Vještačka inteligencija

Ispitati efikasnost TabPFN na tipičnim tabelarnim dataset-ovim (klasifikacija)

Juni, 2025

Camović Melida
Rokša Amina
Ahmatović Hadija

Sadržaj

Faza 1:

Opis problema, osnovni pojmovi, korist rješavanja problema, kratak pregled postojećih dataset-ova

Faza 2:

Analiza trenutnog stanja problema, korištene metode vještačke inteligencije, postignuti rezultati

Faza 3:

Osnovni pregled izabranog dataset-a,metode pretprocesiranja podataka, potencijalni rizici

Faza 4:

Izbor tehnologija, treniranje i evaluacija modela, poređenje rezultata

Faza 5:

Osvrt na postignute rezultate, poređenje sa radovima iz prethodne faze

Opis problema

Faza 1

Cilj ovog rada je evaluacija efikasnosti **TabPFN modela** na tipičnim **tabelarnim dataset-ovima** za **klasifikacijske zadatke**. Fokus je na važnosti tabelarnih podataka kao osnovnog oblika podataka u realnim sistemima, te na njihovoj ulozi u treniranju modela vještačke inteligencije.

Rad obuhvata pregled najčešće korištenih metoda za klasifikaciju tabelarnih podataka, treniranje modela koristeći TabPFN i analizu rezultata.

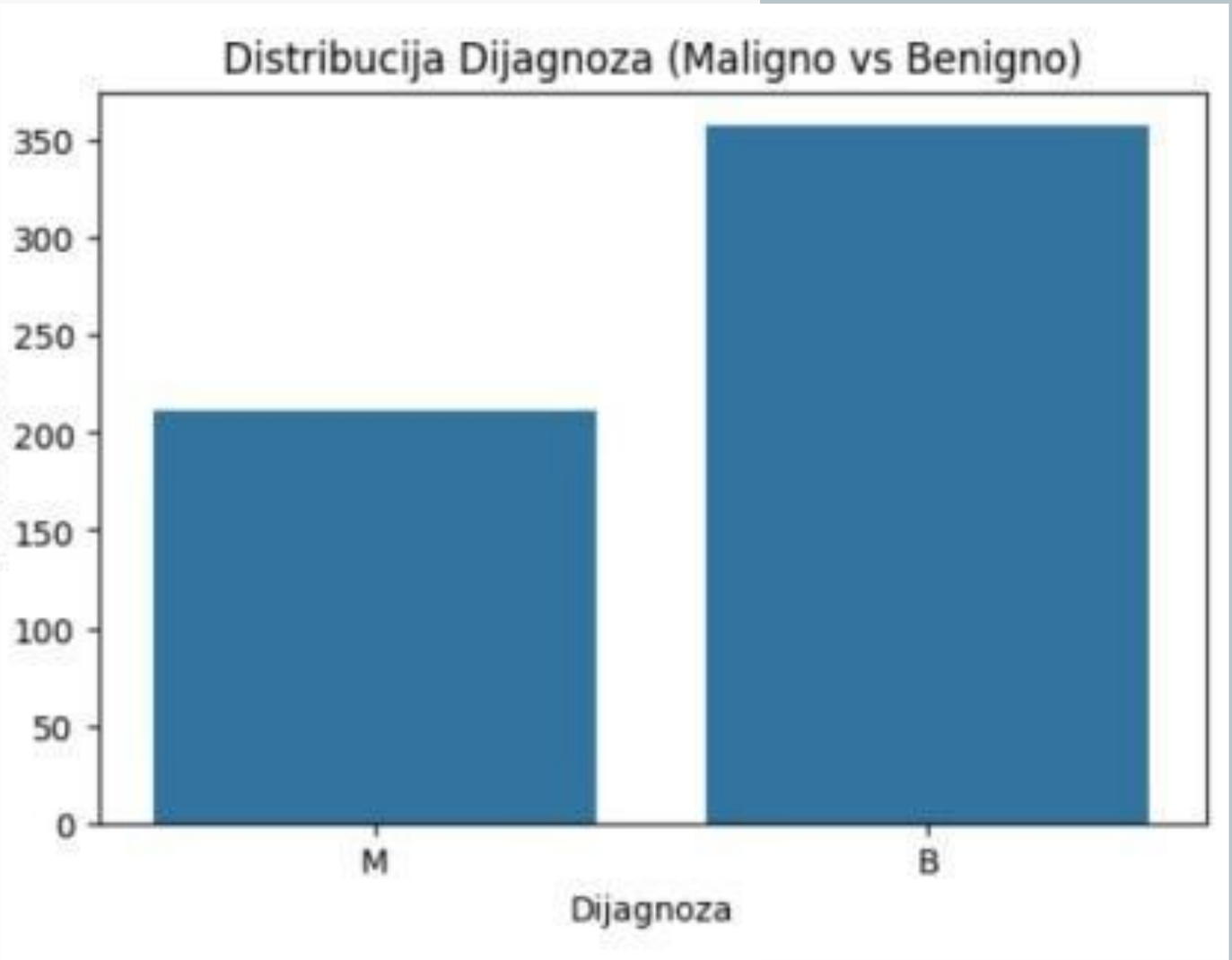
Posebna pažnja posvećena je specifičnostima i izazovima kvaliteta tabelarnih podataka, koji su često heterogeni i podložni greškama, što direktno utiče na pouzdanost modela.

Kratak pregled postojećih dataset-ova

Faza 1

Breast Cancer Wisconsin dataset

- Koristi se za predikciju da li je tumor dojke maligni ili benigni. To se određuje na osnovu različitih mjernih karakteristika dobijenih analizom ćelija biopsije.
- Kreiran je na osnovu digitalizovanih slika uzoraka tkiva dojke dovijenih finom iglenom aspiracijom.
- Sadrži ukupno 569 uzoraka gdje svaki uzorak predstavlja jednu pacijentkinju i njen nalaz biopsije.

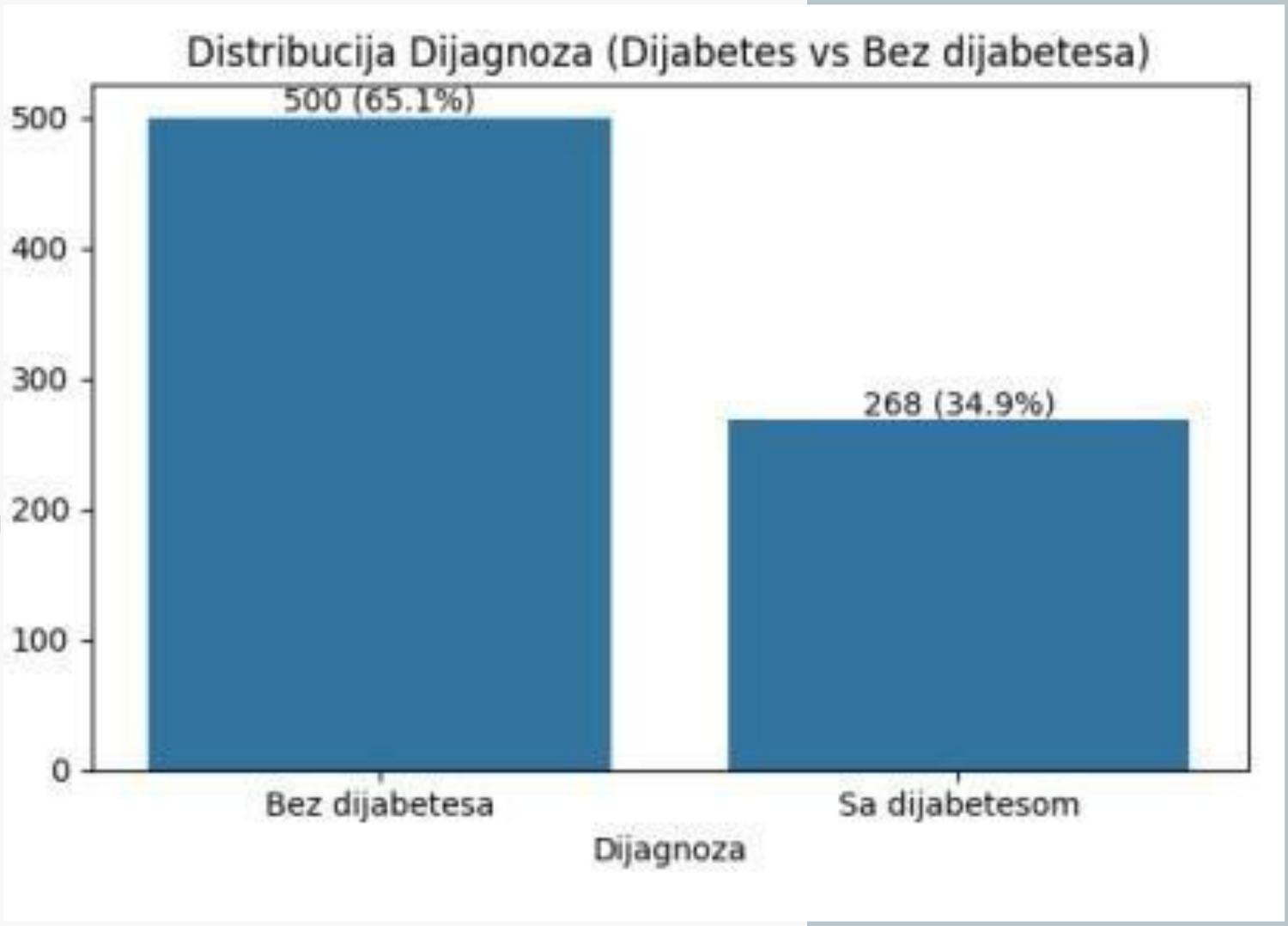


Kratak pregled postojećih dataset-ova

Faza 1

PRIMA Indian Diabetes Dataset

- Popularan skup podataka za klasifikaciju, često upotrebljavan za mašinsko učenje za dijagnozu dijabetesa.
- Skup podataka sadrži dijagnostičke informacije prikupljene od ženskih pacijentica starijih od 21 godinu, Prima Indian porijekla.
- Dataset obuhvata 768 pojedinačnih uzoraka, tako da svaki red predstavlja po jednu pacijenticu.

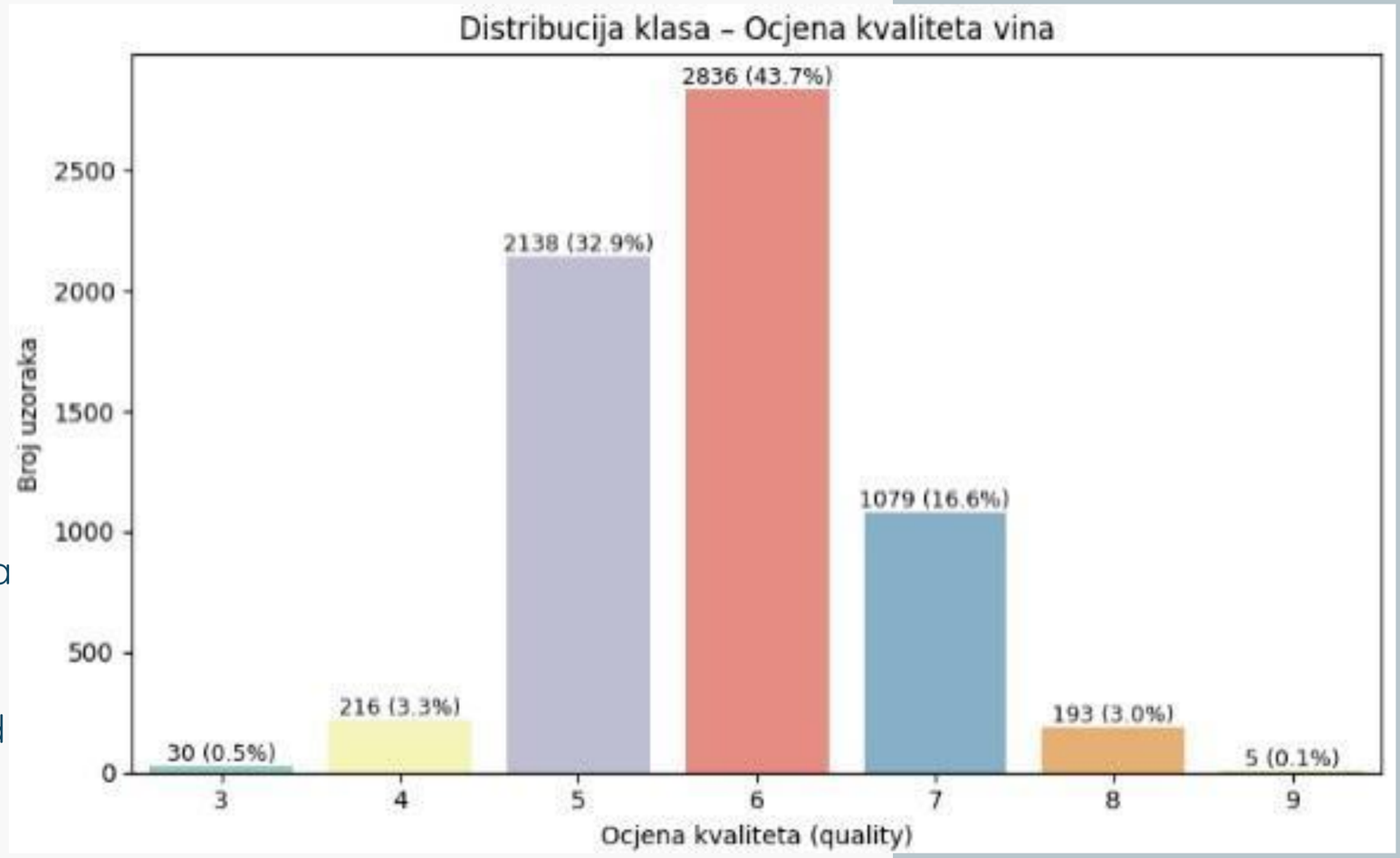


Kratak pregled postojećih dataset-ova

Faza 1

Wine Quality Dataset

- Javno dostupan skup podataka i koristi se za višeklasnu klasifikaciju. Upotrebljava se za predikciju kvaliteta vina na osnovu njegovih hemijskih karakteristika.
- Sastoji od dvije podvrste vina- crno i bijelo vino. Dobijen je spajanjem Red Wine Quality i White Wine Quality datasetov-a.
- Cilj je klasifikacija vina po kvalitetu sa ocjenama od 3 do 9.
- Broj instanci ovog dataset-a je 6497 uzoraka od čega 4898 bijelih i 1599 za crnih vina.

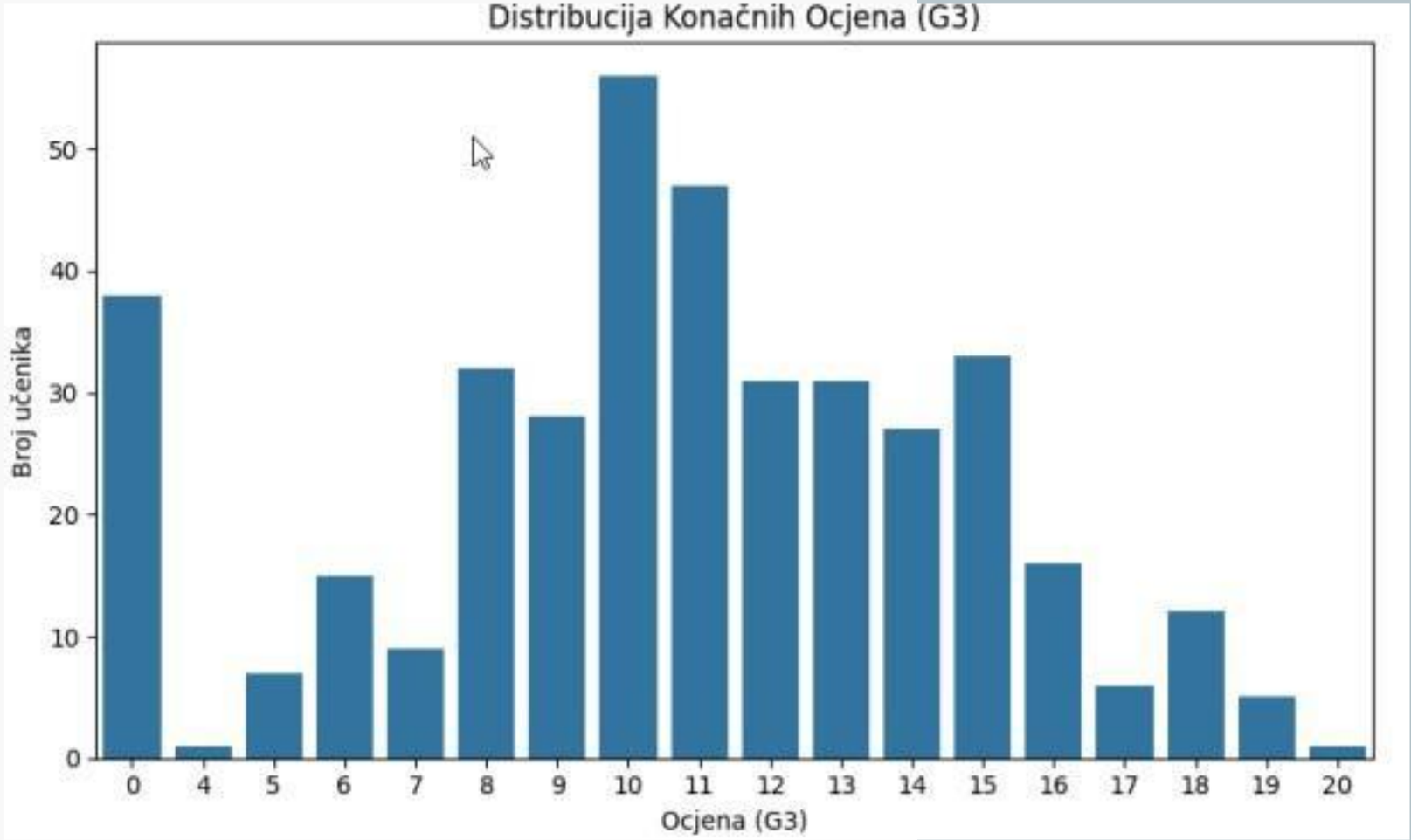


Kratak pregled postojećih dataset-ova

Faza 1

Student performance dataset

- Koristi se za analizu faktora koji utiču na školski uspjeh.
- Radi se o skupini podataka iz dvije portugalske srednje škole koji je objavljen s ciljem modeliranja performanski konkretno za matematiku i portugalski jezik.
- U sklopu dataseta su uključeni razni faktori čiji uticaj na uspjeh učenika želimo ispitati.



Pregled stanja u oblasti

Faza 2

“TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second”

- Transformer model treniran na milionima sintetičkih zadataka za brzu klasifikaciju malih tabularnih datasetova.
- Bez dodatnog treniranja, precizno generalizuje na neviđene podatke.
- Efikasan i brz, koristi GPT-2 stil arhitekture.

“Scaling TabPFN: Sketching and Feature Selection for Tabular Prior-Data Fitted Networks”

- Rješava problem skalabilnosti TabPFN-a na veće datasetove (10k+ instanci).
- Uvodi Sketching (sažetak podataka) i automatsku selekciju najvažnijih karakteristika.
- Smanjuje memorijske zahtjeve uz zadržavanje performansi.

“TabPFN Unleashed: A Scalable and Effective Solution to Tabular Classification Problems”

- Unaprijeđena verzija fokusirana na robustnost i skalabilnost.
- Koristi bagging i dinamički enkoder za adaptaciju na šum i neuravnotežene klase.
- Dokazana superiornost na preko 200 realnih datasetova.

Izbor i analiza dataset-a

Faza 3

Adult Incomne dataset, poznat i pod nazivom Census Incomne dataset

- Realni kontekst i izazovnost
- Mješoviti tipovi podataka
- Raspoloživost i obim podataka
- Distribucija klasa i balansiranje
- Jasna interepretacija rezultata

Jedan od najpoznatijih i često korištenih skupova podataka naročito kada je riječ o klasifikacionim problemima u oblasti društveno-ekonomskog predviđanja.

Klasifikacija koja se tiče ovog dataset-a omogućava primjenu različitih modela mašinskog učenja, pri čemu se na ovom projektu poseban akcenat stavlja na ispitivanje efikasnosti modela TabPFN-transformacijskog Bayesovog prediktora dizajniranog za rad sa tabelarnim podacima.

Izbor i analiza dataset-a

Faza 3

Dataset ukupno ima 32 561 reda tj.instance. Svaka instanca predstavlja demografske i ekonomske karakteristike jedne osobe.

→ U dataset-u se nalazi 14 ulaznih atributa i jedna ciljna varijabla **- income**.

Riječ o binarnoj klasifikaciji, tačnije cilj je predvidjeti da li osoba zarađuje:

- <=50K (manje ili jednako 50 000 USD godišnje)
- >50K (više od 50 000 USD godišnje)

Pregled atributa

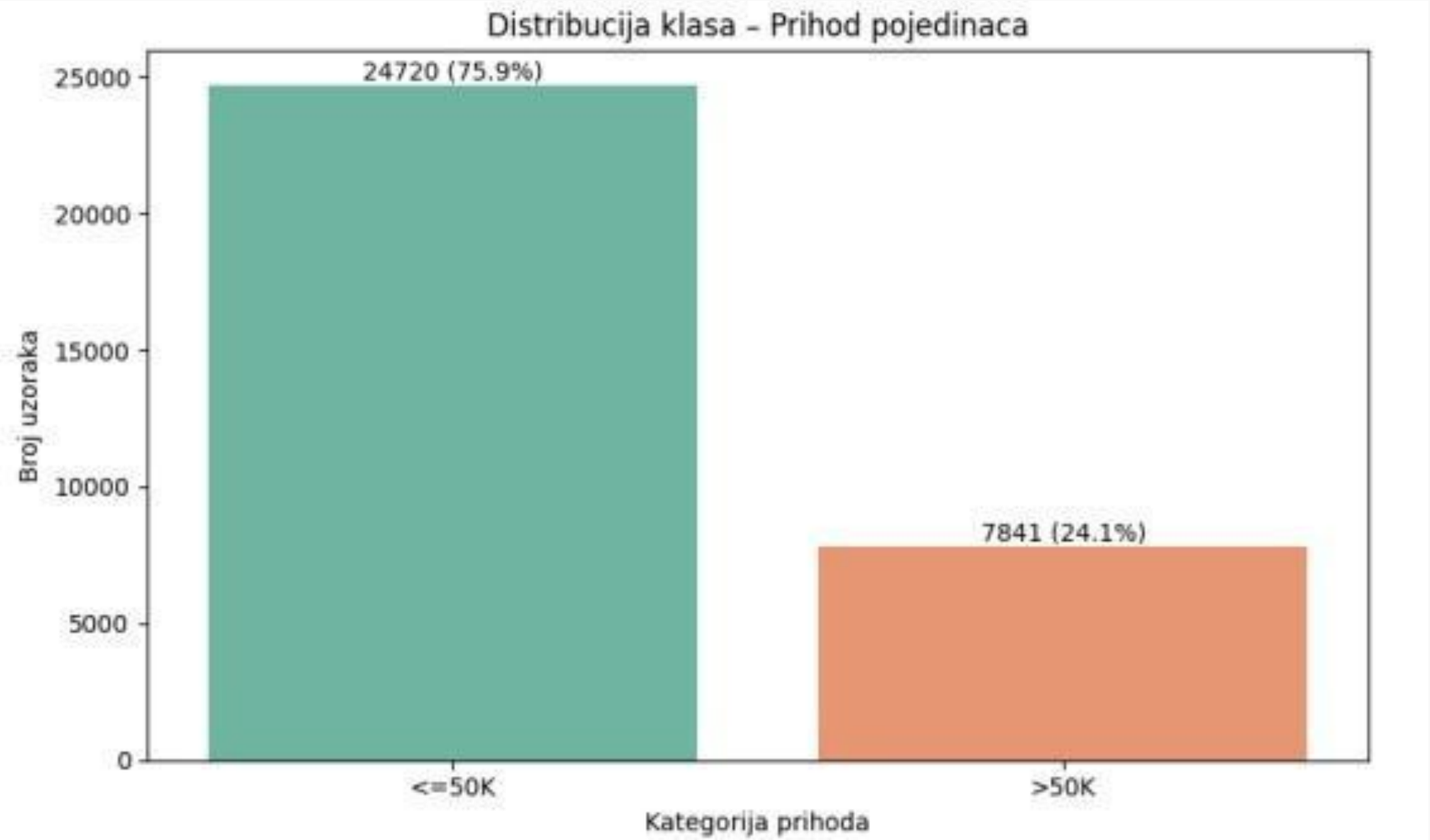
Faza 3

Atribut
age
workclass
fnlwgt
eduction
education.num
marital.status
occupation
relationship
race
sex
capital.gain
capital.loss
hours.per.week
native.country
income

- Starost osobe
- Tip poslodavca
- Statistički težinski faktor
- Nivo obrazobanja
- Brojčana reprezentacija nivoa obrazovanja
- Bračni status
- Vrsta zanimanja
- Povezanost sa domaćinstom
- Rasa
- Pol
- Kapitalna dobit u prethodnoj godini
- Kapitalni gubitak u prethodnoj godini
- Broj radnih sati nedeljno
- Zemlja porijekla
- Klasa prihoda

Analiza dataset-a

Faza 3

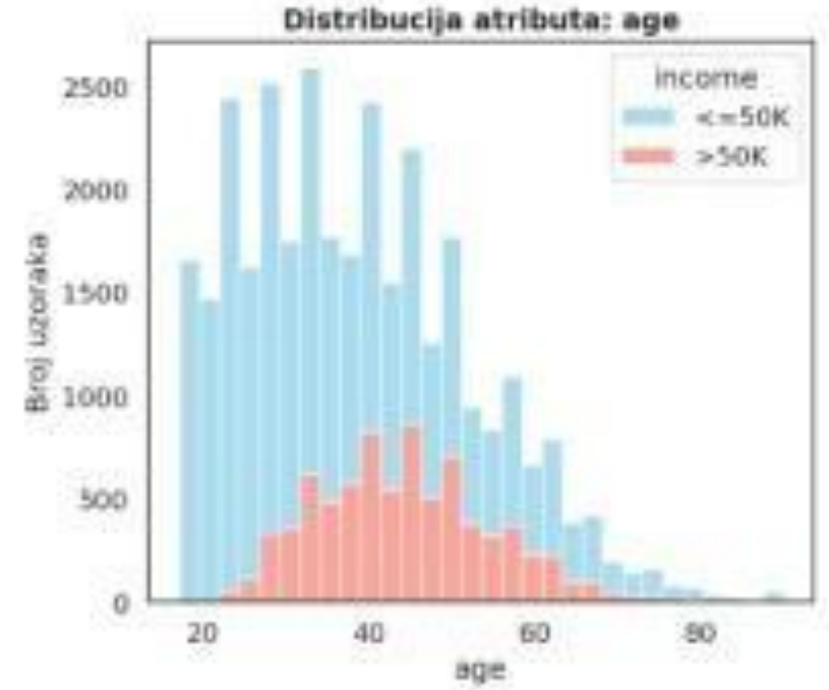


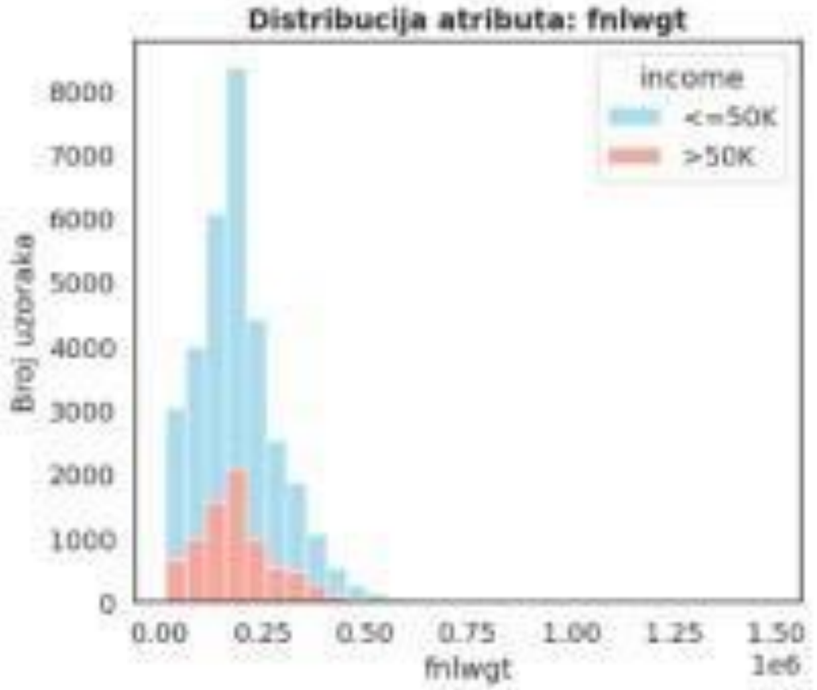
Klasa <=50:

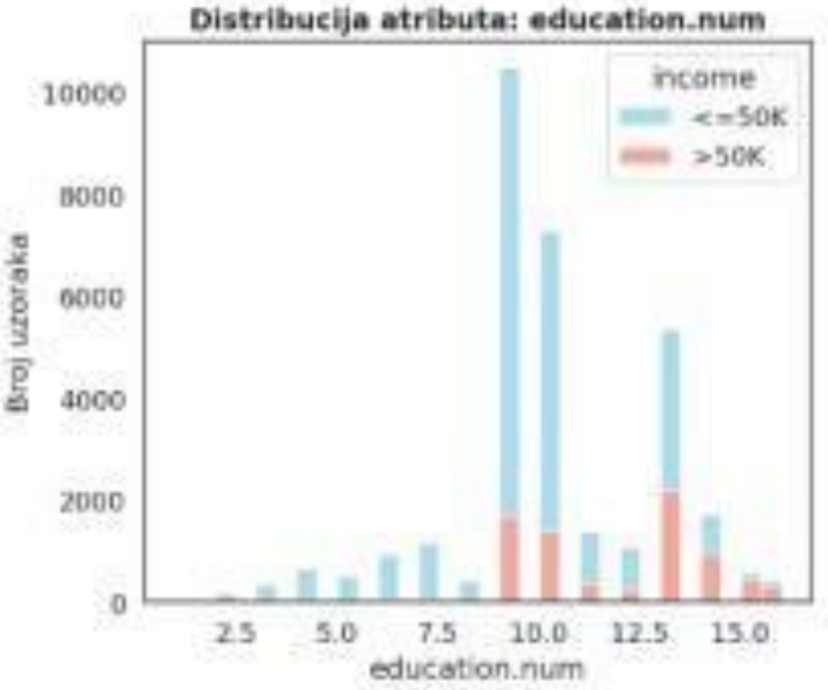
- Sadrži 24 720 instanci
- Čini 75.9% svih uzoraka
- Klasa je tri puta brojnija od klase >50K

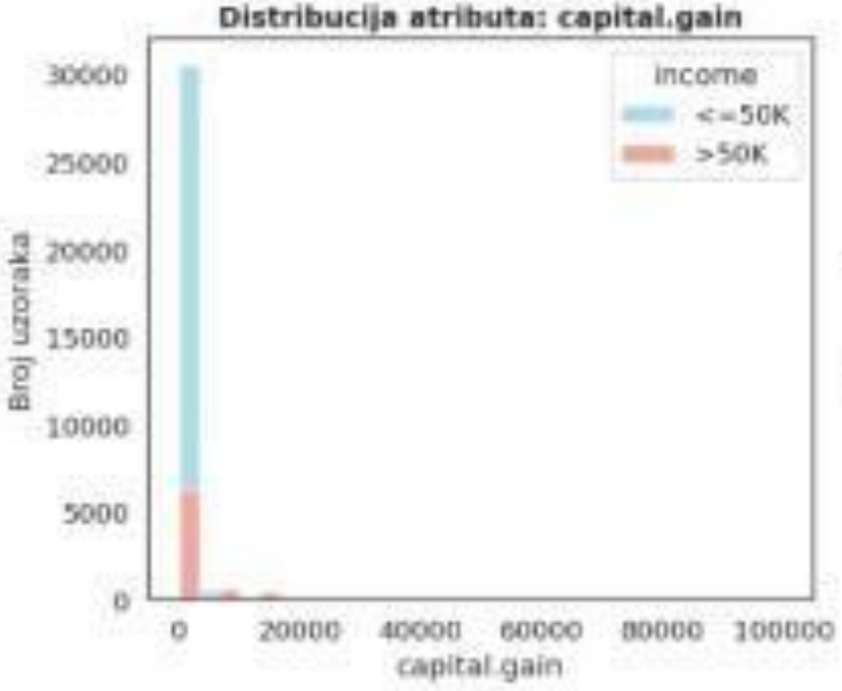
Klasa >50K:

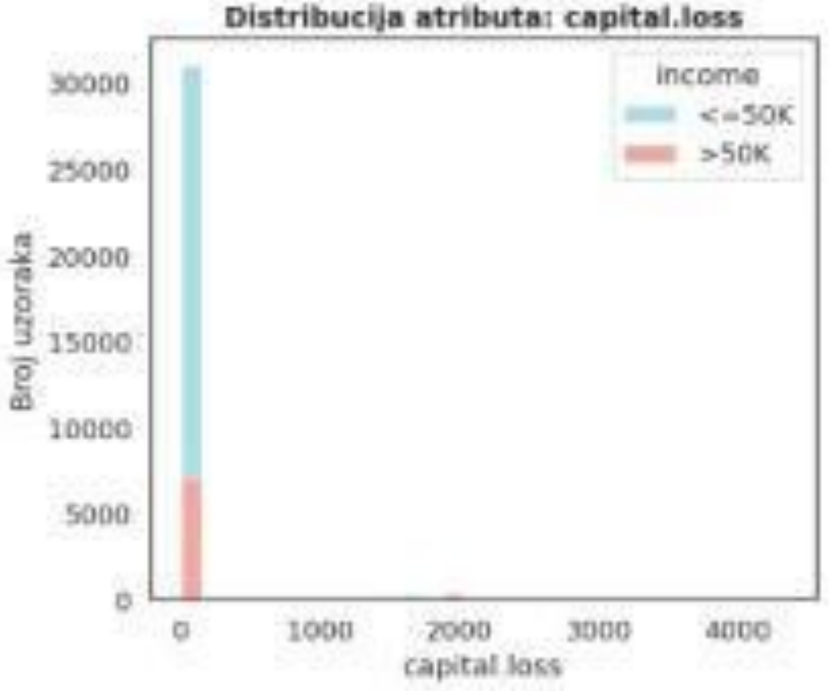
- Sadrži 7 841 instancu
- Čini 24.1% svih uzoraka
- Zauzima jednu trećinu ukupnih uzoraka

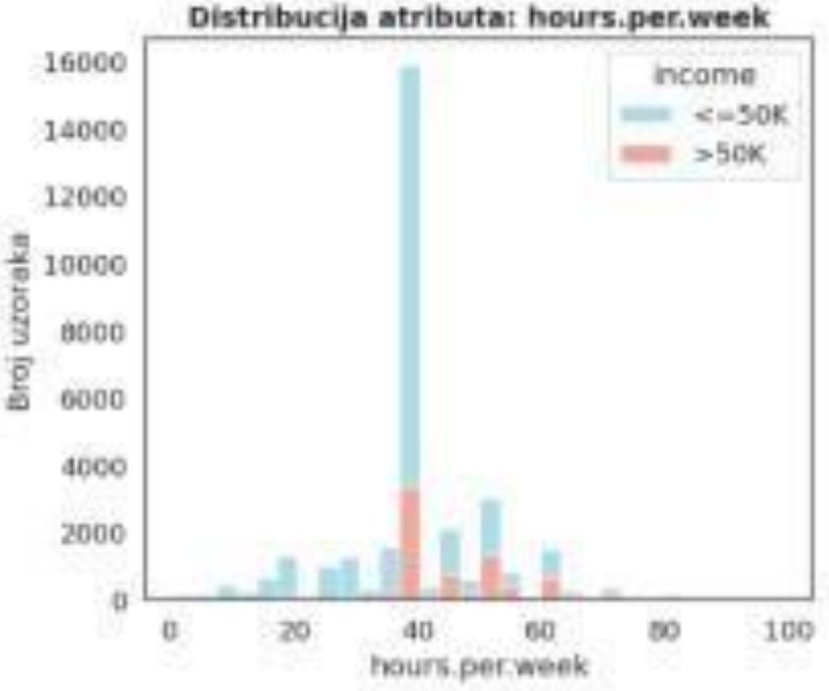

Distribucija atributa: age
income
<=50K
>50K
Broj uzoraka
age


Distribucija atributa: fnlwgt
income
<=50K
>50K
Broj uzoraka
fnlwgt
1e6


Distribucija atributa: education.num
income
<=50K
>50K
Broj uzoraka
education.num


Distribucija atributa: capital.gain
income
<=50K
>50K
Broj uzoraka
capital.gain


Distribucija atributa: capital.loss
income
<=50K
>50K
Broj uzoraka
capital.loss


Distribucija atributa: hours.per.week
income
<=50K
>50K
Broj uzoraka
hours.per.week

Osnovne metode pretprocesiranja

Faza 3

1. Rukovanje nedostajućim vrijednostima

```
import pandas as pd

# Učitavanje CSV fajl-a
df = pd.read_csv('adult.csv')

# Prikaz vrijednosti '?' u atributima u kojima ih ima
print("Broj '?' po kolonama:")
print((df == '?').sum())

# Kolone koje sadrže '?' kao nedostajuće vrijednosti
missing_value_cols = ['workclass', 'occupation', 'native.country']

# Zamjena '?' sa 'Unknown'
for col in missing_value_cols:
    df[col] = df[col].replace('?', 'Unknown')

# Provjera da li su sve '?' uspješno zamijenjene
print("\nPreostale '?' vrijednosti:")
print((df == '?').sum())
```

Zamjena nedostajućih vrijednosti vrijednošću ‘Unknown’

Osnovne metode pretprocesiranja

Faza 3

2. Enkodiranje kategorijskih atributa

- Label encoding
- One-Hot Encoding
- Ordinal Encoding

```
# Kategorijske kolone (bez ciljne varijable)
categorical_cols = [
    'workclass',
    'marital.status',
    'occupation',
    'relationship',
    'race',
    'sex',
    'native.country'
]

# Label encoding za kategorijske kolone
label_encoders = {}
for col in categorical_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    label_encoders[col] = le  # čuvamo enkoder ako bude trebalo za dekodiranje

# Provjera
df[categorical_cols].head()
```

	workclass	marital.status	occupation	relationship	race	sex	native.country
0	7	6	14	1	4	0	38
1	3	6	3	1	4	0	38
2	7	6	14	4	2	0	38
3	3	0	6	4	4	0	38
4	3	5	9	3	4	0	38

Osnovne metode pretprocesiranja

Faza 3

3. Uklanjanje atributa

Atribut fnlwgt (final weight) ima statički značaj i kao takav nije dobar prediktor ciljne varijable.

Atribut eduction se uklanja iz razloga što bi ga trebalo enkodirati sa Ordinal Endocing i u tom slučaju bi imali dva praktično ista atributa.

```
df.drop(columns=['education', 'fnlwgt'], inplace=True)
print(df.columns)
```

```
Index(['age', 'workclass', 'education.num', 'marital.status', 'occupation',
       'relationship', 'race', 'sex', 'capital.gain', 'capital.loss',
       'hours.per.week', 'native.country', 'income'],
      dtype='object')
```

Osnovne metode pretprocesiranja

Faza 3

4. Enkodiranje ciljne varijable income

Ciljna varijabla je tekstualnog formata (<=50K, >50K). Za potrebe binarna klasifikacije potrebno ju je enkodirati.

```
from sklearn.preprocessing import LabelEncoder
# Enkodiranje ciljne varijable
le_target = LabelEncoder()
df['income'] = le_target.fit_transform(df['income'])
# Prikaz mapiranja vrijednosti
print("Mapiranje klasnih oznaka:")
for original, encoded in zip(le_target.classes_, le_target.transform(le_target.classes_)):
    print(f"{original} → {encoded}")
```

```
Mapiranje klasnih oznaka:
<=50K → 0
>50K → 1
```

Napredna pretprocesiranja

Faza 3

- Grupisanje rijetkih kategorija
- Standardizacija numeričkih klasa
- Log-transformacija visoko asimetričnih atributa
- Obrada outljera
- Klasni disbalans

Potencijalni rizici

Faza 3

- Disbalans klasa
- Gubitak informacija kod zamjene ‘?’ sa ‘Unknown’
- Pitanje semantičke tačnosti Label Encoding-a
- Nelinearnost distribucije pojedinih atributa
- Utjecaj veličine uzorka

Priprema podataka

Faza 4

Pripremljeni su podaci u tri različite veličine podskupova: 10 000, 5 000 i 500 instanci.

```
# Stratifikovano uzorkovanje iz cijelog df
df_10000, _ = train_test_split(
    df,
    train_size=10000,
    stratify=df['income'],
    random_state=10000
)

X_10000 = df_10000.drop(columns=['income'])
y_10000 = df_10000['income']

X_10000_np = X_10000.to_numpy().astype('float32')
y_10000_np = y_10000.to_numpy().astype('int64')

X_train_10000, X_test_10000, y_train_10000, y_test_10000 = train_test_split(
    X_10000_np,
    y_10000_np,
    test_size=0.2,
    stratify=y_10000_np,
    random_state=42
```

Treniranje i evaluacija modela

Faza 4

Ovaj model koristi **transformersku** arhitekturu što znači da se temelji na dubokoj neuronskoj mreži.

```
device = 'cuda' if torch.cuda.is_available() else 'cpu'
print(f"Koristi se uređaj: {device}")

# 1. Učitavanje unaprijed treniranog TabPFN modela
model = TabPFNClassifier(device=device)
model.fit(X_train_10000, y_train_10000)

# 2. Predikcija nad test skupom
y_pred_10000 = model.predict(X_test_10000)

# 3. Evaluacija modela
accuracy = accuracy_score(y_test_10000, y_pred_10000)
print(f"\n Tačnost (accuracy) modela na test skupu (10k uzorak): {accuracy:.4f}\n")

# Detaljan izvještaj
print("Klasifikacioni izvještaj:\n")
print(classification_report(y_test_10000, y_pred_10000, target_names=['<=50K', '>50K']))
```

Kod za treniranje i evaluaciju modela

Korištene metrike

Faza 4

- Precision
- Recall
- F1-score
- Support

Tačnost (accuracy) modela na test skupu (10k uzorak): 0.8625

Klasifikacioni izvještaj:

	precision	recall	f1-score	support
<=50K	0.88	0.94	0.91	1518
>50K	0.77	0.61	0.68	482
accuracy			0.86	2000
macro avg	0.83	0.78	0.80	2000
weighted avg	0.86	0.86	0.86	2000

Rezultati klasifikacije TabPFN modela na 10 000 instanci

Faza 4

Poređenje rezultata

Tačnost (accuracy) modela na test skupu (5k uzorak): 0.8530

Klasifikacioni izvještaj:

	precision	recall	f1-score	support
<=50K	0.88	0.93	0.91	759
>50K	0.74	0.61	0.67	241
accuracy			0.85	1000
macro avg	0.81	0.77	0.79	1000
weighted avg	0.85	0.85	0.85	1000

Rezultati klasifikacije TabPFN modela na 5 000 instanci

Tačnost (accuracy) Random Forest modela na test skupu (5.000 uzorak): 0.8330

Klasifikacioni izvještaj za Random Forest:

	precision	recall	f1-score	support
<=50K	0.86	0.93	0.89	759
>50K	0.70	0.53	0.61	241
accuracy			0.83	1000
macro avg	0.78	0.73	0.75	1000
weighted avg	0.82	0.83	0.82	1000

Rezultati klasifikacije RandomForest modela na 5 000 instanci

Različiti podskupovi Random Forest model

Faza 4

Poređenje rezultata

Tačnost (accuracy) modela na test skupu (500 uzorak): 0.8800

Klasifikacioni izvještaj:

	precision	recall	f1-score	support
<=50K	0.89	0.96	0.92	76
>50K	0.83	0.62	0.71	24
accuracy			0.88	100
macro avg	0.86	0.79	0.82	100
weighted avg	0.88	0.88	0.87	100

Rezultati klasifikacije TabPFN modela na 500 instanci

Tačnost (accuracy) Random Forest modela na test skupu (500 uzorak): 0.9000

Klasifikacioni izvještaj za Random Forest:

	precision	recall	f1-score	support
<=50K	0.90	0.97	0.94	76
>50K	0.89	0.67	0.76	24
accuracy			0.90	100
macro avg	0.90	0.82	0.85	100
weighted avg	0.90	0.90	0.89	100

Rezultati klasifikacije RandomForest modela na 500 instanci

Različiti podskupovi Random Forest model

Analiza rezultata

Faza 4

→ TabPFN model je dostigao tačnost od 83.33% na testnom skupu

```
Koristi se uređaj: cuda

Tačnost modela (1500 uzoraka): 0.8333

Klasifikacioni izvještaj:

              precision    recall  f1-score   support

Premium User       0.67      0.04      0.07        51
   Free User       0.84      1.00      0.91       249

    accuracy                           0.83       300
   macro avg       0.75      0.52      0.49       300
weighted avg       0.81      0.83      0.77       300
```

Rezultat klasifikacije dodatno potvrđuje da TabPFN nije prilagođen radu sa neuravnoteženim podacima

Cjelokupni osvrt

Model ima dobre performanse u rješavanju problema klasifikacije tabelarnih podataka

Ima izazove sa nebalansiranim dataset-ovima i prioritetizira dominantnije klase

Kao unaprjeđenje ovog rada bi mogli dodati

Faza 5

- **Testiranje na većem skupu podataka**
- **Poređenje TabPFN v1 i TabPFN v2 modela**
- **Korištenje datasetova kojim nedostaju mnogi podaci**
- **Klasifikacija balansiranih klasa**

Zaključak

Faza 5

TabPFN je napredan alat za klasifikaciju tabelarnih podataka. U određenim scenarijima je pokazao bolje performanse od tradicionalnih modela klasifikacije poput Random Forest-a. Najistaknutija karakteristika TabPFN-a je upravno in-context učenje koje eliminiše potrebu za dodatnim treniranjem modela.

Hvala na pažnji!