrijeme učenja mogu biti veći u odnosu na jednostavnije modele.

Implementacija

U provedbi se koristi `RandomForestClassifier` iz biblioteke *scikit-learn*.

Većina hiperparametara podudara se s onima iz stabla odluke (pogledati poglavlje 3.3.1.),

```
pipeline_rf_smote = Pipeline([
    ('smote', SMOTE(random_state=42)),
    ('random_forest', RandomForestClassifier(random_state=42))
])

param_distributions_rf = {
    'random_forest__n_estimators': [100, 200, 300, 500],
    'random_forest__max_depth': [5, 10, 15, 20, None],
    'random_forest__min_samples_split': [2, 5, 10],
    'random_forest__min_samples_leaf': [1, 2, 4],
    'random_forest__criterion': ['gini', 'entropy'],
    'random_forest__max_features': ['sqrt', 'log2', 1.0]
}

best_rf_model, best_rf_params = optimize_model_random_search(pipeline=pipeline_rf_smote,
                                                             param_distributions=param_distributions_rf,
                                                             X_train=X_train, y_train=y_train)

rf_eval_results = evaluate_model(model=best_rf_model, X_test=X_test, y_test=y_test, model_name="Random Forest")
```

Slika 3.9. Implementacija slučajne šume

a model dodatno uključuje još dva specifična hiperparametra:

- `n_estimators`: broj stabala u šumi
- `max_features`: maksimalan broj značajki pri traženju najbolje podjele

Optimizacija hiperparametara vrši se funkcijom `optimize_model_random_search`, a nakon učenja modela provodi se evaluacija na testnom skupu (slika 3.9.).

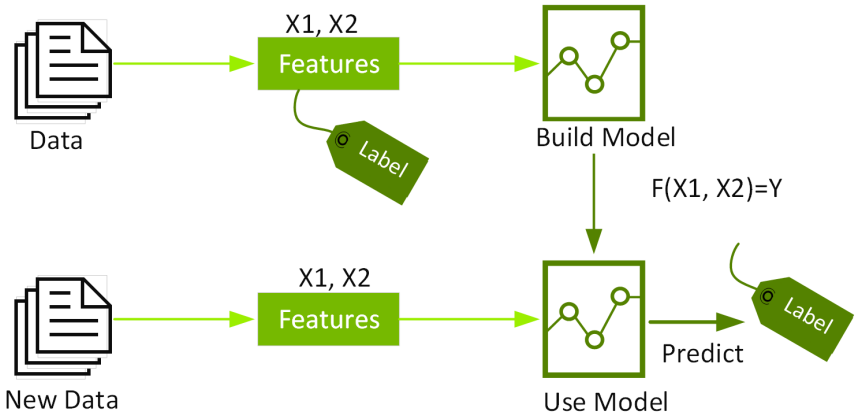
3.4.3. XGBoost

XGBoost (engl. *Extreme Gradient Boosting*) napredna je inačica algoritma gradijentnog pojačanja (engl. *Gradient Boosting*). Riječ je o metodi ansambala koja stvara niz stabala odluke, gdje svako sljedeće stablo pokušava ispraviti pogreške prethodnih modela (slika 3.10. XGBoost je posebno osmišljen da bude visoko učinkovit, točan i skalabilan, te uključuje brojne optimizacije poput paralelizacije, regularizacije i rukovanja s nedostajućim vrijednostima [15].

Prednosti:

- Visoka prediktivna točnost.
- Podrška za regularizaciju (L1 i L2) čime se smanjuje rizik od prenaučenosti.
- Ugrađena podrška za rješavanje neuravnoteženih klasa.

Nedostatci:



Slika 3.10. Vizualizacija algoritma XGBoost [16]

```
pipeline_xgb_smote = Pipeline([
    ('smote', SMOTE(random_state=42)),
    ('xgb_classifier', XGBClassifier(objective='binary:logistic', eval_metric='logloss', random_state=42))
])

param_distributions_xgb = {
    'xgb_classifier__n_estimators': [100, 300, 500, 700, 1000],
    'xgb_classifier__learning_rate': loguniform(0.01, 0.3),
    'xgb_classifier__max_depth': [3, 5, 7, 10],
    'xgb_classifier__subsample': uniform(loc=0.6, scale=0.4),
    'xgb_classifier__colsample_bytree': uniform(loc=0.6, scale=0.4),
}

best_xgb_model, best_xgb_params = optimize_model_random_search(pipeline=pipeline_xgb_smote,
                                                               param_distributions=param_distributions_xgb,
                                                               X_train=X_train, y_train=y_train)

xgb_eval_results = evaluate_model(model=best_xgb_model, X_test=X_test, y_test=y_test, model_name="XGBoost")
```

Slika 3.11. Implementacija XGBoost-a

- Složenost modela otežava interpretaciju.
- Parametri modela su brojni i njihova optimizacija može biti vremenski zahtjevna.
- Veća potrošnja memorije i vremena učenja u usporedbi s jednostavnijim modelima.

Implementacija

Za implementaciju se koristi `XGBClassifier` iz biblioteke `xgboost`.

Optimizacija hiperparametara uključuje:

- `n_estimators`: broj stabala,
- `learning_rate`: veličina koraka pri svakom učenju,

- `max_depth`: maksimalna dubina stabala,
- `subsample`: udio podataka korišten za učenje svakog stabla,
- `colsample_bytree`: udio značajki korišten za svako stablo,

Najbolji model pronalazi se korištenjem metode `optimize_model_random_search`, a zatim se evaluira na testnom skupu (slika 3.11.).

4. Rezultati i rasprava

Nakon učenja i evaluacije svih modela, u ovom poglavlju slijedi analiza dobivenih rezultata i usporedba uspješnosti pojedinih modela.

Za svaki model razmatraju se kvantitativne značajke poput točnosti (engl. *accuracy*), osjetljivosti (engl. *recall*), preciznosti (engl. *precision*), mjere F1, krivulje ROC te matrice konfuzije (engl. *confusion matrix*).

Kod interpretabilnih modela, dodatna pažnja usmjerena je na razumijevanje kako model donosi predikcije. U ovom radu, korišteni interpretabilni modeli, pružaju analizu konkretnih pravila i važnost pojedinih osobina ličnosti u odlučivanju.

Svi modeli koriste istu metodu, `plot_model_result`, za iscrtavanje ili ispis kvantitativnih i kvalitativnih značajki navedenih iznad. Metoda je prikazana na slici 4.1.

4.1. Usporedba modela

Rezultati evaluacije stabla odluke prikazuju umjerene performanse modela, kako se vidi na slici 4.2. Vrijednost ROC-AUC od 0.7902 potvrđuje solidnu sposobnost modela u razlikovanju klasa.

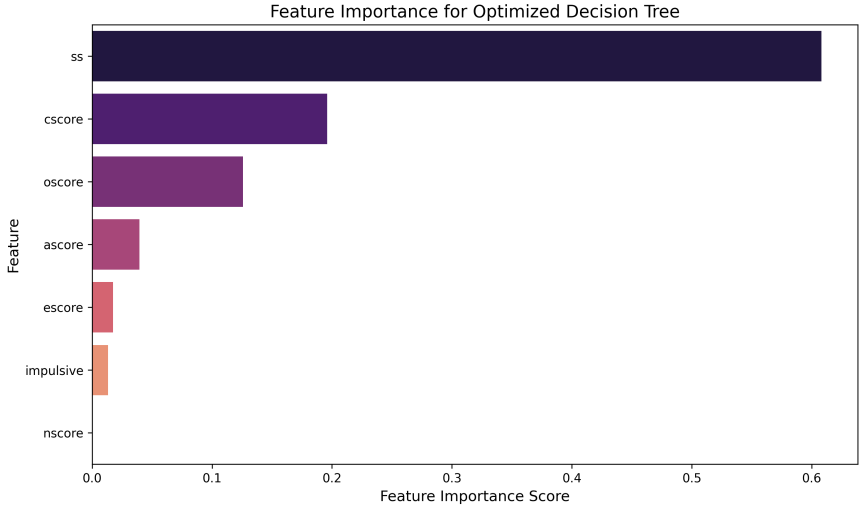
Iako stablo odluke ima slabije rezultate u odnosu na složenije modele, njegova najveća prednost leži u interpretabilnosti. Na slici i 4.3. jasno je vidljivo koje su osobine

```
plot_model_results(evaluation_results=dt_eval_results,
                   feature_names=feature_cols_personality,
                   class_names=['Non-Risky', 'Risky'])
```

Slika 4.1. Funkcija za iscrtavanje značajki modela

```
Decision tree evaluation results:
Accuracy: 0.7287
Precision: 0.6882
Recall: 0.7442
F1-Score: 0.7151
ROC-AUC: 0.7902
Matrica konfuzije:
 [[146  58]
 [ 44 128]]
```

Slika 4.2. Metrike za stablo odluke

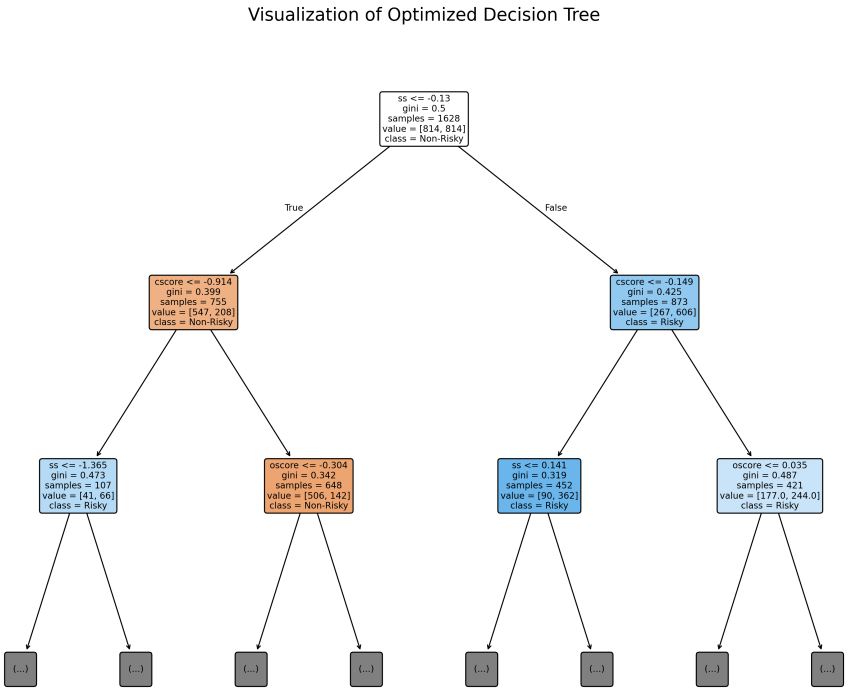


Slika 4.3. Važnost osobina ličnosti po modelu stabla odluke

ličnosti najvažnije za donošenje odluka: traženje uzbuđenja (*ss*), savjesnost (*cscore*) i otvorenost prema iskustvima (*oscore*).

Iako je za optimalan parametar `max_depth` bilo odabrano stablo dubine pet, radi veće preglednosti i razumljivosti slika 4.4. prikazuje stablo odluke s tri razine. Svaki čvor u stablu prikazuje uvjet podjele (npr. ss <= -0.13), broj uzoraka koji prolaze kroz taj čvor, distribuciju klasa (npr. [814, 814]) te Gini indeks.

Ovakav pojednostavljeni prikaz omogućuje lako praćenje toka donošenja odluka, čak i korisnicima bez dubinskog znanja o modelima strojnog učenja. Jasna hijerarhijska struktura stabla olakšava razumijevanje kako pojedine osobine utječu na klasifikaciju, što ga čini vrlo korisnim alatom za interpretaciju i komunikaciju rezultata modela na intuitivan način.

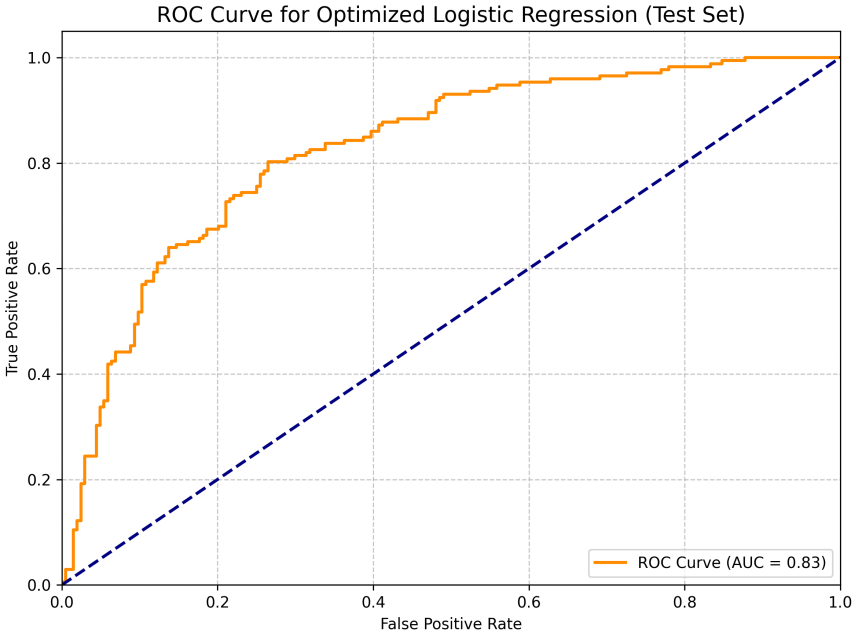


Slika 4.4. Vizualizacija stabla odluke s 3 razine

Drugi interpretabilni model u ovom radu, logistička regresija, ima bolje prediktivne performanse od stabla odluke i to je prikazano na slici 4.6. Od svih modela, logistička regresija ima najbolju krivulju ROC-AUC, koja je prikazana na slici 4.5. i ukazuje na vrlo dobru diskriminativnu moć modela.

Na slici 4.7. prikazane su važnosti pojedinih osobine ličnosti. Lako je uočiti da je i ovdje dominantna osobina ličnosti traženje uzbuđenja (*ss*). Za razliku od stabla odluke, logistička regresija malo veću važnost daje otvorenosti prema iskustvima nad savjesnosti. Također je bitno primijetiti da koeficijenti logističke regresije mogu biti pozitivni ili negativni. Vidljivo je da su *ss* i *oscore* pozitivne vrijednosti, a *cscore*, *escore* i *ascore* negativne vrijednosti. To znači da rizični korisnici psihoaktivnih supstanci pokazuju veću otvorenost prema iskustvima i traženju uzbuđenja, a manje su savjesni i ekstrovertirani.

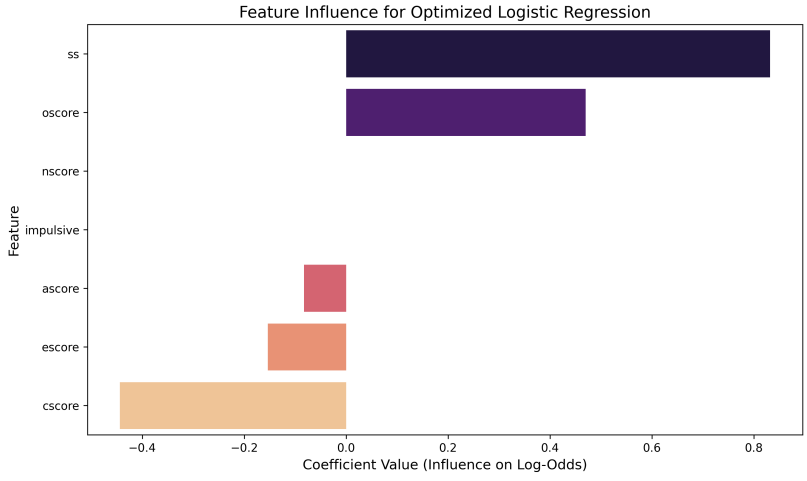
Zanimljivo je da su važnosti značajki neuroticizma (*nscore*) i impulzivnosti (*impulsive*) u oba modela jednaka nuli. Ovakav rezultat iznenađuje s obzirom na to da brojna prethodna istraživanja ističu visok neuroticizam kao jedan od ključnih prediktora rizičnih oblika ponašanja, uključujući i konzumaciju psihoaktivnih tvar [1]. Autori su u ovom radu očekivali su da će osobe koje konzumiraju drogu imati izraženije emocionalne



Slika 4.5. Krivulja ROC-AUC za logističku regresiju

```
Logistic regression evaluation results:
Accuracy: 0.7633
Precision: 0.7150
Recall: 0.8023
F1-Score: 0.7562
ROC-AUC: 0.8269
Matrica konfuzije:
 [[149  55]
 [ 34 138]]
```

Slika 4.6. Metrike logističke regresije



Slika 4.7. Važnost osobina ličnosti po modelu logističke regresije

```
SVM evaluation results:
Accuracy: 0.7553
Precision: 0.7105
Recall: 0.7849
F1-Score: 0.7459
ROC-AUC: 0.8228
Matrica konfuzije:
 [[149  55]
 [ 37 135]]
```

Slika 4.8. Metrike za stroj s potpornim vektorima

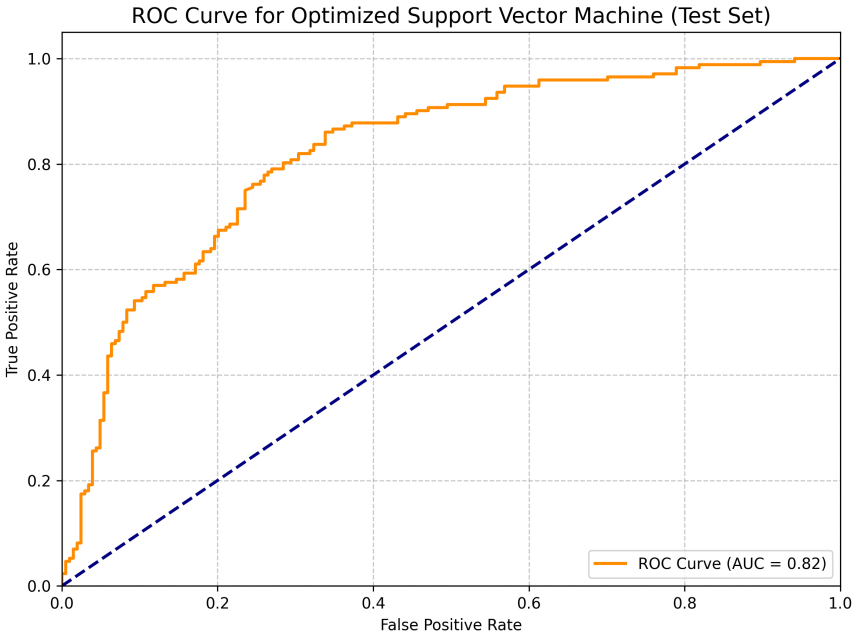
teškoće, veću impulzivnost i sklonost djelovanju pod utjecajem negativnih emocija (engl. *negative urgency*), što je klasično obilježje visoke razine neuroticizma u kombinaciji s niskim razinama savjesnosti i ugodnosti. Međutim, u ovom istraživanju ni neuroticizam ni impulzivnost nisu se pokazali relevantnima, što može upućivati na to da su drugi aspekti ličnosti imali daleko snažniji učinak ili da među ispitanicima razlike u tim značajkama jednostavno nisu bile dovoljno izražene da bi modeli mogli detektirati uzorak.

Stroj s potpornim vektorima prikazuje vrlo dobre rezultate. Ima odziv od 78.5% i F1-score od 75% (slika 4.8.). Zanimljivo je da je kao najbolji hiperparamter za jezgru odabrana jezgra s radijalnom bazom iz čega je moguće zaključiti da je model pristupio ovom problemu kao nelinearnoj klasifikaciji. Uz logističku regresiju, SVM model najbolje diskriminira podatke, što je vidljivo iz krivulje ROC-AUC (slika 4.9.)

XGBoost konkurira spomenutim modelima s dobrim odzivom od 79.66% i krivuljom ROC-AUC koja se od logističke regresije i stroja s potprnim vektorima razlikuje u svega par decimala. No, od svih modela, XGBoost ima najnižu preciznost, od 69% (slika 4.10.). S obzirom da svi modeli variraju oko sličnih vrijednosti za metrike, ovakvi detalji su bitni za određivanje optimalnog rješenja.

Na slici 4.10. prikazana je i matrica konfuzije za XGBoost iz koje je vidljivo da broj krivo klasificiranih rizičnih korisnika kao nerizičnih iznosi 35, što je druga najmanja vrijednost nakon logističke regresije.

Zadnji evaluirani model, slučajna šuma, pokazuje slične rezultate kao i ostali modeli.



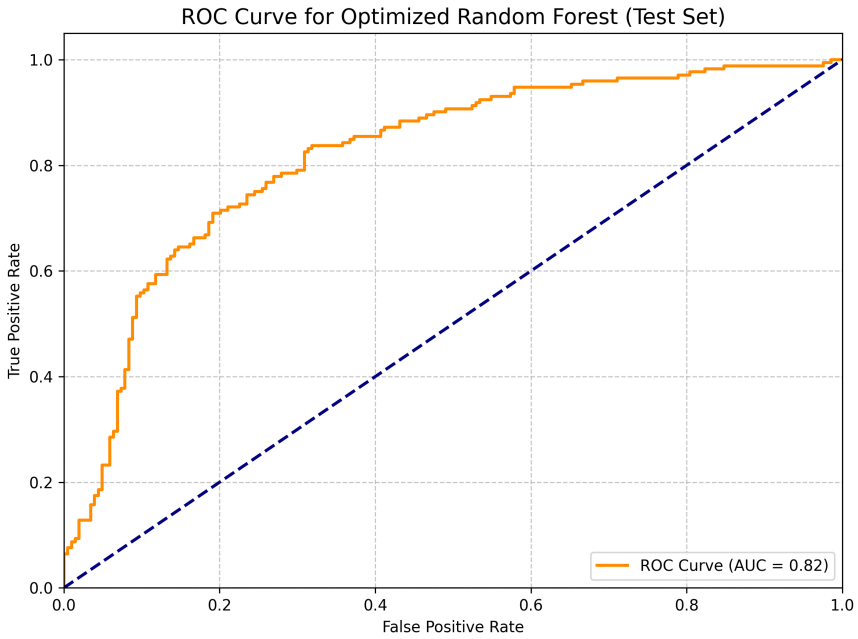
Slika 4.9. Krivulja ROC-AUC za stroj s potpornim vektorima

```
XGBoost evaluation results:
Accuracy: 0.7447
Precision: 0.6919
Recall: 0.7965
F1-Score: 0.7405
ROC-AUC: 0.8223
Matrica konfuzije:
 [[143  61]
 [ 35 137]]
```

Slika 4.10. Metrike za XGBoost

```
Random forest evaluation results:
Accuracy: 0.7500
Precision: 0.7074
Recall: 0.7733
F1-Score: 0.7389
ROC-AUC: 0.8181
Matrica konfuzije:
 [[149  55]
 [ 39 133]]
```

Slika 4.11. Metrike za slučajnu šumu



Slika 4.12. Krivulja ROC-AUC za slučajnu šumu

Na slici 4.11. jasno je da slučajna šuma, iako daje dobre vrijednosti, još uvijek ne nadmašuje logističku regresiju. Dobro diskriminira podatke, što je vidiljivo iz slike 4.12., koja prikazuje da je krivulja ROC-AUC jednako dobra kao i kod logističke regresije i SVM-a.

5. Zaključak

Na temelju rezultata i rasprave u prošlom poglavlju, može se zaključiti da svi modeli pokazuju slične performanse. Logistička regresija i stroj s potpornim vektorima (SVM) pokazali su se kao najučinkovitiji modeli za predviđanje rizične konzumacije supstanci na danom skupu podataka o osobinama ličnosti. Oba modela postižu vrlo dobru ravnotežu između preciznosti i odziva, što je od iznimne važnosti u procjeni rizika. Blagu prednost preuzima logistička regresija, s obzirom da je, uz dobre prediktivne performance, lako interpretirati rezultate modela.

Rezultati potvrđuju da osobine ličnosti imaju ulogu u oblikovanju sklonosti prema konzumaciji psihoaktivnih supstanci. Osobe s izraženim traženjem uzbuđenja i otvorenošću prema iskustvima češće pokazuju rizične obrasce ponašanja, što je u skladu s teorijskim očekivanjima.

S obzirom na ograničenja korištenog skupa podataka, koji obuhvaća uglavnom nerizične korisnike, vjerojatno bi raznovrsniji uzorak, s većom ravnotežom klasa, omogućio detaljniju analizu i doveo do preciznijih uvida.

Modeliranje je bilo usmjereno na maksimizaciju odziva kako bi se smanjio broj rizičnih korisnika koji bi ostali neprepoznati. Iako same osobine ličnosti nisu dovoljne za visoko točne predikcije, njihova značajnost u kombinaciji s drugim čimbenicima, poput demografskih i socijalnih, može davati vrlo dobre rezultate za procjenu rizika, čime ovaj rad postavlja dobru osnovu za daljnja istraživanja i razvoj učinkovitijih modela.

Literatura

[1] E. Fehrman, A. K. Muhammad, E. M. Mirkes, V. Egan, i A. N. Gorban, "The five factor model of personality and evaluation of drug consumption risk", 2017. [Mrežno]. Adresa: https://arxiv.org/abs/1506.06297

[2] J. McNeely, L. K. Hamilton, S. D. Whitley *et al.*, *Substance Use Screening, Risk Assessment, and Use Disorder Diagnosis in Adults*. Baltimore (MD): Johns Hopkins University, May 2024. [Mrežno]. Adresa: https://www.ncbi.nlm.nih.gov/books/NBK565474/

[3] Wikipedia contributors, "Kategorija:psihoaktivne droge", Adresa: https://hr.wikipedia.org/wiki/Kategorija:Psihoaktivne_droge, 2025., mrežno, pristupljeno: lipanj 2025.

[4] Leksikografski zavod Miroslav Krleža, "Crte ličnosti", Adresa: https://www.enciklopedija.hr/clanak/crte-licnosti, 2013., mrežno, pristupljeno: lipanj 2025.

[5] E. Fehrman, V. Egan, i E. Mirkes, "Drug consumption (quantified)", https://doi.org/10.24432/C5TC7S, 2015., dataset available at UCI Machine Learning Repository.

[6] GeeksforGeeks, "Supervised and unsupervised learning", https://www.geeksforgeeks.org/machine-learning/supervised-unsupervised-learning/, 2025., mrežno, pristupljeno: lipanj 2025.

[7] M. Lapan, "Parameters and hyperparameters", https://towardsdatascience.com/parameters-and-hyperparameters-aa609601a9ac/, 2018., mrežno, pristupljeno: lipanj 2025.

[8] GeeksforGeeks, “Hyperparameter tuning in machine learning”, https://www.geeksforgeeks.org/machine-learning/hyperparameter-tuning/, 2025., mrežno, pristupljeno: lipanj 2025.

[9] C. Maklin, “Synthetic minority over-sampling technique (smote)”, https://medium.com/@corymaklin/synthetic-minority-over-sampling-technique-smote-7d419696b88c, 2019., mrežno, pristupljeno: lipanj 2025.

[10] R. Biswas, “Bank data: Smote”, https://medium.com/analytics-vidhya/bank-data-smote-b5cb01a5e0a2, 2020., mrežno, pristupljeno: lipanj 2025.

[11] C. Molnar, *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*, 2024. [Mrežno]. Adresa: https://christophm.github.io/interpretable-ml-book/interpretability.html

[12] GeeksforGeeks, “Decision tree in machine learning”, https://www.geeksforgeeks.org/machine-learning/decision-tree/, 2025., mrežno, pristupljeno: lipanj 2025.

[13] IBM, “What are svms?” https://www.ibm.com/think/topics/support-vector-machine, 2025., mrežno, pristupljeno: lipanj 2025.

[14] ——, “What is random forest?” https://www.ibm.com/think/topics/random-forest, 2025., mrežno, pristupljeno: lipanj 2025.

[15] GeeksforGeeks, “Xgboost in machine learning”, https://www.geeksforgeeks.org/machine-learning/xgboost/, 2025., mrežno, pristupljeno: lipanj 2025.

[16] NVIDIA, “Xgboost”, https://www.nvidia.com/en-us/glossary/xgboost/, 2025., mrežno, pristupljeno: lipanj 2025.

Sažetak

Modeli procjene rizika za konzumaciju psihoaktivnih supstanci temeljeni na osobinama ličnosti i metodama strojnog učenja

Jana Gazdek

U ovom radu istražuje se mogućnost procjene rizika konzumacije psihoaktivnih supstanci na temelju osobina ličnosti. Korišten je javno dostupan skup podataka koji sadrži psihološke i demografske karakteristike pojedinaca, s naglaskom na petofaktorski model ličnosti. Cilj je bio izgraditi i evaluirati različite modele strojnog učenja, uključujući logističku regresiju, stablo odluke, SVM, slučajnu šumu i XGBoost, kako bi se predvidjela sklonost konzumaciji.

Rezultati pokazuju da osobine poput traženja uzbuđenja i otvorenosti prema iskustvima mogu imati utjecaj na rizično ponašanje. Iako modeli temeljeni isključivo na osobinama ličnosti ne postižu vrhunsku točnost, pružaju vrijedan uvid u obrasce ponašanja te bi u kombinaciji s drugim vrstama podataka mogli činiti snažan alat za ranu identifikaciju rizika. Poseban naglasak stavljen je na interpretabilnost i smanjenje lažno negativnih klasifikacija. Rad postavlja temelje za daljnja istraživanja u ovom području te ukazuje na potencijal primjene psiholoških značajki u kontekstu preventivnih strategija.

Ključne riječi: strojno učenje, logistička regresija, stablo odluke, slučajna šuma, XGBoost, SVM, osobine ličnosti, psihoaktivne supstance

Abstract

Risk assessment models for the consumption of psychoactive substances based on personality traits and machine learning methods

Jana Gazdek

This study explores the potential of assessing the risk of psychoactive substance use based on personality traits. A publicly available dataset was used, containing psychological and demographic characteristics of individuals, with a focus on the Five-Factor Model of personality. The aim was to build and evaluate several machine learning models, including logistic regression, decision tree, SVM, random forest and XGBoost, to predict substance use tendency.

The results suggest that traits such as sensation seeking and openness to experience may be linked to risky behavior. Although personality traits alone do not yield highly accurate predictive models, they offer valuable insights into behavioral patterns and could form a strong foundation for early risk detection when combined with other data sources. The study emphasizes interpretability and minimizing false negatives, and provides a basis for further research on the role of psychological characteristics in preventive strategies.

Keywords: machine learning, logistic regression, decision tree, random forest, XGBoost, SVM, personality types, psychoactive substances