UNIVERZITET U SARAJEVU
ELEKTROTEHNIČKI FAKULTET
ODSJEK ZA INFORMATIKU I RAČUNARSTVO

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

VJEŠTAČKA INTELIGENCIJA
- PRVI CIKLUS STUDIJA -

Pripremili: Camović Melida, Rokša Amina i Ahmatović Hadija
Supervizor: Bajrić Amina
Predmetni nastavnik: Vanr. prof. dr Amila Akagić

Faza 1: Izbor teme i opis problema

1.1 Opis problema

Cilj ovog seminarskog rada je evaluacija efikasnosti TabPFN na tipičnim tabelarnim dataset-ovima za rješavanje problema klasifikacije. Fokusirati ćemo se na značaj tabelarnih podataka za treniranje modela vještačke inteligencije. Izvršiti ćemo pregled trenutno najzastupljenijih metoda klasifikacije tabelarnih podataka, izvršiti treniranje našeg modela pomoću TabPFN-a i analizirati dobivene rezultate. Radi toga najprije trebamo razujmeti svrhu tabelarnih podataka i po čemu se razlikuju od drugih tipova podataka

1.2 Značaj tabelarnih podataka

Podaci imaju ključnu ulogu u oblasti vještačke inteligencije, jer predstavljaju osnovu za treniranje modela mašinskog učenja. U stvarnim sistemima često imamo podatke predstavljene u tabelarnom obliku, što je posljedica velike upotrebe relacijskih baza koje u tabelama sa jasno definisanim redovima i kolonama zapisuju i čuvaju podatke.

Kvalitet dataset-a direktno utiče na pouzdanost donesenih odluka i prepoznavanja obrazaca modela. [1] Dataset niskog kvaliteta može izazvati lančane greške koje se prenose kroz cijeli sistem. Greške u podacima često nastaju prilikom samog prikupljanja podataka.

Jedan od ilustrativnih primjera tog problema se nalazi u zdravstvenoj industriji. Prema nedavnim procjenama, približno 30% ukupne količine podataka u svijetu generira zdravstvena industrija. [2] Količina podataka ne obećava visok kvalitet, a samim time i upotrebljivost istih. Greške u mjerenju podataka, različiti akronimi za iste dijagnoze i korištenje zastarjelih podataka dovode do gubitka konzistentnost, što dovodi do toga da naš VI model, treniran na tim podacima, može propustiti šablone koje nastojimo otkriti. Tokom pandemije COVID-19 razvijen je velik broj VI alata za detekciju virusa, ali se pokazalo da većina ima ograničenu upotrebljivost upravo zbog lošeg kvaliteta podataka dostupnih za treniranje. [3]

Tabelarni podaci se razlikuju od drugih tipova modela po tome što često obuhvataju podatke raznih tipova - od numeričkih do korisnički definisanih i binarnih vrijednosti. Zbog toga će biti potrebno prilikom pretprocesiranja prilagoditi ove podatke kontekstu modela.

1.3 Definicije osnovnih pojmova

Tabelarni podaci su struktuirani podaci organizovani u redove i kolone, gdje svaka kolona predstavlja određenu varijablu (atribut), a svaki red jednu instancu. Zbog svoje forme, ovi podaci su čitljivi i lako ih je uporediti.

Klasifikacija je oblik nadziranog učenja kod kojeg model uči iz označenih podataka kako bi kasnije mogao da dodjeljuje nove, neoznačene primjere u jednu od unaprijed definisanih klasa.

TabPFN je neuronski model koji pokušava iskoristiti snagu neuronskih mreža upravo na stuktuiranim tabelarnim podacima tako što na osnovu učenja na treniranom modelu, omogućava trenutno izvođenje klasifikacije bez potrebe za dodatnim treniranjem ili podešavanjem hiperparametara.

1.4 Kratak pregled postojećih dataset-ova

U nastavku ćemo izvršiti analizu nekoliko javno dostupnih dataset-ova koji se koriste prilikom rješavanja problema klasifikacije.

1.4.1 Breast Cancer Wisconsin dataset

Breast Cancer Wisconsin dataset jedan od najpoznatijih i često korištenih dataset-ova u oblasti mašinskog učenja i biomedicinske klasifikacije. 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 dobijenih finom iglenom aspiracijom.
Dostupan je u okviru *sklearn.dataset* bibliotke (*load_breast_cancer()*).
Takođe , moguće ga je pronaći na *UCI Machine Learning Repository* i *Kaggle*. Na Kaggle dostupan je putem sljedećeg linka:
https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data
Dataset sadrži ukupno 569 uzoraka gdje svaki uzorak predstavlja jednu pacijentkinju i njen nalaz biopsije. Dataset ima ukupno 32 kolone koje su raspoređene na sljedeći način:

Redni broj	**Ime kolone**	**Opis**
1	id	Jedinstveni identifikator uzorka
2	diagnosis	Klasa tumora(M ili B)
3-32	30 numeričkih atributa	Opis karakteritsika ćelija

Numerički atributi dobijeni su na osnovu slika i odnose se na različite statističke karakteristike 10 osnovnih osobina ćelije.
Osnovne osobine ćelije:

1. **Poluprečnik** *(radius)* - srednja vrijednost udaljenosti od centra do tačaka na obodu ćelije

2. **Tekstura** *(texture)* - standardna devijacija sivih tonova (intenziteta piksela)
3. **Obim** *(perimeter)*- obim ćelije
4. **Površina** *(area)*- površina jedra ćelije
5. **Glatkoća** *(smoothness)*
6. **Kompaktnost** *(compactness)*- mjera zbijenosti (perimeter^2 / area - 1.0)
7. **Konkavnost** *(concavity)* - dubina udubljenja u konturi ćelije
8. **Konkavne tačke** *(concave points)*- broj konkavnih tačaka na konturi
9. **Simterija** *(symmetry)*- odnos simterije oblika ćelije
10. **Fraktalna dimenzija** *(fractal dimension)*- mjera složenosti ivice ("coastline approximation" - 1)

Za svaku od navednih karakteristika izračunate su sljedeće tri vrijednosti:

- Srednja vrijednost *(mean)*
- Standardna greška *(standard error)*
- "Najgora" ili najveća vrijednost *(worst)*- prosjek tri najveće vrijedosti za tu karakteristiku

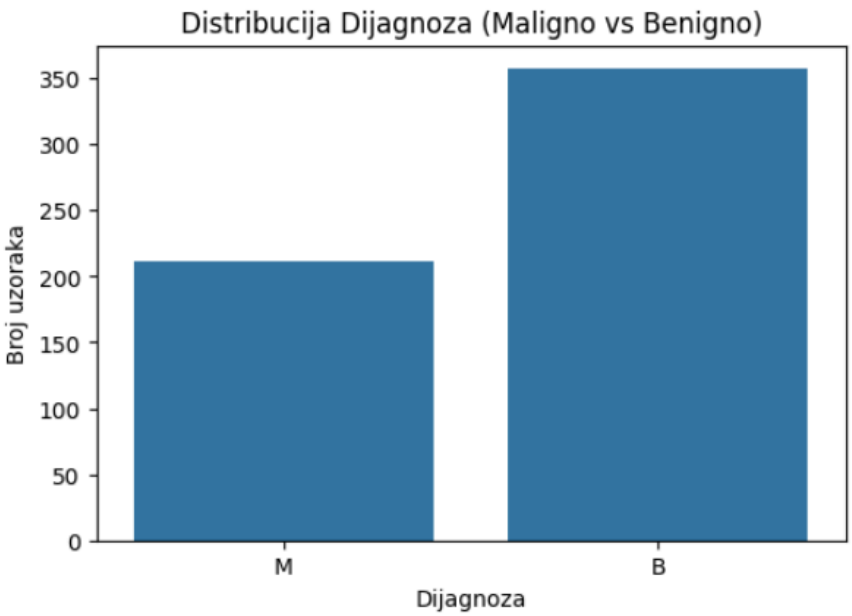
Sve ovo rezultira sa 30 vrijednosti za svaku instancu. Za instancu, polje 3 predstavlja srednju vrijednost poluprečnika (*Mean Radius*), polje 13 standardnu grešku poluprečnika (*Radius SE*), a polje 23 predstavlja najgoru vrijednost poluprečnika (*Worst Radius*).[4]
Svi atributi su kodirani sa 4 cifre.
Nema nedostajućih vrijednosti.
Raspodjela klasa: Benigni tumor - 357 uzoraka
Maligni tumor- 212 uzoraka
Dataset je detaljno analiziran u našem Analiza_predloženih_datasetova.ipynb [5], gdje imamo tablerni prikaz i vizualizacije o kojima će biti riječi u nastavku.

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...	radius_worst	texture_worst	perimeter_worst	area_worst	smooth
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.30010	0.14710	...	25.380	17.33	184.60	2019.0	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.08690	0.07017	...	24.990	23.41	158.80	1956.0	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.19740	0.12790	...	23.570	25.53	152.50	1709.0	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.24140	0.10520	...	14.910	26.50	98.87	567.7	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.19800	0.10430	...	22.540	16.67	152.20	1575.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	...	25.450	26.40	166.10	2027.0	
565	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	...	23.690	38.25	155.00	1731.0	
566	926954	M	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	...	18.980	34.12	126.70	1124.0	
567	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	...	25.740	39.42	184.60	1821.0	
568	92751	B	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	...	9.456	30.37	59.16	268.6	

569 rows × 32 columns

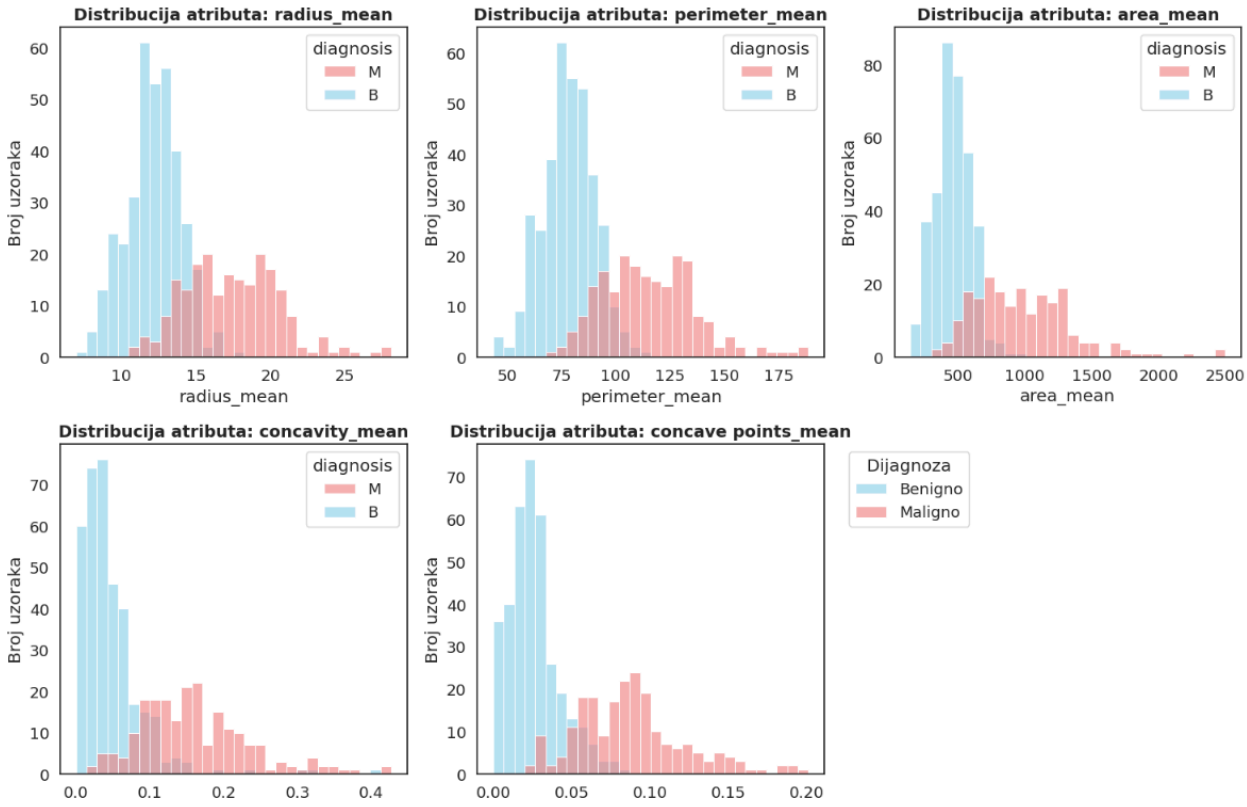
Slika 1: Tabelarni prikaz Brest Cancer dataset-a

Kao što se može vidjeti na slici Breast Cancer Wisconsin biblioteke ima 589 redova i 31 kolonu. Po današnjim standardima Breast Cancer Wisconsin spada u male datasetove. Većina modela ga može lako trenirati.



Slika 2: Dijagram distirbucije ciljne varijable Breast Cancer dataset-a

Grafički prikaz distribucije klasa u Breast Cancer Wisconsin prikazan je na slici iznad i prikazuje odnos između dvije katerogije dijagnoza tumora. Iz prikazanog grafa se može zaključiti da dataset nije potpuno balansiran. Može se primjeiti veći broj benignih slučajeva. Ovo može rezultirati pristrasnošću modela ka dominatnoj klasi. Međutim, umjeren disbalans klasa ne predstavlja veliki problem za TabPFN. TabPFN je obučen na velikom broju sintetičkih podataka, što mu omogućava robusnost na umrenu neravnotežu klasa, zahvaljujući sposobnosti da aproksimira Bayesovsko zaključivanje.



Slika 3: Histogrami bitnih atributa

Na slici iznad je grafički prikaz bitnih atributa u Breast Cancer dataset-u.
Na osnovu slike moguće je primjetiti značajne razlike u domenu i rasponu vijrednosti između atributa. Pa tako imamo da je atribut *area_mean* na rasponu od 0 do 2500 i atribut concave points_mean u rasponu između 0 i 0,4.
Vrijednosti atributa u slučaju benignih oboljenja su manje što se vidi na grafiku s obzirom na veću koncentraciju plavih stubaca u lijevom dijelu x ose.
Pokrivenost vrijednostima je solidna, tačnije postoje podaci na cijelom opsegu ali evidentno sa manjom gustinom na početku i kraju opsega.
Takođe, podaci za dijagnozu malignih i benignih tumora se dosta razlikuju što će olakšati učenje.

1.4.2 PRIMA Indian Diabetes Dataset

Prima Indian Diabetes Dataset je popularan skup podataka za klasifikaciju, često upotrebljavan za mašinsko učenje za dijagnozu dijabetesa. Napravljen je od strane National Institute of Diabetes and Digetsive and Kindey Diseases.
Dostupan je na Kaggle u okviru UCI Machine Learning repozitorija, putem sljedećeg linka: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
Format fajla je CSV, a veličina fajla je 24KB.
Skup podataka sadrži dijagnostičke infrmacije prikupljene od ženskih pacijentica starijih od 21 godinu, Prima Indian porijekla.[6]
Cilj je dijagnostikovati da li pacijentica ima dijabetes na osnovu podataka iz dataset-a.
Dataset obuhvata 768 pojedinačnih uzoraka, tako da svaki red predstavlja po jednu pacijenticu. Svaki uzorak je definisan sa 8 različiti atributa odnosno kolna, uz dodatnu kolonu koja sadrži ciljnu varijablu tj. Ishod dijagnoze. Ova veličina dataset-a se je pogodna za TabPFN, s obzirom da je on dizajniran za tabelarne podatke podatke male do srednje veličine.
Svi atributi su numeričke prirode, što je svakako pogodno.

Atrbuti ovog dataset-a su:

1. *Pregnancies*- odnosi se na broj trudnoća pacijentkinje
2. *Glucose*- koncentracija glukoze u krvi
3. *BloodPressure*- dijastolni krvni pritisak (mm Hg)
4. *SkinThickness*- debljina kožnog nabora (mm)
5. *Insulin*- nivo insulina (mm U/ml)
6. *BMI*- indeks telesne mase
7. *DiabetesPedigreeFunction*- genetski faktor
8. *Age*- godine pacijentkinje
9. *Outcome*- **ciljna varijabla** koja označava dijagnozu

Varijabla outcome ima vrijednost 1 u slučaju prisustva dijabetesa, a 0 u slučaju odsustva što ovaj problem čini binarnim klasifikacionim problemom.
Izazov kod ovog datasets jesu implicitne nedostajuće vrijednosti. Ovaj dataset ne sadrži eksplicitne NaN vrijednosti ali neke kolone imaju vrijednost 0 iako je to biloški nemoguće. To se odnosi na atribute: glukoza (*Glucose*), krvni pritisak (*BloodPressure*), debljina kože (*SkinThickness*), insulin (*Insulin*) i BMI.

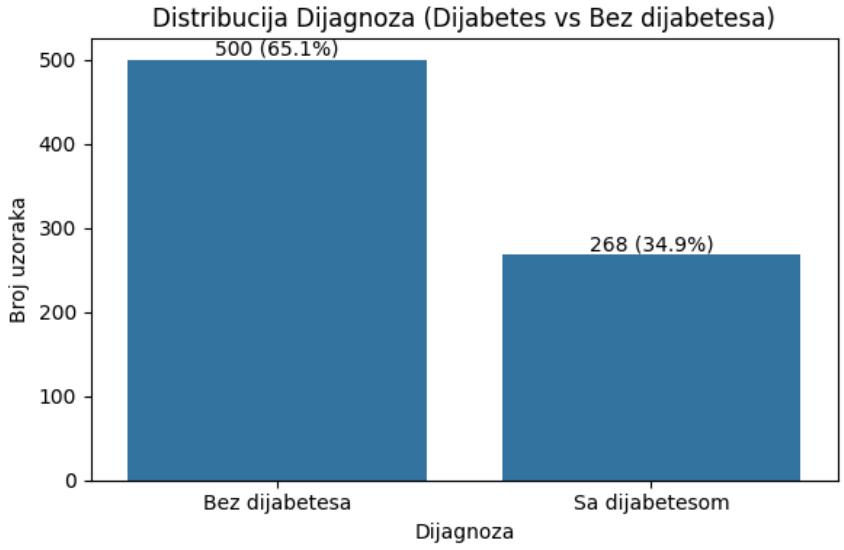
Raspodjela klasa: Sa dijabetesom- 500 uzoraka
Bez dijabetesa- 268 uzoraka
U našem Analiza_predloženih_datasetova.ipynb [7] izvršena je analiza dataset-a i ukviru njega dostupan je tabelarni prikaz i grafici o kojima će u nastavku biti riječi.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

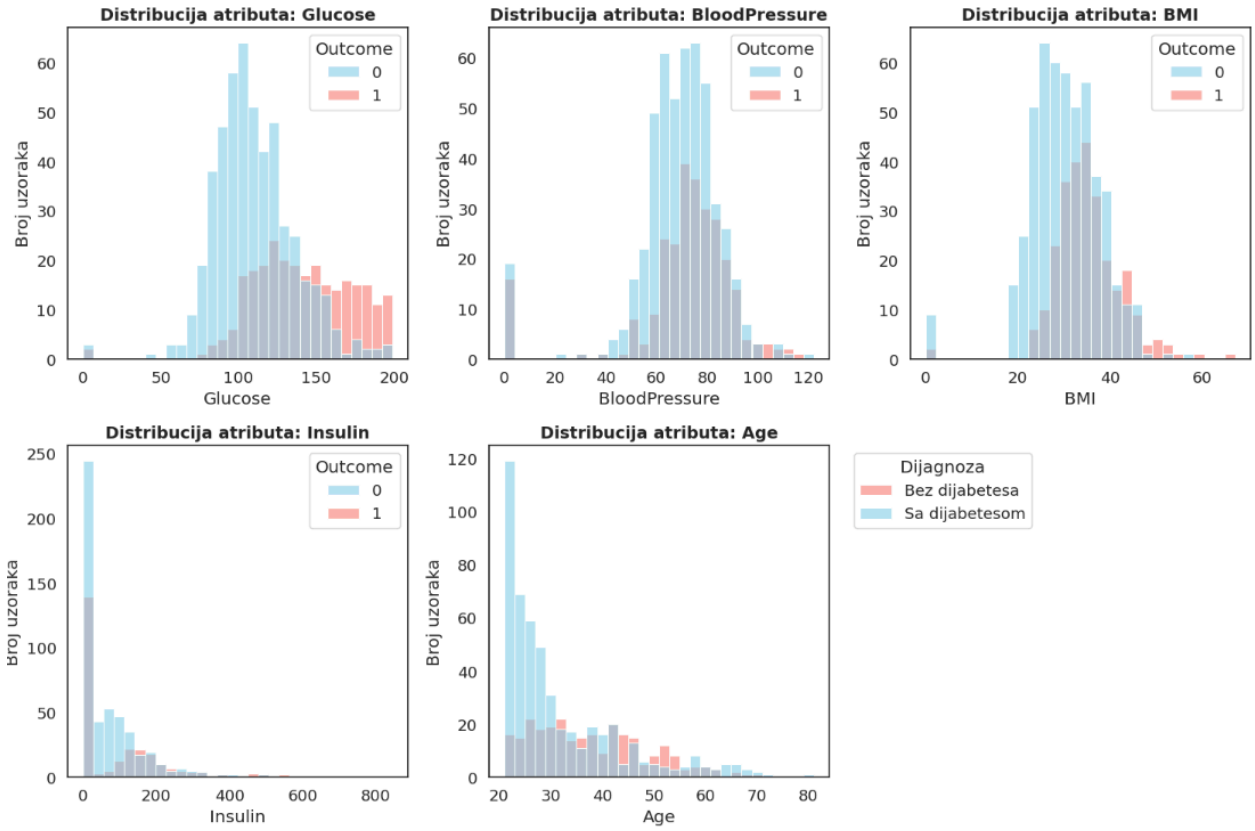
Slika 4: Tabelarni prikaz Prima Indians Diabetes dataset-a

Kao što se može videti na slici iznad, dataset sadrži 768 redova sa po 9 kolona. Već na slici su primjetne implicitne NaN vrijednosti u obliku nule, kao u primjeru 764. reda u koloni *Insulin*.



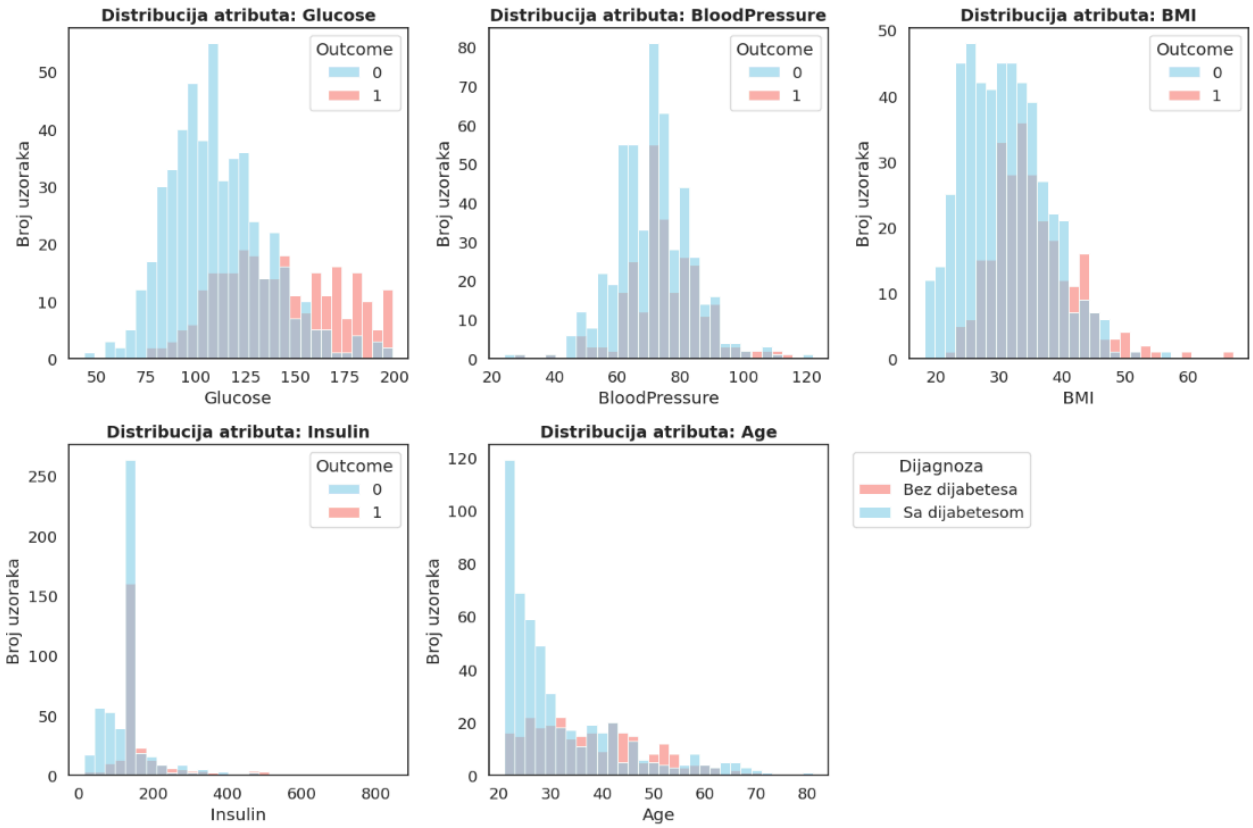
Slika 5: Dijagram distirbucije ciljne varijable Prima Indians Diabetes dataset-a

Na osnovu dijagrama sa slike iznad se može zaključiti da distibucija klasa ima blagi disbalans, jer je klasa 0 značajno zastupljenijia (65.1%). Ipak ovaj disbalans nije ekstreman i podogan je za binarnu klasifikaciju bez potrebe za agresivnim tehnikama balansiranja. TabPFN model svakako nije osetljiv na ovaj problem i pokauje robusnost na ovakav stepen neuravnoteženosti.



Slika 6: Histogrami bitnih atributa

S obzirom da je na histogramima primjetno prisustvo implicitnih NaN vrijednosti u obliku nula, a izrazito primjetno kod atributa *Insulin*, potrebno je obaviti pretprocesiranje.
Najprije će se nule u sumnjivim kolonama zamijeniti NaN vrijednostima. Nakon toga, potrebno je popuniti te vrijednosti. Popunjavanje je obavljeno pomoću medijane (median). Ovaj korak je obavezan jer TabPFN ne podržava NaN vrijednosti.
Histogrami atributa po klasama nakon ovih koraka su prikazani na slici ispod.



Slika 7: Histogrami bitnih atributa nakon pretprocesiranja

-Distirbucija glukoze prikazuje prilično zvonast oblik, pri čemu je najveća gustina uzoraka između 100-120. Pokrivenost podacima je dobra na cijelom opsegu, sa većom gustinom uzoraka u centralnom dijelu. Primetna je razlika između distribucija za klase 0 i 1. Pacijenti bez dijabetesa evidentno i potpuno očekivano imaju niže vrijednosti glukoze. Ovo glukozu čini vjerovatno najbitnijim atributom pilikom klasifikacije.
-Distribucija krvnog pritiska nema više nagomilavanja u nuli što prikazuje uspešnost pretprocesiranja. Pokrivenost vrijednostima je dobra na opsegu od 40 do 100, ali ne i na cijelom opsegu. Distribucije po klasama se dosta preklapaju, što ovaj atribut čini lošim prediktorom.
-Distribucija *BMI* pokazuje najvišu gustinu između 25-35 i pokrivenost je solidna u opsegu 20-50, dok se ka ekstremnim vrijednostima gustina značajno smanjuje. Distribucije se ne preklapaju, primjetno je da osobe sa većim *BMI* imaju veću šansu za dijabetes.
-Distribucija insulina ima drastično smanjen pik na 0. Pokrivenost podacima je veća u nižem opsegu, dok se gustina podataka značajno smanjuje iznad 200 sa dosta prazina. Distribucije se preklapaju i podaci su dosta raspršeni što ovaj atribut čini ne tako dobrim prediktorom.
-Distribucija *Age* prikazuje veću koncentraciju pacijenata između 20 i 70 godina, naručito u osegu od 20 do 30 godina. Pokrivenost je solidna.

1.4.3 Wine Quality Dataset

Wine Quality dataset je javno dostupan skup podataka i koristi se za višeklasnu klasifikaciju. Upotrebljava se za predikciju kaliteta vina na osnovu njegovih hemijskih karakteristika. Ovaj dataset se sastoji od dvije podvrste vina- crno i bijelo vino. Ovaj dataset dobijen je spajanjem Red Wine Quality i White Wine Quality datasetov-a. Prilikom toga dodata je i kolona type koja može imati vrijednsti red i white da bi bilo moguće razlikovati ih.
Cilj je klasifikacija vina po kvalitetu sa ocjenama od 3 do 9. Ocjene su dobijene od strane profesionalnih somelijera.
Set podataka je preuzet sa Kaggle i dosputan je na sljedećem linku:
https://www.kaggle.com/datasets/rajyellow46/wine-quality
Originaln izvor ovog skupa podataka je UCI Machine Learning Repository.
Format fajla je CSV, a veličina fajla je (390.38 kB).
Broj instanci ovog dataset-a je 6497 uzoraka od čega 4898 bijelih i 1599 za crnih vina. Ovaj dataset je nešto veći od prethodna dva, kao takav je pogodan za ispitivanje efikasnoti TabPFN modela na dataset-ovima srednje veličine. Takođe, za razliku od ranijih datasetov-a u ovom poglavlju, ovdje se radi o višeklasnom klasifikacionom zadatku, što je interesantno za evaluaciju sposobnosti TabPFN-a u ovim uslovima.
Svaki uzorak je defenisan sa 13 atributa raspoređenih na sljedeći način:

Redni broj	**Ime kolone**	**Opis**
1	type	Kategorijski atribut koji se odnosi na vrstu vina- red ili white. Dodata nakon spajanja dva dataset-a
2-12	10 numeričkih atributa	Opis karakteristika vina
13	quality	Klasa kvaliteta (od 3 do 9)

Atributi ovog dataset-a su:

1. *Type*- vrsta vina
2. *Fixed acidity*- kiselost koja ne isparava lako
3. *Volatile acidity*- kiselost koja može ispariti
4. *Citric acid*- limunska kiselina
5. *Residual acid*- ostatak šećera nakon fermentacije
6. *Chlorides*- sadržaj soli
7. *Free sulfur dioxide*- slobodni SO_2
8. *Total silfur dioxide*- ukupna količina SO_2
9. *Density*- gustoća tečnosti

10. *pH*- vrijednost koja je pokazatelj kiselosti
11. *Sulphates*- sulfati
12. *Alcohol*- sadržaj alkohola (u %)
13. *Quality*- **ciljna varijabla** koja označava kvalitet u obliku ocjene u rasponu od 3 do 9 [8]

Raspodjela klasa:

Svje ocjene nisu podjednako zastupljene. Ocjene 5, 6 i 7 su daleko najčešće. Dok su ocjene 3,4,8 i 9 dosta slabije zastupljene.

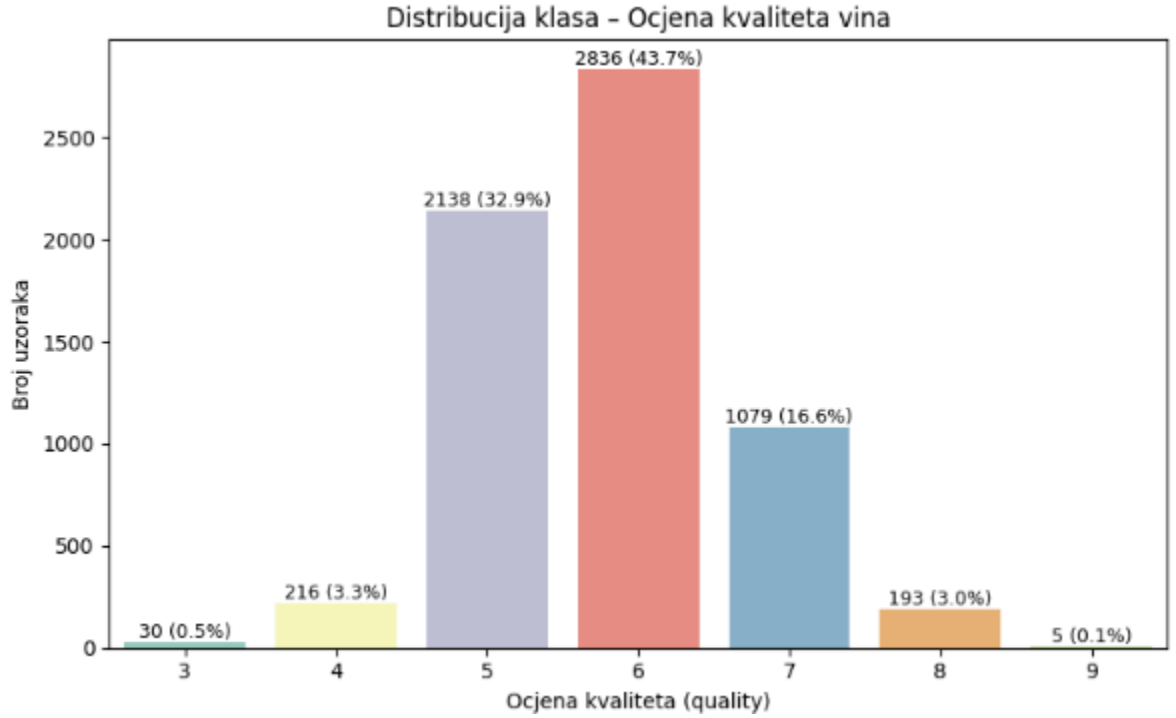
U našem fajlu Analiza_predloženih_datasetova.ipynb [9] dostupni su tabelrani prikaz i grafici.

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.270	0.36	20.7	0.045	45.0	170.0	1.00100	3.00	0.45	8.8	6
1	white	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6
2	white	8.1	0.280	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.44	10.1	6
3	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
4	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
...	...	...	...	...	...	...	...	...	...	...	...	...	...
6492	red	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.58	10.5	5
6493	red	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	NaN	11.2	6
6494	red	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6
6495	red	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.71	10.2	5
6496	red	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.66	11.0	6

6497 rows × 13 columns

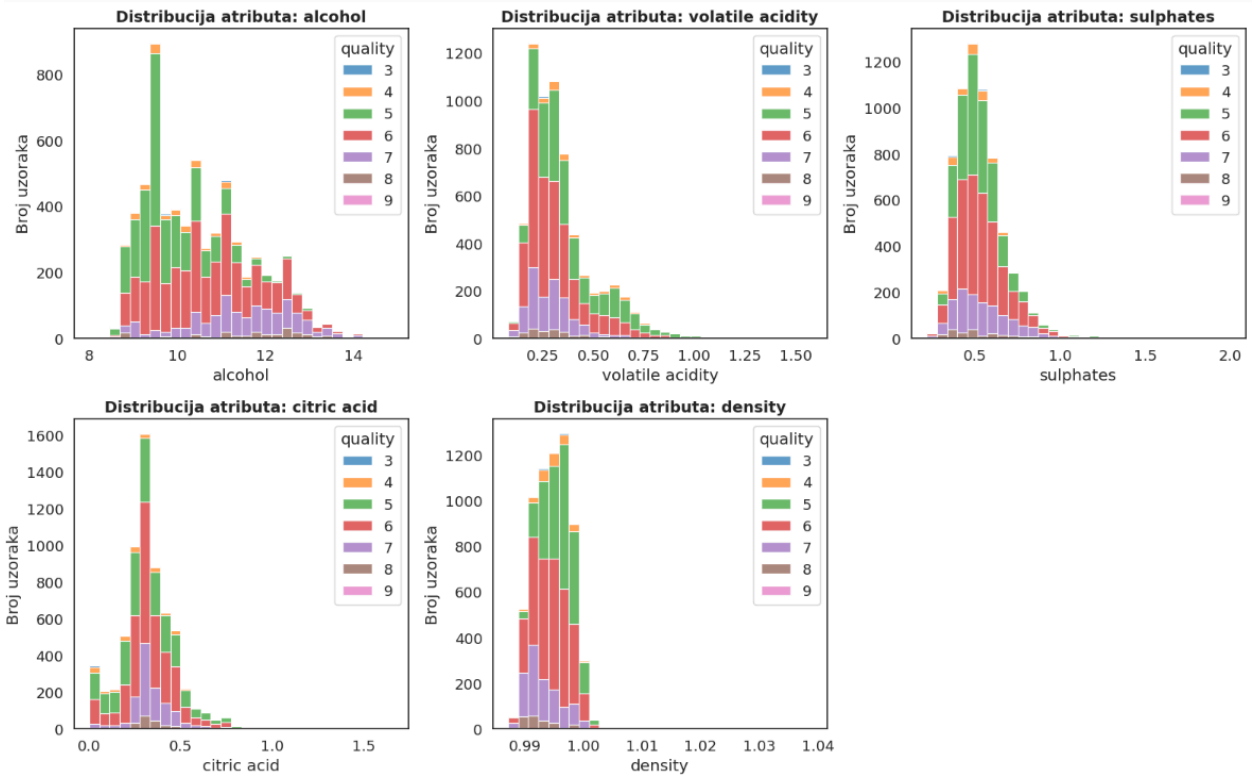
Slika 8: Tabelarni prikaz Wine Quality dataset-a

Već na osnovu slike iznad je primjetna velika zastupljenost srednjih ocjena, konkretno u ovom slučaju 5 i 6.



Slika 9: Dijagram distibucije ciljne varijable Vine Quality dataset-a

Grafikon na slici iznad jasno prikazuje izuzetnu disbalasniranost. Dominate su klase 5 i 6 čineći zajedno 76.6% cijelog datasets. Kada se tome doda i klasa 7, one zajedno čine 93% podataka. Ovo može rezultirati većom pristasnošću ka dominantnijim klasama i lošijim performansama pri prepoznavanju klasa 3,4,8 i 9. Međutim, TabPFN zbog svoje obučenosti na raznovrsnim datasetovbima, uključujući i nebalanrisane kao što je ovaj, može se pokazati robusnijim od klasičnih modela.



Slika 10: Histogram najrelevantnijih atributa u Wine Quality Dataset-u

Sa slike iznad može se zaključiti da je raspon vrijednosti atributa različit, pa tako za atribut alcohol imamo raspon od 8 do 14, dok za atribut density 0.99 do 1.04.

- Atribut *alcohol*: većina uzoraka se nalazi na opsegu između 9% i 11.5%. Distribucije različitih klasa se znatno preklapaju naročito na već spomenutom opsegu. Međutim, uočljivo je da se vina većeg kvaliteta sa ocjenama 7,8 i 9 javljaju pri većim vrijednostima alhokola, dok su vina sa ocjenama od 3 do 5 dominantnije pri nižim koncentracijama alkohola.
- Atribut *violate acidity*: Najveći broj uzoraka je između 0.2 i 0.4. Opet imamo preklapanje vrijednosti, s tim da vina nižeg kvaliteta češće imaju veće vrijednosti ovog atributa.
- Atribut *sulphates*: Većina uzoraka se kreće na rasponu od 0.4 do 0.8. Preklapanje vrijednosti je dosta veliko, ali je primjetno da vina višeg kvaliteta imaju manje vrijednosti ovog atributa.

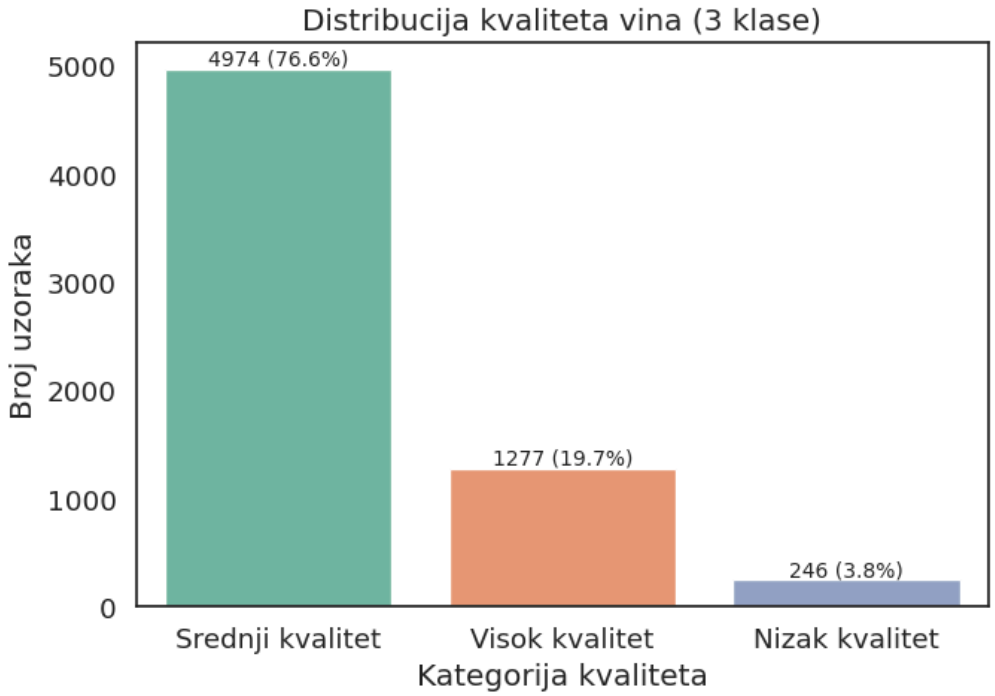
- ➢ Atribut *citric acid*: distribucija je najveća na rasponu između 0.2 i 0.5. Preklapanje je veoma izraženo. Postoji umjerena pozitivna povezanost sa kvalitetom.
- ➢ Atribut *density*: Svi uzorci su u uskrom rasponu od 0.99 do 1.002, dok su razlike između vrijednosti atributa za pojedine klase veoma suptilne. Ovaj atribut je teško iskoristiv za precizno razdvajanje klasa.

Takođe, možemo uočiti da je pokrivenost podacima dosta dobra i da uzorci sa ocjenom 3 i 9 nisu primjetni na histogram.

Originalna varijabla *quality* u Wine Quality dataset-u sadrži sedam diskretnih klasa. Međutim, ove klase su disbalansirane, pri čemi je većina uzorka u klasama 5 i 6. Kako bi se zadatak pojednostavio i napravio bolji balans između klasa, izvršeno je grupisaje u tri nove kategorije:

- Niži kvalitet: ocjene 3 i 4
- Srednji kvalitet: ocjene 5 i 6
- Visok kvalitet: ocjene 7, 8 i 9

Ova transfomarcija je implementirana dodavanjem klase *quality_grouped*, koja će sada predstavljati ciljnu varijablu.



Slika 11: Distribucija novog 3-klasnog klasifikaconog problema

Gornja slika prikazuje distribuciju nove ciljne varijable. Uočljiva su poboljšana u balansiranosti između klasa.

1.4.4 Student performance dataset

Student performance dataset korisitmo 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 dataset-a su uključeni razni faktori čiji uticaj na uspjeh učenika želimo ispitati. Navedno je koliko sati sedmično učenik/učenica provede učeći za određeni predmet, socio-ekonomski status porodice, pol učenika/učenice, završen stepen stručne spreme roditelja, broj izostanaka, zdravlje i drugi faktori koji su navedeni kao atributi dataset-a. U ovom segmentu ćemo samo analizirati verziju student-math.csv, pri čenu nam je cilja varijabla finalna ocjena iz matematike (atribut G3) na skali od 0 do 20 što omogućava klasifikaciju koja ima

```
Osnovni podaci o datasetu
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 395 entries, 0 to 394
Data columns (total 33 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   school      395 non-null    object
 1   sex         395 non-null    object
 2   age         395 non-null    int64
 3   address     395 non-null    object
 4   famsize     395 non-null    object
 5   Pstatus     395 non-null    object
 6   Medu        395 non-null    int64
 7   Fedu        395 non-null    int64
 8   Mjob        395 non-null    object
 9   Fjob        395 non-null    object
 10  reason      395 non-null    object
 11  guardian    395 non-null    object
 12  traveltime  395 non-null    int64
 13  studytime   395 non-null    int64
 14  failures    395 non-null    int64
 15  schoolsup   395 non-null    object
 16  famsup      395 non-null    object
 17  paid        395 non-null    object
 18  activities  395 non-null    object
 19  nursery     395 non-null    object
 20  higher      395 non-null    object
 21  internet    395 non-null    object
 22  romantic    395 non-null    object
 23  famrel      395 non-null    int64
 24  freetime    395 non-null    int64
 25  goout       395 non-null    int64
 26  Dalc        395 non-null    int64
 27  Walc        395 non-null    int64
 28  health      395 non-null    int64
 29  absences    395 non-null    int64
 30  G1          395 non-null    int64
 31  G2          395 non-null    int64
 32  G3          395 non-null    int64
dtypes: int64(16), object(17)
memory usage: 102.0+ KB
None
```

Slika 12: Prikaz osnovnih informacija

U naš *student_performance_dataset_analiza.ipynb* [10] fajl smo ispisali osnovne informacije o datasetu preko metode *info()*. Svi atribuiti su kompletni, što možemo vidjeti iz non-null count-a.

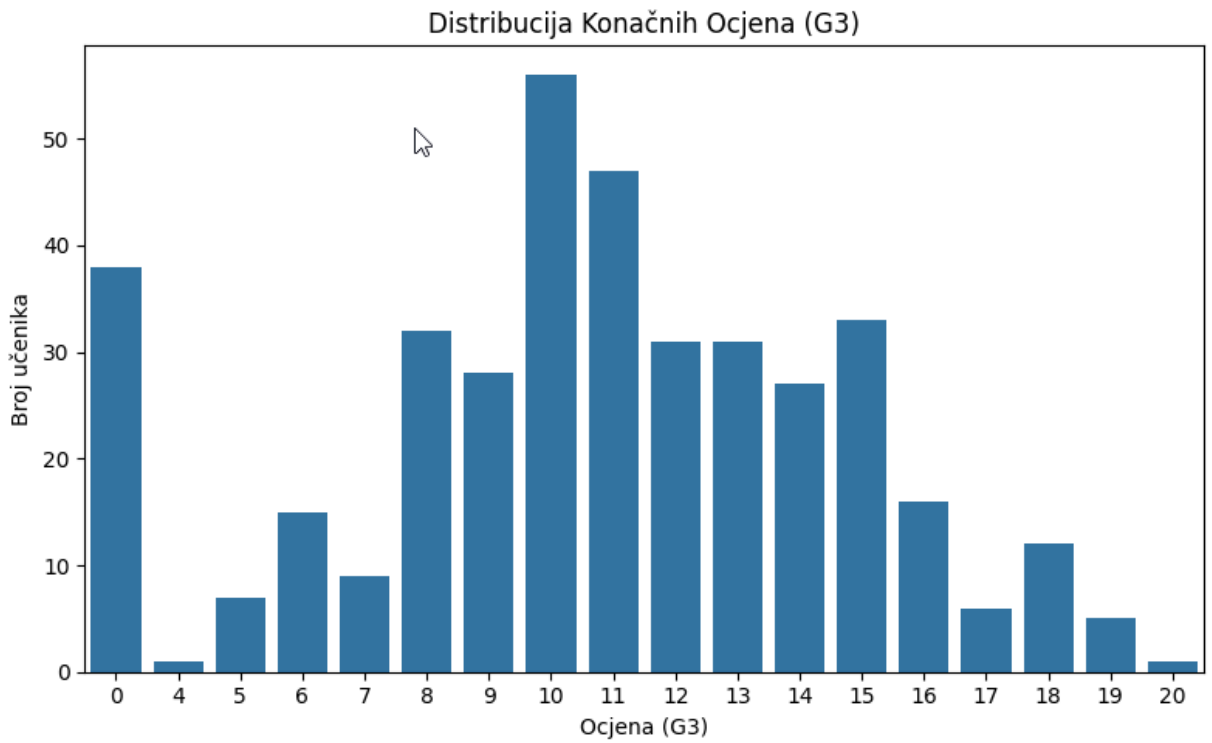
	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	famrel	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	3	4	1	1	3	6	5	6	6
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	5	3	3	1	1	3	4	5	5	6
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	4	3	2	2	3	3	10	7	8	10
3	GP	F	15	U	GT3	T	4	2	health	services	...	3	2	2	1	1	5	2	15	14	15
4	GP	F	16	U	GT3	T	3	3	other	other	...	4	3	2	1	2	5	4	6	10	10
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
390	MS	M	20	U	LE3	A	2	2	services	services	...	5	5	4	4	5	4	11	9	9	9
391	MS	M	17	U	LE3	T	3	1	services	services	...	2	4	5	3	4	2	3	14	16	16
392	MS	M	21	R	GT3	T	1	1	other	other	...	5	5	3	3	3	3	3	10	8	7
393	MS	M	18	R	LE3	T	3	2	services	other	...	4	4	1	3	4	5	0	11	12	10
394	MS	M	19	U	LE3	T	1	1	other	at_home	...	3	2	3	3	3	5	5	8	9	9

395 rows × 33 columns

Slika 13: Prikaz Student Performance dataframe-a

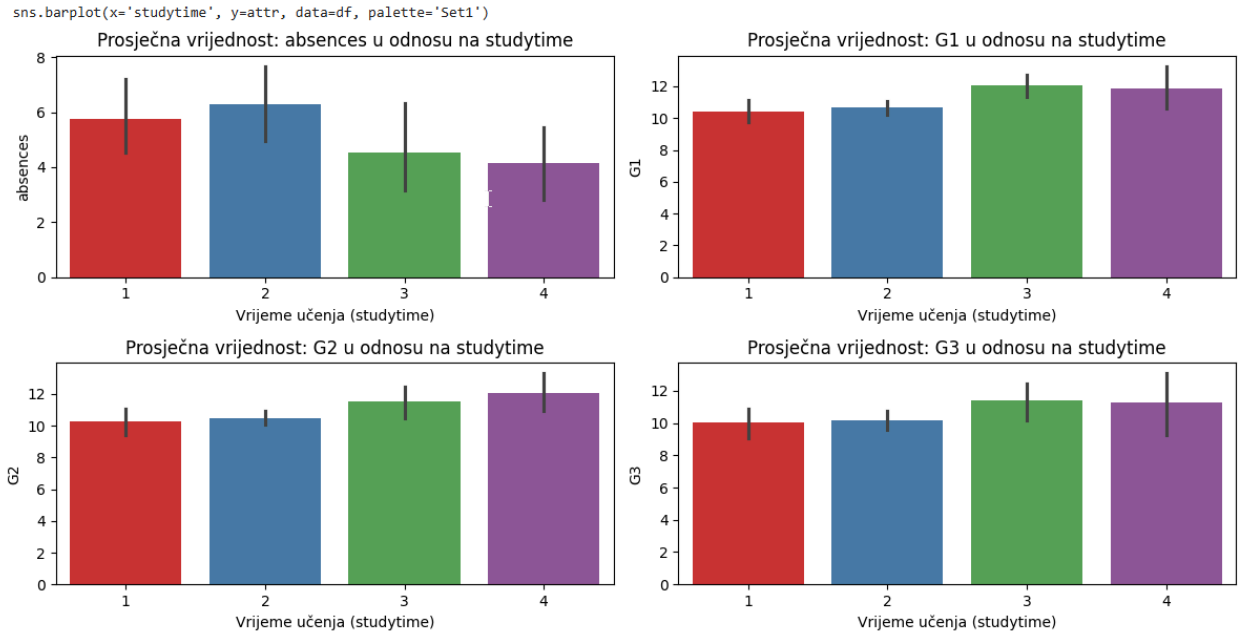
Navedeni csv fajl ima 395 redova i 33 atributa. U pitanju je dataset sa malim brojem redova što nam odgovara za TabPFN analizu. TabPFN očekuje attribute tipa float32. Nijedan naš atribut nema taj tip, pa bi za klasifikaciju potrebno enkodirati podatke

Csv fajl za navedeni dataset je preuzet sa Kaggle platforme i dostupan je na sljedećem linku: https://www.kaggle.com/datasets/rabieelkharoua/students-performance-dataset



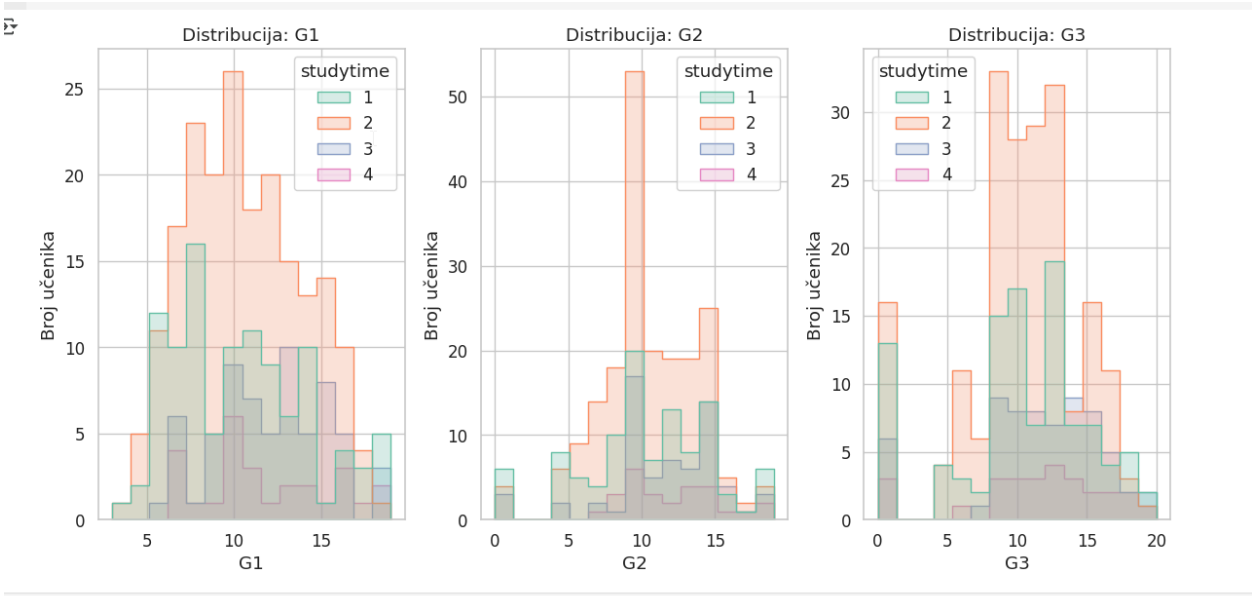
Slika 14: Distribucija konačnih ocjena iz matematike

Distribucija ima bimodalni oblik (imamo 2 izražena vrha). Većina učenika ima ocjene u srednjem rasponu.



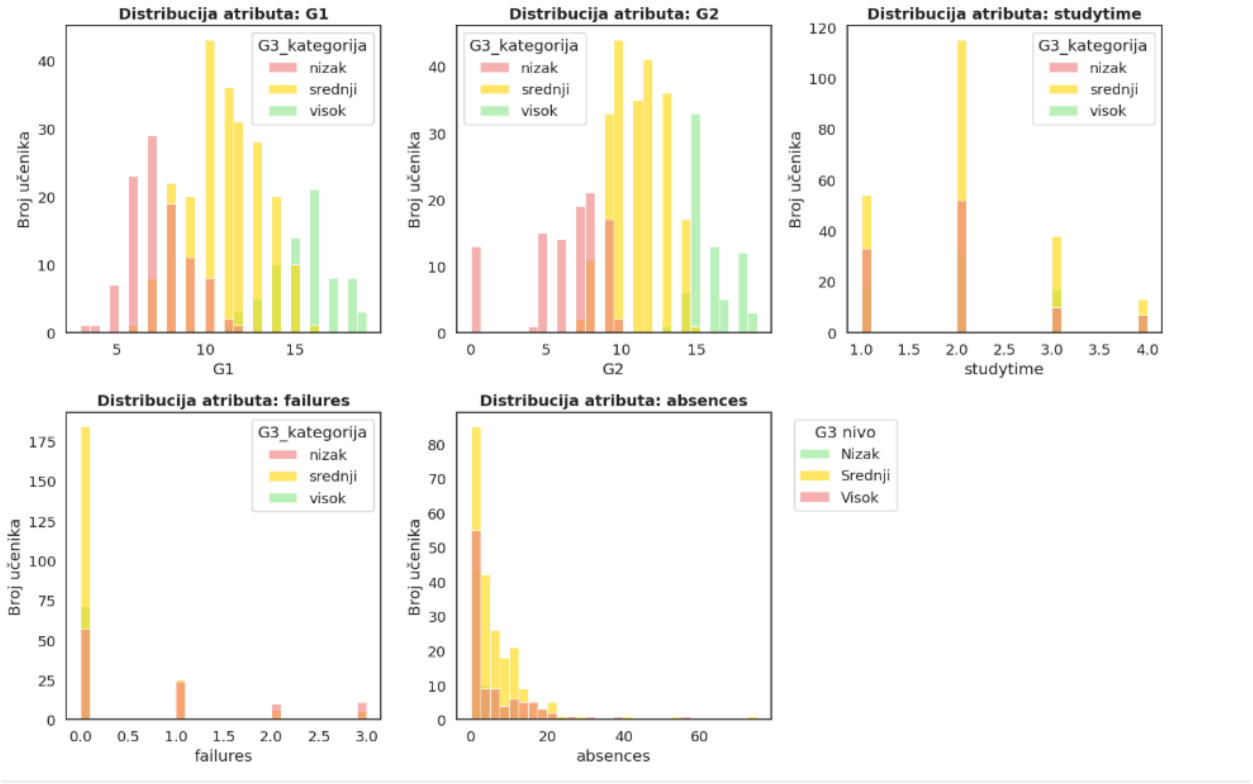
Slika 15: Prosječna vrijednost atributa absences, G1, G2 i G3 u zavisnosti od vremena učenja

Vidimo da ocjene rastu sa većim studytime, a izostanci opadaju. Dakle učenici koji više uče pored toga što imaju veće ocjene, imaju i manje izostanaka.



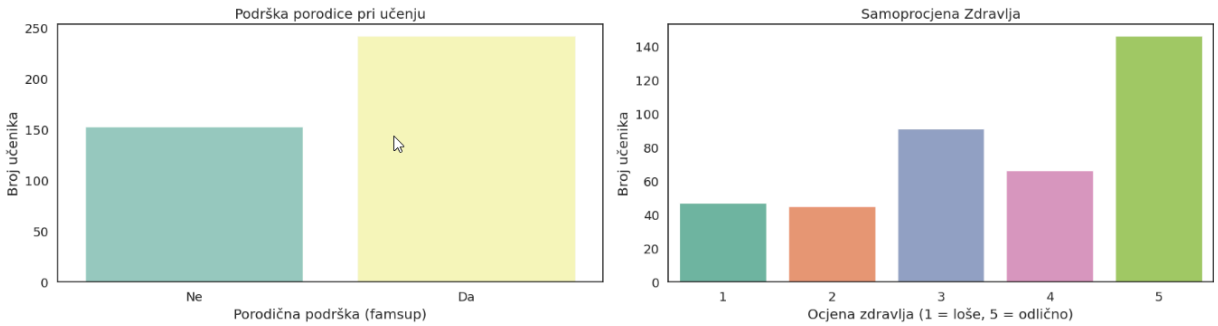
Slika 16: Distribucija ocjena u odnosu na vrijem učenja

Vidimo da većina učenika ima ocjene između 8 i 14 bez obzira na vijeme učenja. Možemo zaključiti da većina učenika uči umjereno, jer je kategorija 2 (umjereno učenje) najzastupljenjija u svim distribucijama.Možemo primijetiti za G3 da imamo učenike sa niskom finalnom ocjenom, uprkos visoke vrijednosti atributa sedmičnog učenja upućuje na uticaj dodatnih faktora na završnu ocjenu.



Slika 17: Histogram za finalnu ocjenu G3

Studenti sa visokim ocjenama G1 i G2, imaju visoku završnu ocjenu. Najjači prediktor niske finalne ocjene je broj prethodnih neuspjeha.



Slika 18: Distribucija porodične podrške, zdravlja

Većina učenika ima podršku porodice i procjenjuje zdravstveno stanje kao odlično.

Faza 2: Pregled stanja u oblasti

U posljednjih nekoliko godina, duboko učenje i posebno transformer arhitekture doživjele su veliki napredak u različitim oblastima, ali njihova primjena na tabelarne podatke dugo je bila ograničena u poređenju sa tradicionalnim metodama popsut random forest-a ili gradient booking-a. Ipak, noviji pristupi kao što je **TabPFN** (Tabular Prior-Data Fitted Network) unose značajne inovacije u ovu oblast, omogućavajući brzo i efikasno rješavanje problema klasifikacije bez potrebe za dodatnim treniranjem na ciljnim podacima. U nastavku su prikazana tri relevantna rada koja predstavljaju temeljna i savremena istraživanja u vezi s ovom tematikom.

2.1. Prvi rad

Naziv : ***"TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second"***

Autori : Noah Hollmann, Samuel Muller, Katharina Eggensperger, Frank Hutter
U ovom radu predstavljen je model TabPFN, baziran na transformer arhitekturi, treniran na više milion sintetičkih klasifikacionih zadataka sa ciljem da generalizuje na nove, do tada neviđene datasetove bez dodatnog treniranja. Ulaz u model je cijeli dataset (sve instance i ciljna instanca), a izlaz je predikcija klase za jednu ciljnu tačku.
Model koristi GPT-2-style transformer, gdje se svaka instanca kodira sa svojim karakteristikama, labelom (ako je poznata) i binarnom maskom koja označava da li je to ciljna instanca. Evaluacija je vršena na 18 datasetova iz OpenML-CC18 kolekcije i dodatnih 67 numeričkih skupova. Model je poređen sa klasičnim tehnikama kao što su XGBoost, Random Forest i AutoML sistemi.

Što se tiče strukture modela, sastoji se od input sloja kojeg čini vektor dimenzije 512 za svaku instancu, koji je nastao spajanjem karakteristika i meta-podataka. Ulaz se proširuje preko learnable linear transformacije. Koristi 12 encoder slojeva gdje svaki sloj ima multi-head self-attention mehanizam sa 8 glava pažnje. Nakon pažnje slijedi dvostruki linearni sloj sa ReLU aktivacijom. Slojevi koriste Layer Normalization i Residual Connections. Vektor odgovarajuće ciljne instance se izdvaja i šalje kroz linearnu projekciju koja predviđa klasu. Aktivacijska funkcija na izlazu je softmax, jer se radi o klasifikaciji.

Problem klasifikacije tabelarnih podataka do sada je rješavan prvenstveno korištenjem tradicionalnih algoritama kao što su Random Forest, Gradient Boosting, logistička regresija i SVM. Ovi algoritmi daju dobre rezultate, naročito kod manjih datasetova i problema sa jasno definisanim strukturiranim podacima.

Ključna prednost TabPFN-a je izuzetno brza inferencija - model može klasifikovati novu instancu u milisekundama, zahvaljujući prethodno stečenom znanju tokom obuke na sintetičkim zadacima. Rezultati pokazuju da model ima sposobnost da generalizuje i na nepoznate distribucije podataka, te da održava visoke performanse bez dodatnog treniranja. Na taj način, TabPFN nudi rješenje koje kombinuje visoku preciznost, otpornost na overfitting i vrlo malu latentnost, što ga čini pogodnim za stvarne aplikacije gdje je brzina odlučivanja od ključne važnosti.

Važnost ovog rada za naš projekat ogleda se u činjenici da prikazuje kako se in-context učenje može efikasno primijeniti na tabelarne klasifikacione zadatke, te kako se distribucije podataka mogu simulirati unaprijed i koristiti za generalizaciju. Ovo omogućava da se razvijaju modeli koji su trenirani "offline", ali vrlo brzo i precizno klasifikuju podatke "online". [11]

2.2. Drugi rad

Naziv: ***"Scaling TabPFN: Sketching and Feature Selection for Tabular Prior-Data Fitted Networks"***

Autori: Benjamin Feuer, Chinmay Hegde, Niv Cohen

Rad se bavi problemom ograničene skalabilnosti osnovnog TabPFN modela. S obzirom da transformeri zahtijevaju memoriju koja raste kvadratno sa brojem instanci, model postaje teško primjenljiv na većim datasetovima. Uvedene su dvije ključne komponente, Sketching i Feature Selection. Sketching je tehnika koja koristi slučajni sažetak podataka da bi smanjila broj instanci koje se obrađuju u modelu, uz zadržavanje statističke reprezentacije. Feature Selection je automatski algoritam koji selektuje najinformativnije karakteristike koristeći kriterije kao što su mutual information i varijabilnost.

Model se dalje prilagođava tako da koristi ograničen broj "ulaznih slotova" koji se dinamički popunjavaju najrelevantnijim podacima. Testiran je na različitim realnim i simuliranim skupovima podataka sa više od 10 000 instanci.

Strukturu modela čine Sketching, Feature selection i transformer encoder slojevi. Prije ulaska u transformer, koristi se random projection koji redudira broj instanci na reprezentativni podskup. Nakon toga je implementirana varijanca-based selekcija atributa koja eliminiše najmanje varijabilne osobine, smanjujući ulaznu dimenziju. Da bi se stabilizovalo treniranje i spriječila degradacija performansina većim dubinama, svaki sloj koristi normalizaciju ulaza i rezidualne veze koje preskaču sloj i omogućavaju direktni protok informacija. Model koristi 8 do 12 transformera enkodera u nizu.

Uvođenjem Sketching-a i Feature Selection-a, ovaj rad kombinuje statističke tehnike obrade podataka sa principima efikasne arhitekture dubokih modela. Takođe, po prvi put se u kontekstu TabPFN-a jasno modelira kako se resursno intenzivni modeli mogu optimizovati za veće količine podataka bez gubitka performansi.

Rezultati ukazuju da model ne samo da održava konkurentne performanse u odnosu na originalni TabPFN, već ih u nekim slučajevima i nadmašuje na velikim datasetovima, zahvaljujući boljem upravljanju ulaznim informacijama. Time se validira ideja da pametna selekcija instanci i atributa može nadomjestiti potrebu za obradom cijelog skupa podataka. Eksperimenti su pokazali da primjenom sketching i selekcije atributa može da se zadrži visoka tačnost modela, uz drastično smanjenje memorijskih i procesorskih zahtjeva.

Važnost ovog rada za naš projekat leži u tome što pokazuje načine optimizacije memorijske efikasnosti TabPFN modela, čime se otvara prostor za njegovu primjenu i na datasetove koji prelaze prvobitna ograničenja. Ove metode su korisne i kao inspiracija za dizajn vlastitih pretprocesorskih koraka ili modula za selekciju podataka prije primjene AI modela. [12]

2.3. Treći rad

Naziv: ***"TabPFN Unleashed: A Scalable and Effective Solution to Tabular Classification Problems"***

Autori: Si-Yang Liu, Han-Jia Ye

TabPFN Unleashed je unaprijeđena verzija osnovnog TabPFN modela koja uvodi robustne i skalabilne komponente sa ciljem boljeg ponašanja na šumovitim i heterogenim podacima. Arhitektura koristi princip bagging-a, gdje se više varijacija modela izvršava paralelno na različitim podskupovima podataka. Kombinacija daje stabilniju i manje pristrasnu predikciju.

Uveden je dinamički enkoder koji se prilagođava distribuciji podataka tako što automatski modifikuje tokove podataka i težine karakteristika, što značajno poboljšava generalizaciju.

Model je treniran i testiran na preko 200 datasetova, uključujući oblasti poput medicine, finansija i informatike. Pokazano je da u većini slučajeva prevazilazi performanse modela kao što su CatBoost i AutoGluon, a posebno se istakao na datasetovima sa neuravnoteženim klasama i velikim brojem nerelevantnih atributa.

Strukturu modela čine Ensamble, Dynamic encoding, transformer backbone i Fusion slojevi. U Ensamble sloju dataset se dijeli u više podskupova, a za svaki se pokreće zaseban transformer model. Svaki model koristi do 512 instanci kao ulaz. Prije enkodovanja, podaci se transformišu adaptivno na osnovu statistika, čime se ulazi bolje prilagođavaju domenskoj distribuciji. Sadrži 6 do 12 transformer backbone slojeva gdje svaki sloj ima pažnju i FFN blokove. Rezultati iz svih bagging pod-modela kombinuju se pomoću weighted averaging mehanizma. Softmax aktivacija nad predikcijama nakon Fusion sloja daje konačnu klasu ciljne instance. Zbog ovakve strukture, model može obraditi razne vrste i veličine datasetova, uključujući one sa šumom, nedostajućim vrijednostima i velikim brojem nerelevantnih atributa.

Zahvaljujući bagging strukturi, rezultati modela su stabilniji, sa manjom varijansom i većom robusnošću. Kombinacija predikcija iz više podmodela omogućava ublažavanje uticaja slučajnih varijacija u podacima, dok dinamičko enkodovanje omogućava bolju adaptaciju na različite distribucije - što se direktno odrazilo na preciznost klasifikacije u realnim uslovima.

Važnost rada za naš projekat ogleda se u njegovoj fokusiranosti na robustnost, interpretabilnost i realne podatke, što je presudno za primjenu u okruženjima gdje klasifikacija mora biti pouzdana čak i u prisustvu nepotpunih ili šumovitih podataka. [13]

Faza 3: Izbor, analiza i pretprocesiranje dataset-a

3.1 Izbor dataset-a

Za realizaciju ovog projekata **Adult Income dataset**, poznat i pod nazivom **Census Income dataset**. Riječ je o jednom 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.
Razlozi za izbor *Adult Income dataset*-a:

1. Realni kontekst i izazovnost: Podaci su prikupljeni iz američkog cenzusa i predstavlja stvarne uslove pojedinaca. Ovaj problem klasifikacije ima direktnu primjenljivost u socijalnim analizama, marketingu i finansijama.
2. Mješoviti tipovi podataka: Dataset sadrži kombinaciju numeričkoj i kategorijskih atributa, što ga čini idealnim za testiranje modela i njegovih sposobnosti da se nosi sa komleksnošću stvarnih tabelarnih podataka. TabPFN je u osnovi dizajniran da funkcioniše u ovakvim uslovima.
3. Raspoloživost i obim podataka: S obzirom da sadrži skoro 49 000 instanci, omogućava eksperimentisanje sa različitim veličinama ulaznih podataka. U okviru projekta planirano je evaluiranje modela na uzorcima od:

- 10 000 instanci (maksimalan kapacitet TabPFN-a)
- 5 000 instanci
- 2 500 instanci
- 500 instanci (granični slučaj)

Ova analiza će pružiti uvid u robusnost i skalabilnost našeg modela.

4. Distribucija klasa i balansiranje: Ciljna varijabla *income* ima blago neuravnoteženu distribuciju. Ovo će nam omogućiti ispitivanje kako model radi sa ovakvi neuravnoteženim klasama.
5. Jasna interepretacija rezultata: Klasifijacija prihoda je jasna za tumačenje i vizualizaciju. Ovo će nam omogućiti jasan prikaz performanski modela.

3.2 Analiza dataset-a

U sklopu ovog projekta izabrana je verzijaAdult Income dataset-a koji je dostupan putem Kaggle platforme, pod nazivom *Adult Census Income*.
Kreirali su ga Ronny Kohavi i Barry Backer na osnovu podataka iz popisa stanovništva SAD-a iz 1994.godine.
Ovaj dataset je prvobitno objavljen na UCI Machine Learning Repository. Originalni dataset ima 48 842 instance. Često se dijeli na:

- adult.data: 32 561 instanca (koristi se kao trening set)
- adult.test: 16 281 instanca (koristi se kao test set)

Dataset je kasnije prenesen na Kaggle i dodatno formatiran radi lakšeg korištenja, ovo će biti verzija koju ćemo koristit.Link do verzije korištene na ovom projektu:
https://www.kaggle.com/datasets/uciml/adult-census-income
Preuzet je u CSV formatu. Također, može se učitati uz pomoć Pandas biblioteke nakon preuzimanaj datoteke. Veličina fajla je 4.1 MB.
Dataset ukupno ima 32 561 reda tj.instance. Svaka instanca predstavlja demografske i ekonomske karakteritiske jedne osobe.
Također, u dataset-u se nalazi 14 ulaznih atributa i jedna ciljna varijabla *income*.
Ovde je riječ o binarnoj klasifikaciji, tačnije cilj je predvidjeti da li osoba zarađuje:

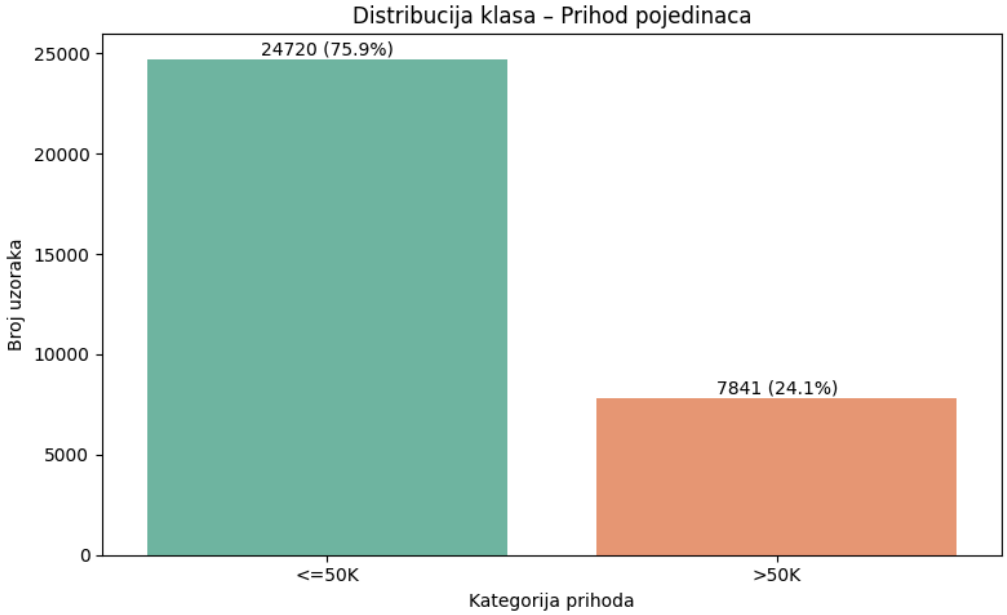
- <=50K (manje ili jednako 50 000 USD godišnje)
- >50K (više od 50 000 USD godišnje)

Pregled atributa po tipu uz kratak opis:

Atribut	**Tip podatka**	**Opis**
age	numerički	Starost osobe
workclass	kategorijski	Tip poslodavca
fnlwgt	numerički	Statistički težinski faktor

eduction	kategorijski	Nivo obrazobanja
education.num	numerički	Brojčana reprezentacija nivoa obrazovanja
marital.status	kategorijski	Bračni status
occupation	kategorijski	Vrsta zanimanja
relationship	kategorijski	Povezanost sa domaćinstom
race	kategorijski	Rasa
sex	kategorijski	Pol
capital.gain	numerički	Kapitalna dobit u prethodnoj godini
capital.loss	numerički	Kapitalni gubitak u prethodnoj godini
hours.per.week	numerički	Broj radnih sati nedeljno
native.country	kategorijski	Zemlja porijekla
income	kategorijski(ciljna varijabla)	Klasa prihoda

Na osnovu tabele primjetan je značajan broj kategorijskih atributa. U našem fajlu Analiza_Adult_Census_dataseta.ipynb [14] dostupni su grafici o kojima će u nastavku biti riječi.



Slika 19: Distribucija klasa Adult Census dataset-a

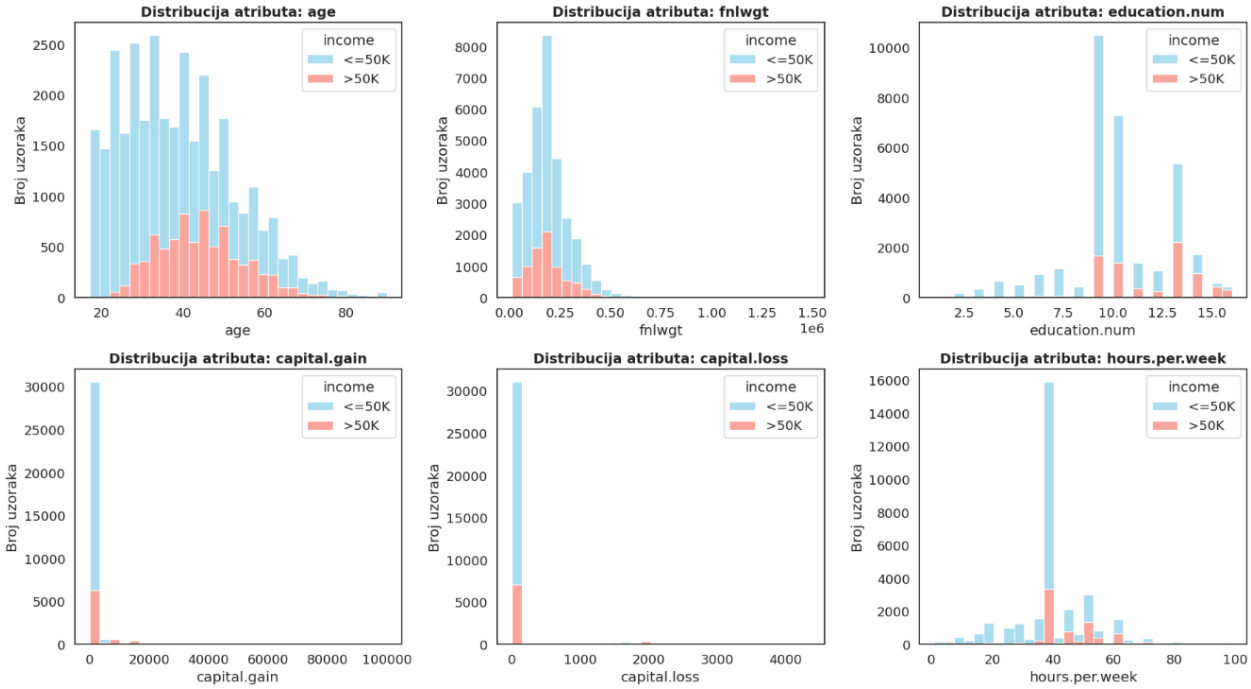
Na osnovu slike iznad možemo analizirati distribuciju klasa.
Klasa <=50:

- 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

Iz ovih podataka je jasno da je klasa <=50K znatno dominantnija što ovaj dataset čini disbalansiranim. Kod tradicionalnih modela disbalans može dovesti do favorizovanja većinske klase. Međutim, TabPFN je otporniji prema disbalansu jer koristi Bayesove pistupe i već je treniran na problemima sa sličnim strukturama. Ovo predstavlja rizik ali i idealnu priliku za ispitivanje kako se TabPFN u praksi ponaša u ovakim uslovima.



Slika 20: Distribucija numeričkih atributa

Na osnovu slike koja prikazuje distribuciju numeričkih atributa možemo zaključiti:

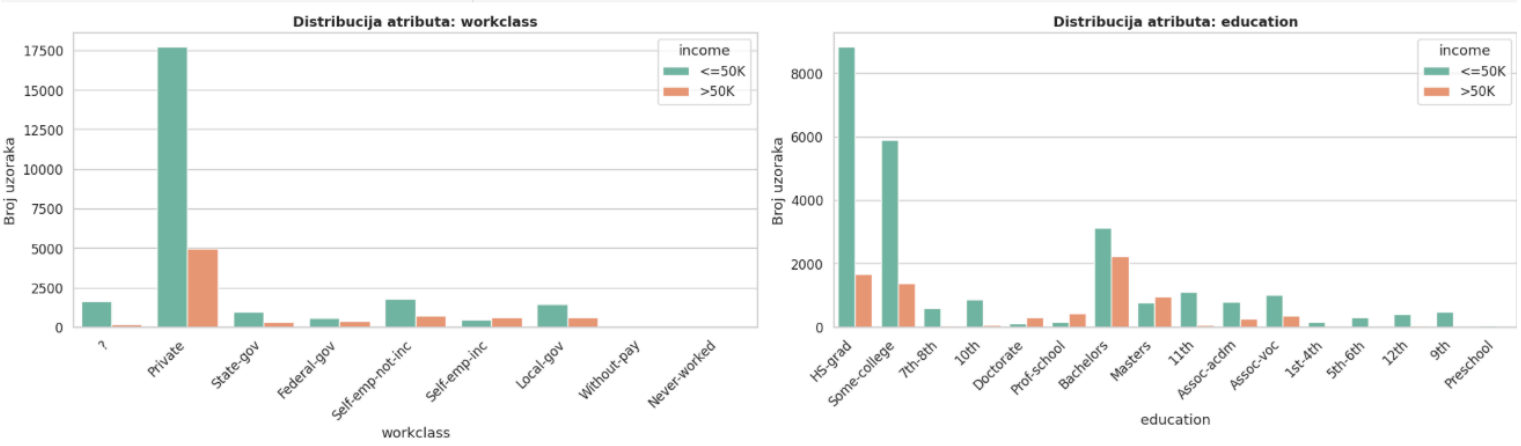
- ➢ *age:* Atribut age prikazuje relativno normalnu ali i desno-iskrivljenu distribuciju. Najveći broj uzoraka je na rasponu između 20 do 50 godina sa postupnim smanjenjem ka starijim godinama. Distribucije za <=50K i >50K se značajno preklapaju. Međutim, svakako je

primjetno da klasa >50K ima veći udio uzoraka između 30-50 godina što govori da ljudi u srednjoj životnoj dobi više zarađuju.

- ➢ *fnlwgt:* Atribut je izrazito desno iskrivljen. Većina vrijednosti je koncentrisana oko 0.25 sa dugim repom koji se proteže do većih vrijedosti. Pokriva širok raspon vrijednosti sa jako malom gutinom pri većim vrijedostima. Distribucije se gotovo potpuno preklapaju što ovom atributu daje vrlo malu prediktivnu moć.
- ➢ *educaton.num:* Diskretna distribucija s jasnim vrhovima. Najčešće vrijednosti su 9-10(HS-grad) i 13(Bachelors). Pokriva raspon od 1 do 16. Postoji razlika u distibuciji klasa. Klasa <=50K ima veći udio kod nižeg obrazovanja, dok klasa >50K ima značajno veči udio pri većem obrazovanju.
- ➢ *capital.gain:* Ova distribucija ima izrazitu koncentaciju na nuli. Ovo je normalna pojava jer ovaj atribut predstavlja neto dobitak iz investicija (akcije, nekretnine itd.). Međutim, postoji mali broj instanci koje imaju značajno veće vrijedosti (npr. 5 000, 10 000), koje su potpuno odvojene od glavne mase podataka na nuli. Ove visoke vrijednosti su ekstremne i predstavjaju outlier. Iako su vrijednosti različite od 0 rijetke, mogu biti snažan prediktor kada se pojave. Evidentno je da klasa >50K ima veći udio u vrijednostima većim od nule.
- ➢ *capital.loss:* Slična je situacija kao sa capital.gain.
- ➢ *hous.per.week:* Multimodalna distribucija sa istaknitim vrhom na 40 radnih sati nedeljno. Distribucije klasa se značajno preklapaju, ali je uočljivo da uzorci klase >50K više zastupljeni u višim radnim satima.

Primjetna je i velika razlika u rasponu vrijednosti između atributa, a najbolje je vidljivo na primjeru atributa *fnlwgt* i *capital.gain*.

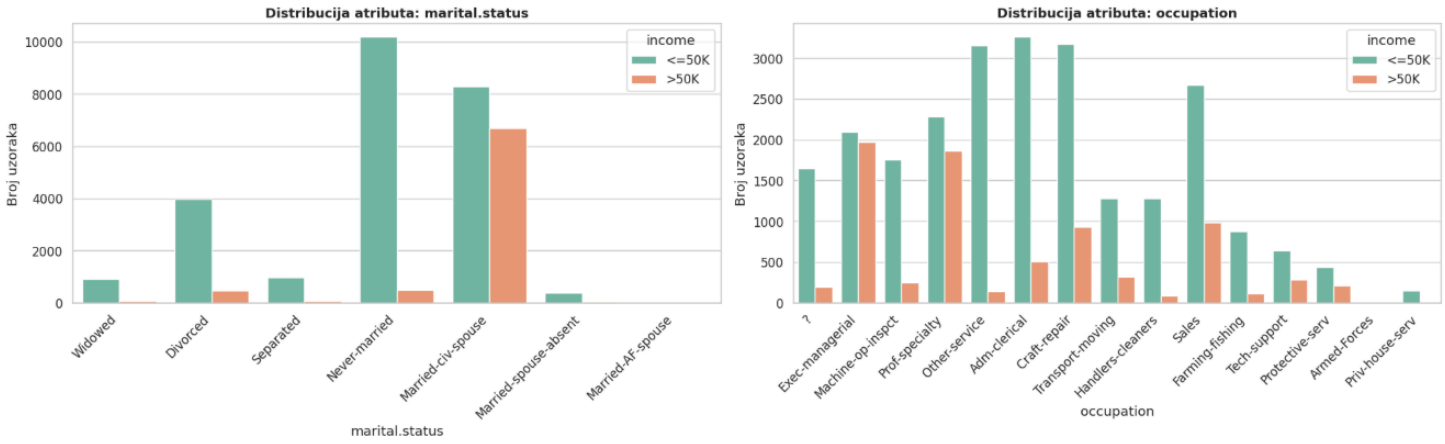
Distribucija kategorijskih atributa je prikazana na nešto drugačiji način, a koji je prikazan na narednim slikama.



Slika 21: Distribucija workclass i education atributa

Na osnovu slike moguće je zaključiti:

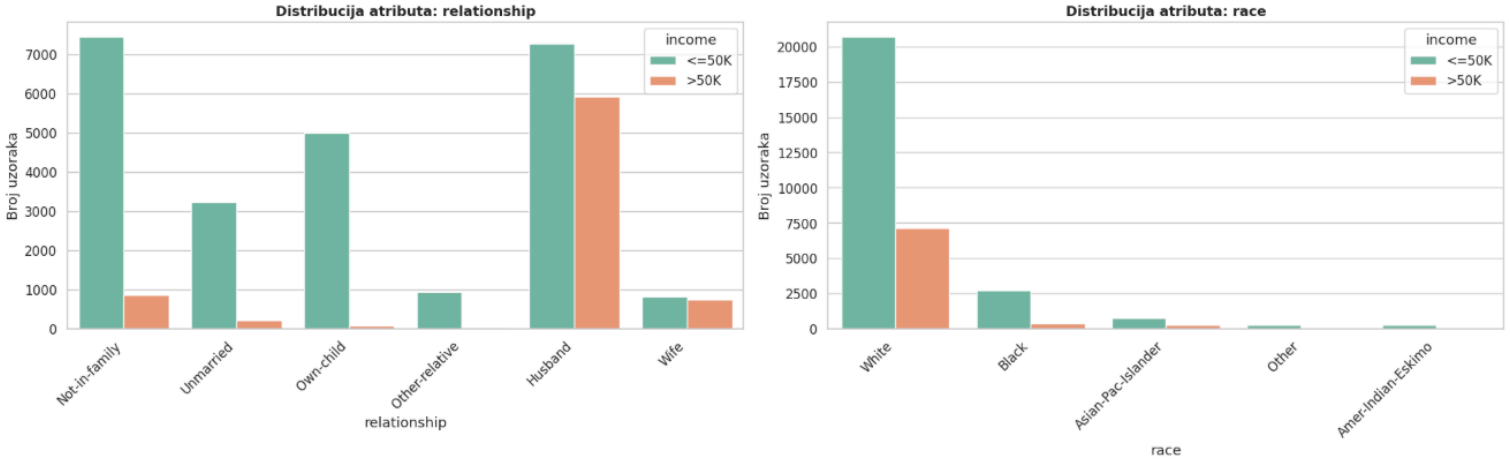
- workclass: Dominatna je kategorija private i veoma je zastupljena je u klasi <=50K. Većina kategorija ima uzorke u obje klase. Prisustvo ? kao kategorije će zahtjevazti pretprocesiranje.
- education: Najčešći nivo obrazovanja je HS-grad, Some-college kao i Bachelors. Obrazovanje je snažan prediktor prihoda jer su uzorci iz klase >50K zastupljeniji pri višem obrazovanju. Ovo je primećeno već kod numeričkog atribita education.num.



Slika 22: Distribucija atributa marital.status i occupation

Na osnovu slike iznad moguće je zaključiti:

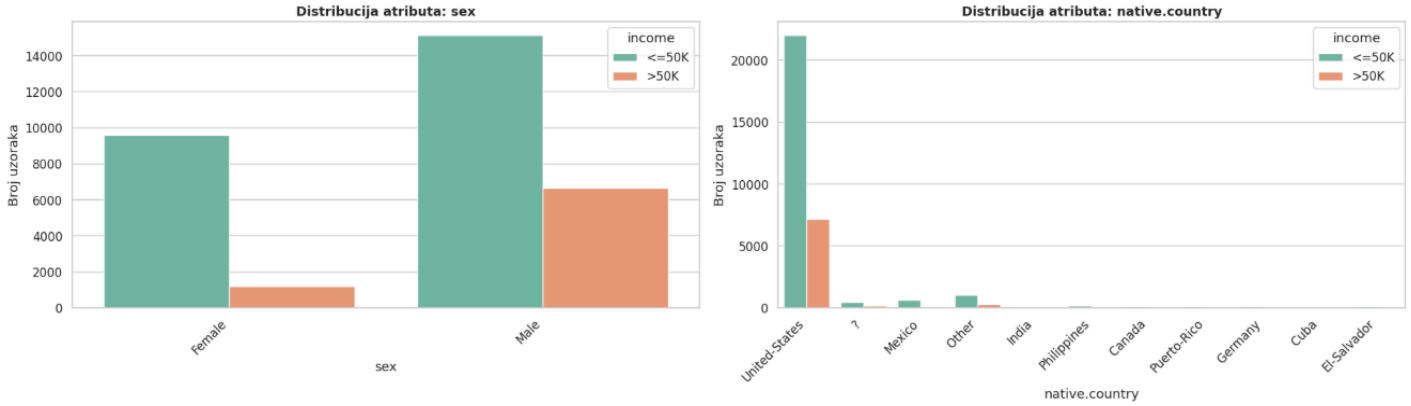
- Matrial.status: Atribut prkazuje jak odnos s prihodom. Kategorija Married-civ-spouce ima visoku zastupljenost u klasi >50K. Kategorija Married-AF-spouce je dosta rijetka što otežava zaključivanje.
- Occupation: Ovaj atribut ima mnogo kategorija. Kategorije poput Exec-managerial i Prof-speciality imaju veću zastupljenost u klasi >50K. Uzorci su zastupljeni u obje klase za sve kategorije. Prisustvo kategorije ? će zahtjevati pretprocesiranje.



Slika 23: Distribucija atributa relationship i race

O atributima sa slike se može reći:

- ➢ Relationship: Kategorije Not-in-family i Husband su dominantne u datasetu-u. Ovaj atribut ima snaznu prediktivnu moć s obzirom da kategorija Husband i Not-in-family (u manjoj mjeri) imaju dosta uzoraka u klasi >50K.
- ➢ Race: White kategorija je dominantna po pitanju brojnosti uzoraka, ali i po pitanju broja uzoraka u klasi >50K. Amer-Indian-Eskimo ima mali broj uzoraka i to zastupljenih u klasi <=50K.



Slika 24: Distribucija atributa sex i native.country

Na osnovu slike iznad možemo zaključiti:

- ➢ Sex: Dataset ima znatno više muških uzoraka.Primjetna je razlika u distribuciji. Female kategorija je pretežno u klasi <=50K.
- ➢ Native.country: Dominatna kategorija je Unitet-States s daleko najvećim brojem uzoraka. Iako postoje razlike, izrazita dominacija jedne kategorije i veliki broj rijetkih kategorija predstavljaju izazov. Prisustvo klase ? je evidentna i zahtjeva pretprocesiranje.

3.3 Pretprocesiranje

Iako je Adult Income dataset relativno čist i struktuiran, pretprocesiranje je neophodan korak kako bi se mogla ispitati efikasnost TabPFN modela. On, kao i većina savremenih modela, zahtjeva da podaci budu numerički, bez nedostajućih vrijednosti i skalirani. Metode pretprocesiranja smo podijelili u osnovne i napredne. Osnovne će biti objašnjene i urađene u ovom poglavlju, dok će napredne biti samo objašnjenje i primjenjene u fazi 4 kao dodatni korak ispitivanja efikasnosti.

3.3.1 Osnovne metode pretprocesiranja

1. **Rukovanje nedostajućim vrijednostima:**
   U dataset-u su prisutne implicitne nedostajuće vrijednsoti, koje su označene sa *?*. Pojavljuje se u kategoričkim varijablama woekclass, occuoation i native-country. Umjesto zamijene sa NaN, bolja je opcija zamijene sa stringom '*Unknown'*. Razlog je što

TabPFN ne podržava NaN vrijednosti, '*Unknown*' omogućava da se te informacije sačuvaju kao zasebna kategorija. Time model može učiti iz njih, bez da ih ignoriše ili izgubi.

```python
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())
```

Slika 25: Zamjena nedostajućih vrijednosti vrjednošću 'Unknown'

```
Broj '?' po kolonama:
age                 0
workclass        1836
fnlwgt              0
education           0
education.num       0
marital.status      0
occupation       1843
relationship        0
race                0
sex                 0
capital.gain        0
capital.loss        0
hours.per.week      0
native.country    583
income              0
dtype: int64

Preostale '?' vrijednosti:
age               0
workclass         0
fnlwgt            0
education         0
education.num     0
marital.status    0
occupation        0
relationship      0
race              0
sex               0
capital.gain      0
capital.loss      0
hours.per.week    0
native.country    0
income            0
dtype: int64
```

Slika 26: Rezultat izvršenja koda namijenjenog za rukovanje nedostajućim vrijednostima

2. **Enkodiranje kategorijskih atributa:**
   S obrizom da TabPFN traži da sve ulazne vrijednosti budu numeričke potrebno je enkodirati kategoričke atribute. Metode enkodiranja i njihova podobnost za naš projekat:
   - Label encoding:S obzirom na veći broj kategoričkih atributa i veliki broj mogućih kategorija pojedinih atributa **izabrali smo Label Encoding.** Iako uvodi vještački numerički redoslijed, TabPFN je manje osetljiv na ovaj problem od tradicioalnih modela. Razlog je jer koristi transformer arhitekturu i attention mehanizam.
   - One-Hot Encoding:Alternativa poput One-Hot Encoding bi značajno povećala broj atributa, posebno za kolone native-country i occupation, čime bi se povećala dimenzioalnost, usporilo treniranje i otežala generalizacija modela.
   - Ordinal Encoding: Poseban slučaj je kolona *education*. Ona posjeduje inherentan redoslijed i potrebno je upotrebiti Ordinal Encoding. Međutim, postojanje varijable *education.num* otklanja potrebu za ovim korakom.

```
from sklearn.preprocessing import LabelEncoder

# 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

Slika 27: Enkodiranje kategorijskih atributa i rezultat izvršenja koda

3. **Uklananje atributa**
   Atribut *fnlwgt* (final weight) predstavlja numerički faktor koji govori o tome koliko je svaki red reprezentativan za američku populaciju u cenzusu. Ona ima statički značaj i kao takva nike dobar prediktor ciljne varijable.

Atribut *eduction* se ukljanja iz razloga što bi ga trebalo enkodirati sa Ordinal Endocing i u tom slučaju bi imali dva praktično ista atributa. Encoding.num je ekvivalentan ovom atributu.

```
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')
```

Slika 28: Uklanjanje varijabli i rezultat izvršenja koda

4. **Enkodiranje ciljne varijable *income***
   Ciljna varijabla je tekstualnog formata (<=50K, >50K). Za potrebe binarna klasifikacije potrebno ju je enkodirati. Primjenićemo Label Encoding koji će ove podatke mapirati u 0 i 1.

```
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
```

Slika 29: Enkodiranje ciljne varijable income i rezultat

3.3.2 Naprednija pretprocesiranja

1. **Grupisanje rjetkih kategorija**
   Kod atributa native-country koja ima veliki broj jedinstvenih kategorija, prisutan je veliki broj kategorija koje su prisutne u veoma malom broju redova. Ove vrijednosti mogu unijeti šum i izavati pretreniranje modela. Ovo je razlog zbog koga bi se one treble grupisati u jednu kategoriju pod nazivom ‘Other’.
2. **Standardizacija numeričkih klasa**
   Iako je TabPFN manje osetljiv na razlike u rasponu vrijednosti atributa, ovaj korak može doprinijeti većoj stabilnosti. Može se izvršiti skaliranje numeričkih atributa da imaju srednju vrijednost 0 i standardnu devijaciju 1 koristeći StandardScaler metode.

3. **Log-transformacija visoko asimetričnih atributa**
   Atributi kao što su capital-gain i capital-loss sadrže veliki broj nula. Ovakva distibucija otežava učenje. Da bi se ovaj problem ublažio, koristi se log-transformacija pomoću funkcije log1p. Ovime se ekstremne vrijednosti zbijaju i dobija se ravnomernija distribucija.
4. **Obrada outliera**
   Atributi hous-per-week i age sadržavaju pdatke koji značajno odstupaju od ostalih podataka. Ovo može imati negativan učinak na performanse. Može se koristiti tehnika winsorization gdje se outlier zamjenjuju s pragom 1. i 99. Percentila.
5. **Klasni disbalans**
   TabPFN dobro podnosti neuravnoteženost klasa. Moglo bi se primjeniti balansiranje ali je cilj da se na ovaj način ispita robusnost TabPFN-a.

3.4 Potencijalni rizici

Nakon detaljne analize i osnovnih pretprocesiranja mogu se identifikovati sljedeći potencijalni rizici:

1. **Disbalans klasa:**
   Iako će na ovaj dataset bili primjenjeno stratifikovao uzorkovanje od 10 000, 5 000, 2 500 i 500 uzoraka to neće riješiti osnovni problem disbalansa. Model može biti skloniji predviđanju dominantne klase iako je TabPFN dizajniran da bude robusniji na ovaj problem od tradicionalnih modela.
2. **Gubitak informacija kod zamjene '*?*' sa '*Unknown*'**
   Iako je došlo do zamjene ovih vrijednosti i kasnijeg enkodiranja, model će 'Unknown' tretirati kao bilo koju drugu kategoriju. Ovo može dovesti do propuštanja specifičnog razloga zašto podatak nedostaje.
3. **Pitanje semantičke tačnosti *Label Encoding*-a**
   Ova vrsta enkodiranja uvodi redoslijed među kategorijama koji nema stvarno značenje. Iako TabPFN ne koristi direktnu lineranu kombinaciju atributa, postoji realan rizik da model intrepretira redoslijed kao rang.
4. **Nelinearnost distribucije pojedinih atributa**
   Atributi capital-gain i capital-loss imaju specifičnu distribuciju sa velikim fokusom na nuli. Model može imati poteškoća pri pravilnom razlikovanju informacija, kao na primjer da napravi jasnu razliku između nula i pozitivnih vrijednosti.
5. **Utjecaj veličine uzorka**
   Model će biti ispitivan na uzorcima od 10 000, 5 000 i 500 uzoraka. S obzirom da se TabPFN koristi za datasetov-e do 10 000 instanci, očekuju se manje performanse kod većih uzoraka. Međutim, postoji i rizik da manji uzorci daju manje reprezentativan raspored klasa i varijaciju atributa, što može smanjiti tačnost i sposobnost generalizacije.

Faza 4: Odabir, formiranje, treniranje i testiranje modela

Cilj ovog projekta je binarna klasifikacija koristeći podskupove Adult Income dataset-a, gdje model na osnovu demografskih i radnih karakteristika predviđa da li osoba zarađuje manje ili više od 50 000 USD godišnje.

Za problem klasifikacije će se koristiti TabPFN (Prior-data Fitten Network) model.
Njegove ključne karakteristike su:

- Nije potreban klasičan trening: Za razliku od većine modela koji se moraju trenirati, TabPFN je obučen (pre-trained) na ogromnoj količini raznih podataka. Kada mu se da neki dataset, on će ga koristiti za učen je iz konteksta (in-context learning), prilagođavajući se tako jako brzo našem problemu.
- Radi veoma brzo: Model daje predikciju u samo jednom prolazu, bez epoha, iteracija i validacije.
- Zasnovan je na transfomer ahitekruti: to je moderna vrsta neurosnkih mreža koja koristi mehanizam pažnje kako bi se obratila pažnju na najvažnijie podatke za donošenje odluka.
- Jednostavno korištenje: Ne traži puno ručnih podešavanja i pretprocesiranja poput balanrisanja i skaliranja numeričkih podataka.

4.1 Izbor tehnologija

1. **Google Colab**
   Google Colab je okruženje koje omogućava izvršavanje Python koda u web pretraživaču ,bez potrebe za instalacijom softvera. Prednosti su: besplatan pristup računarskim resursima, jednostavno dijeljenje i kolaboracija, direktno je integrisan sa Google Drive-om. Za potrebe ovog projekta korišten je T4 GPU kako bi se ubrzala obrada i omogućio rad sa uzorcima do 10 000 instanci. CPU omogućava rad samo do 1000 instanci i sporiji je.
2. **Python**
   Python je odabran kao glavni jezik zbog jednotavnosti, čitljivosti i velikog broja dostupnih bibliteka za obradu podataka i mašinsko učenje. Verzija koja je korištena je Python 3.10 koja predstavlja standardnu verziju za Google Colab.
3. **Biblioteke**
   U projektu je korišteno više Python biblioteka:

Biblioteka	**Namjena**
pandas	Učitavanje i obrada tabelarnih podataka
numpy	Numeričke operacije
matplotlib i *seaborn*	Vizualizacija podataka

scikit-learn	Pretpocerisranje, train/test skup, metričke evaluacije
tabpfn	Korištenje unaprijed treniranog TabPFN modela
touch	Za izvođenje modela unutar tabpfn biblioteke

4.2 Priprema podataka

Za testiranje performansi modela TabPFN, pripremljeni su podaci u tri različite veličine podskupova: 10 000, 5 000 i 500 instanci. Cilj je ispitati efikasnost u zavisnosti od količine podataka- od malog do maksimalnog preporučenog uzorka za TabPFN.
Uzorak veličine 10 000 instanci biće korišten za detaljnu analizu pripreme podataka, treniranja i evaluacije modela. Kod i postupak identičan je i za uzorke od 500 i 5 000 instanci.
U dokumentu Ispitivanje_efikasnosti_TabPFN.ipynb[15] dostupani su kodovi koji će biti prikazani u okviru ovog poglavlja, ali i poglavlja 4.3 i 4.4.

```
# 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
)
```

Slika 30: Kod za pripremu podataka

Nakon učitavanja i pretprocesiranja podataka koje smo uradili u fazi 3, prelazi se na priremu podataka.

Najprije je izdvojen podskup od 10 000 uzoraka pomoću funkcije *train_test_split* iz biblioteke **scikit-learn** kako bi se izvršilo **stratifikovano uzorkovanje**. Ovime je omogućeno da raspodjela ciljnih klasa ostane proporcionalna u odnosu na cijeli skup, odnosno da balans ostane isti. Nakon toga smo dobijeni uzorak *df_10000* podijelili na ulazne attribute *X_10000* i ciljnu promjenljivu *y_10000*. Potom smo u sljedećem koraku podatke konvertovali u **NumPy nizove**, pri čemu su ulazni podaci konvertovani u **float32**, a cilje varijable u **int64** tip i ovime smo prilagodili podatke zahtjevima modela.
Kao krajnji korak pripreme podataka izvršili smo podjelu na trening i test skupove u omjeru 80:20. Trening skup sadrži 8 000, a test skup 2 000 redova. Ovo će nam omogućiti objektivnu evaluaciju modela.

4.3 Treniranje i evaluacija modela

TabPFN predstavlja inovativni model za klasifikaciju nad tabelarnim podacima. Ovaj model koristi **transformersku arhitekturu** što znači da se temelji na dubokoj neuronskoj mreži, pored toga koristi i pristup **Bayesva interferencija**.[16] Ona se odnosi na donošenje odlika na osnovu vjerovatnoće umjesto na osnovu fiksnih pravila.
Kada je upitanju treniranje kod TabPFN modela, važno je naglasiti da ne dolazi do klasičnog treniranja neuronske mreže, kao što je to slučaj kod drugih modela. Model je već **prethodno istreniran** na milijardama sintetičkih klasifikacionih zadataka i posjeduje već naučeno znanje koje se generalizuje na nove podatke. Poziv metode *fit()* u ovom slučaju ne znači klasično treniranje, odnosno da model uči iz početka, već pokušava da prepozna obrasce u novim podacima na osnovu sličnih zadataka na kojima je već unaprijed treniran, koristeći **Bayesove principe zaključivanja**.
Prije treniranja i evaluacije modela bilo je potrebno intalirati biblioteke **tabpfn** i **torch** koje su neophodne za pokretanje TabPFN modela.

```
!pip install tabpfn torch --quiet
```

Slika 31: Instalacija potrebnih biblioteka za pokretanje TabPFN modela

Kod koji se tiče treniranja i evaluacije modela dostupan je u dokumentu Ispitivanje_efikasnosti_TabPFN.ipynb:

```
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']))
```

Slika 32: Kod za treniranje i evaluaciju modela

U kodu smo najprije provjerili da li je dosrupan GPU, jer ovaj modela može biti spor i dozvoljava rad sa uzorcima samo do 1 000 instanci ako se radi sa klasičnim CPU-u. Kako smo već putem opcije *Runtime→Change runtime type* izabrali T4 GPU, on je i bio korišten.
Nakon toga smo učitali unaprijed trenirani TabPFN model i trenirali ga na 80% podataka iz podskupa od 10 000 instanci, pri čemu je već objašnjenja uloga funkcije *fit()* u ovom koraku.
Potom je izvršena predikcija nad testnim skupom pomoću funkcije *predict()*.
Nakon predikcija izvršena je evaluacija modela tačnosti korištenjem metričkih funkcija iz biblioteke sckit-learn.

4.3.1 Metrike koje su korištene

Prva i osnovna mjera je tačnost (accuracy), a izračunali smo je pomoću funckije *accuracy_score()*. Tačnost predstavlja odnos između broja tačnih predikcija i ukupnog broja primjera u test skupu.
Međutim, s obzirom na već spomenuti problem disbalasiranosti klasa u našem dataset-u koji je očuvan kroz stratifikovano uzorkovanje, tačnost može biti zavaravajuća.
Stoga smo se odlučile za detaljan izvještaj koji uključuje sljedeće metrike:

- Precision: pokazuje koliko su tačne predikcije određene klase
- Recall: prikazuje koliko je model dobro prepoznao sve stvarne instance određene klase
- F1-score: balans ocjena između precisnosti i odziva
- Support: predstavlja broj stvarnih instanci klase u test skupu, korisno je za razumjevanje težina svake klase

Pored toga, prikazane su i dvije zbirne vrijednosti:

- Macro avg: prosjek svih gore pomenutih metrika po klasama posmatrajući sve klase jednako
- Weigheted avg: prosjek metrika ali uzimajući u obzir broj instanci po klasama

4.3.2 Analiza dobijenih rezultata

```
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
```

Slika 33: Rezultati klasifikacije TabPFN modela na 10 000 instanci

Vrijednost accuracy znači da je model ispravno kasifikovao 86.25% instanci u test skupu.
S obzirom na rizik od neuravnoteženosti klasa i nestandardizacije numeričkih atributa ovo predstavlja dobar rezultat.
Tumačenjem klasifikacionog izvještaja došli smo do sljedećih zaključaka:

- **Precision:**
  Klasa <=50: 88% tačnih predikcija, što je prilično dobar reultat
  Klasa >50K: 77% tačnih predikcija, rezultat je očekivano lošili u odnosu na drugu klasu jer je ova klasa manje dominantna
- **Recall:**
  Klasa <=50K: Ima odziv 0.94, što znači da je model gotovo sve instance ove klase ispravno identifikovao
  Klada >50K: Ima odziv 0.61, što znači da model često ne prepoznaje osobe iz ove klase
- **F1-score:**
  Za klasu <=50 je prilično visok rezultat (0.91), dok je za drugu prilično niži jer model teško detektuje manjinsku klasu.

Na osnovu rezultata evidento je da rizik neizbalansiranosti klasa ima veliki uticaj na rezultate, jer su rezulteti dominantne klase bolji.
S obzirom da **Macro avg** tretira klase jednako, ne uzimajući u obzir njihovu veličinu, množemo uočiti da su njegove vrijednosti (precistion 0.83, recall 0.78 i F1-score 0.80) niže od ukupne tačnosti modela (0.86), što dodatno naglašava disbalans u perfomansama između klasa. Nasuprot tome, **Weighted avg** (0.86 po svim metrikama) pruža realističniji prikaz ukupnih performansi modela, iz razloga što uzima u obzir broj istanci u svakoj klasi, pokazujući da model postiže dobru ukupnu tačnost kada se uzme u obzir veličina svake klase.

4.4 Poređenje rezultata

4.4.1 Poređenje rezultata različitih podskupova

Rezultati klasifikacije za uzorak od 5 000 instanci:

```
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
```

Slika 34: Rezultati klasifikacije TabPFN modela na 5 000 instanci

Rezultati klasifikacije za uzorak od 500 instanci:

```
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
```

Slika 35: Rezultati klasifikacije TabPFN modela na 500 instanci

Na osnovu svih dosada priloženih rezultata zaključujemo:

- ➔ Model TabPFN pokazuje visoku tačnost na svim testiranim podskupovima podataka, što pokazuje da je dobar na opsegu za koji je i napravljen (do 10 000).
- ➔ S obrizom da je tačnost najveća na 500 uzoraka TabPFN potvrđuje da je dizajniran za male do srednje datasetove i da bolje radi na malim skupovima.
- ➔ Ipak najveću pouzdanost i ravnotežu u prepoznavanju obje klase TabPFN postiže na uzorku od 10 000 zbog većeg broja podataka.
- ➔ U svim testovima je imao bolju tačnost za klasu <=50K po svim metrikama, što je posljedica disbalansa klasa što je i napomenuto kao potencijalni rizik.

4.4.2 Poređenje rezultata sa Random Forest modelom

Rezultati klasifikacje pomocu Random Forest za uzorak od 10 000 instanci:

```
Tačnost (accuracy) Random Forest modela na test skupu (10.000 uzorak): 0.8485

Klasifikacioni izvještaj za Random Forest:

              precision    recall  f1-score   support

       <=50K       0.88      0.93      0.90      1518
        >50K       0.73      0.59      0.65       482

    accuracy                           0.85      2000
   macro avg       0.80      0.76      0.78      2000
weighted avg       0.84      0.85      0.84      2000
```

Slika 36: Rezultati klasifikacije RandomForest modela na 10 000 instanci

Rezultati klasifikacje pomocu Random Forest za uzorak od 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
```

Slika 37: Rezultati klasifikacije RandomForest modela na 5 000 instanci

Rezultati klasifikacje pomocu Random Forest za uzorak od 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
```

Slika 38: Rezultati klasifikacije RandomForest modela na 500 instanci

Nakon poređenja rezultata prethodno priloženih možemo zaključiti:

- ➔ TabPFN je pokazao stabilnije performanse na većim uzorcima od RandomForest. Random forest briljira na malom uzorku od 500 instanci i bolji je od TabPFN po svim metrikama, što pokazuje da je ipak Random Forest otrporniji na ograničenu količinu podataka.

- ➔ Kod svih uzoraka, TabPFN je nešto bolji u kalsifikaciji manjinske klase u odnosu na Random Forest. Ovo pokazuje veću otpornost TabPFN-a na disbalans klasa.
- ➔ S obzirom na arhitekturalnu razliku ova dva modela, TabPFN izvlači efikasnije predikcije iz već prethodno naučenog čak i kod malo većih skupova, dok Random Forest zbog svoje arhitekture ima prednost u jednostavnijim klasifikacijama i manjim skupovima.
- ➔ Random Forest je brži i ne zavisi od GPU. TabPFN zahtjevniji model koji sa CPU može raditi sa skupovima do 1 000 instanci, dok sa GPU sa 10 000 i znatno je brži.

4.5 Testiranje TabPFN-a na nepoznatim podacima

U dokumentu Ispitivanje_efiksnosti_TabPFN_nepoznati_podaci.ipynb [17] smo testirali TabPFN na sintetički generisanom datasetu koji je potpuno nepoznat modelu. Naš sintetički dataset ima 1500 uzoraka i simulira podatke o korisnicima mobile aplikacije. Dataframe prikazuje prelaz korisnika na premium paket. Klasifikacijom želimo predvidjeti da li će kosnik preći na premium paket na osnovu njegovih karakteristika i ponašanja (broj sesija, trajanje korištenja, starost, region, itd.)

```
np.random.seed(42)

# Generisanje 500 lažnih korisnika
n_samples = 1500
df_synthetic = pd.DataFrame({
    'age': np.random.randint(16, 60, n_samples),
    'device_type': np.random.choice(['Android', 'iOS', 'Other'], n_samples),
    'session_count': np.random.poisson(lam=15, size=n_samples),
    'avg_session_time': np.random.normal(loc=5, scale=2, size=n_samples).clip(1, 15),
    'region': np.random.choice(['North', 'South', 'East', 'West'], n_samples),
})

# Dodavanje ciljne varijable na osnovu ponašanja
def calculate_premium_prob(session_count, avg_time):
    return min(0.05 * session_count + 0.03 * avg_time, 0.9)

probs = [calculate_premium_prob(s, t) for s, t in zip(df_synthetic['session_count'], df_synthetic['avg_session_time'])]
df_synthetic['is_premium'] = np.random.binomial(1, probs)
```

Slika 39: Generisanje sintetičkih podataka

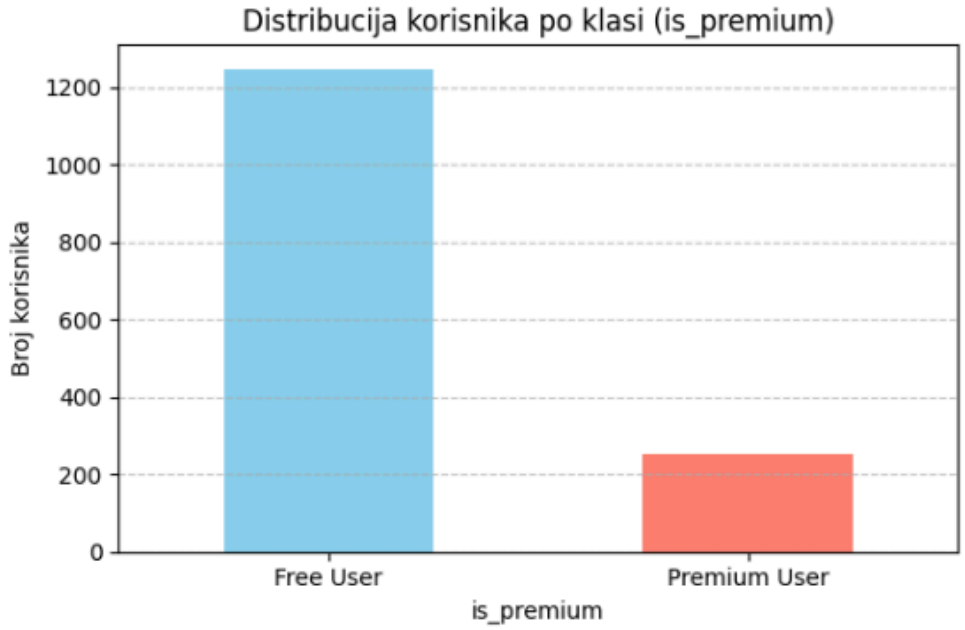
Dataframe posjeduje atribute:

- **age** – Starost korisnika (nasumična vrijednost od 16 do 59 godina)
- **device_type** – Tip uređaja (*Android*, *iOS* ili *Other*)
- **session_count** – Ukupan broj sesija koje je korisnik ostvario, modeliran *Poissonovom distribucijom* sa srednjom vrijednošću 15 (to znači da je očekivana srednja vrijednost 15)
- **avg_session_time** – Prosječno trajanje sesije u minutama, ograničeno na interval od 1 do 15 minuta.
- **region** – Geografska regija korisnika (*North*, *South*, *East* i *West*)
- **is_premium** – Ciljna varijabla koja označava da li korisnik ima premium pretplatu (0 = Free, 1 = Premium)

Atribut is_premium smo generisali na osnovu prosječnog trajanja sesije i broja sesija. Na taj način smo izbjegli da radimo sa potpuno randomiziranim vrijednostima iz kojih zapravo ne bi imalo smisla da model uči.

```
Broj uzoraka po klasi:
is_premium
1    1247
0     253
Name: count, dtype: int64

Procenat po klasi:
is_premium
1    83.13
0    16.87
Name: proportion, dtype: float64
```



Slika 40: Distribucija korisnika

Naš dataframe ima distribuciju 83.13:16.87 u korist besplatnih korisnika, što nam odgovara, jer većina korisnika uglavnom koristi besplatne verzije aplikacija.

4.5.1 Analiza rezultata

TabPFN model je dostigao tačnost od 83.33% na testnom skupu. Na prvi pogled, to izgleda kao dobar rezultat, ali detaljnija analiza pokazuje da nije tako. Model izuzetno dobro prepoznaje korisnike *Free* paketa sa preciznošću od 84% i F1-score-om 0.91, dok *Premium* korisnike nije uspio identificirati sa bilo kakvom pouzdanošću. F1-score za Premium klasu iznosi 0.07, što pokazuje da model gotovo u potpunosti zanemaruje tu klasu.

Model pokazuje ekstremno slabe performanse u klasifikaciji manjinske klase. Rezultat klasifikacije dodatno potvrđuje da TabPFN nije prilagođen radu sa neuravnoteženim podacima.

```
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
```

Slika 41: Rezultat klasifikacije

Faza 5: Cjelokupni osvrt na problem i dobijeno rješenje

Analiza upotrebe TabPFN-a na različitim podskupovima Adult Income dataset-a pokazuje da model ima dobre performanse u rješavanju problema klasifikacije tabelarnih podataka. Najpouzdaniji rezultati su postignuti pri radu sa uzorkom od 10000 redova. To predstavlja i gornju granicu za rad sa modelom u Google Colab okruženju. Za manje datasetove (do 500 uzoraka) smo zaključili da ne radi jednako dobro kao Random Forest, koji je pokazao bolje rezultate. Uočili smo da ima izazove sa nebalansiranim dataset-ovima i da prioritetizira dominantnije klase.
Kroz pet faza analize alata smo pokazali prednosti i ograničenja u odnosu na druge metode poput Rain Forest-a.

5.1 Poređenje sa naučnim radovima iz faze 2

Naziv: ***"TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second"***

Naše istraživanje je potvrdilo da je model sposoban da brzo klasificira podatke bez dodatnog treniranja, koristeći in-context learning na malim dataset-ovima. Navedena efikasnost je naročito došla do izražaja kod dataset-ova veličine između 500 i 5000 redova. Naše istraživanje je odstupalo od naučnog rada kada je u pitanju robustnost modela na razne distribucije. Naši rezultati su pokazali da je model osjetljiv na disbalans klasa. To možemo primijetiti kod osoba sa većim prihodima (klasa >= 50K).

Naziv: ***"Scaling TabPFN: Sketching and Feature Selection for Tabular Prior-Data Fitted Networks"***

Ovaj rad je skrenuo pažnju na problem skalabilnosti modela. Mi smo to na neki potvrdili kroz pokušaje da testiramo modele koji su imali preko 10000 podataka. Svaki pokušaj je rezultirao greškom, zbog čega nismo mogli izvršiti analizu istog unutar Google Colab okruženja zbog memorijskog ograničenja. Jedan od prostora za napredak kod našeg rada je primjena automatske selekcije atributa koju ovaj rad preporučuje, a koju mi nismo implementirali u svom rješenju. Preporučena integracija bi mogla poboljšati robustnost modela, naročito kod prisustva nekonzistentnih karaktera.

Naziv: ***"TabPFN Unleashed: A Scalable and Effective Solution to Tabular Classification Problems"***

Ovaj rad se fokusira na robustnost u pristustvu šuma i disbalansa klasa. Naše ispitivanje je pokazalo da TabPFN ima problema sa nebalansiranim klasama. Klasa >50K biva znatno lošije kvalifikovana u odnosu na dominantniju drugu klasu. To potrvđuje F1-score, koji za klasu <=50 iznosi 0.9, dok za drugu klasu >50K iznosi 0.68. Sličan efekat smo primijetili kod našeg

generisanog dataset-a koji je bio nepoznat modelu. Za taj dataset je F1-score razlika još veća i iznosi 0.91 za dominantnu klasu (Free User) i 0.07 (Premium User). Promjena veličine dataset-a u slučaju našeg generisanog dataset-a bi mogla dovesti do drugačijih rezultata.

Neka ograničenja koja TabPFN posjeduje su već adresirana i djelimično ispravljena kod **TabPFN v2** koji uvodi dinamičko enkodiranje i unaprijeđenu skalabilnost.

Kao unaprjeđenje ovog rada bi mogli dodati sljedeće:

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

Zaključak

TabPFN je napredan alat za klasifikaciju tabelarnih podataka. U određenim scenarijima je pokazao bolje performanse od tradicionalnih modela klasifikacije poput Random Forest-a. S obzirom na to da mnoge industrije koriste tabelarne podatke, javlja se potreba za modelima koji efikasno analiziraju iste. Najistaknutija karakteristika TabPFN-a je upravno in-context učenje koje eliminiše potrebu za dodatnim treniranjem modela. Predstavlja dobro rješenje za problem klasifikacije tabelarnih podataka, jer omogućava brzu primjenu u realnim scenarijima.

Reference

[1] Ranjani Ramamurthy. “Why does data quality matter?”. Medium, https://medium.com/llmed-ai/why-does-data-quality-matter-40540e8727f3

[2] Greg Wiederrecht, Ph.D. “The Convergence of Healthcare and Technology”. RBC Capital Markets, https://www.rbccm.com/en/gib/healthcare/episode/the_healthcare_data_explosion

[3] Will Douglas Heaven.”Hundreds of AI tools have been built to catch covid. None of them helped.” MIT Technology Review, https://www.technologyreview.com/2021/07/30/1030329/machine-learning-ai-failed-covid-hospital-diagnosis-pandemic/

[4]UCI Machine Learning Repository, “Breast Cancer Wisconsin (Diagnostic) Data Set”, Kaggle, https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

[5] Camović Melida, Rokša Amina i Ahmatović Hadija, “Analiza predloženih datasetov-a”. VI projekat, https://colab.research.google.com/drive/1hMGlSCT8Pe7_6wviAQimjdRfu1DRKPIp?authuser=1

[6] UCI Machine Learning Repository, “Pima Indians Diabetes Database”, Kaggle, https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[7] Camović Melida, Rokša Amina i Ahmatović Hadija, “Analiza predloženih datasetov-a”. VI projekat, https://colab.research.google.com/drive/1hMGlSCT8Pe7_6wviAQimjdRfu1DRKPIp?authuser=1

[8] Raj Parmar, “Wine Quality”, Kaggle, https://www.kaggle.com/datasets/rajyellow46/wine-quality

[9] Camović Melida, Rokša Amina i Ahmatović Hadija, “Analiza predloženih datasetov-a”. VI projekat, https://colab.research.google.com/drive/1hMGlSCT8Pe7_6wviAQimjdRfu1DRKPIp?authuser=1

[10] Camović Melida, Rokša Amina i Ahmatović Hadija. “Student performance dataset analiza ”. VI projekat, https://colab.research.google.com/drive/1StVZlc-8bmJRxpcQscr8olnaoCAdUPCw?usp=sharing

[11] Noah Hollmann, Samuel Müller, Katharina Eggensperger, Frank Hutter. “Noah Hollmann∗,1,2 Samuel Müller∗,1 Katharina Eggensperger1 Frank Hutter”. Arxiv, https://arxiv.org/pdf/2207.01848

[12] Benjamin Feuer, Chinmay Hegde, Niv Cohen. “Scaling TabPFN: Sketching and Feature Selection for
Tabular Prior-Data Fitted Networks”. Arxiv, https://arxiv.org/pdf/2311.10609

[13] Si-Yang Liu, Han-Jia Ye. “TabPFN Unleashed: A Scalable and Effective Solution to Tabular Classification Problems”. Arxiv, https://arxiv.org/pdf/2502.02527

[14] Camović Melida, Rokša Amina i Ahmatović Hadija, “Analiza_Adult_Census_dataseta”, VI projekat, https://colab.research.google.com/drive/1kRdJIBfZ_JNOeZa0gseejGmcS8vHIsV7?authuser=1

[15] Camović Melida, Rokša Amina i Ahmatović Hadija, “Ispitivanje_efikasnosti_TabPFN”, VI projekat, https://colab.research.google.com/drive/1BxVALM-o_wiTgMByTSHbe2vShTOrtnX_?authuser=1
[16] N. Hollmann, S. Muller, K. Eggensperger, F. Hutter , “TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Single Forward Pass”, *arXiv preprint arhXiv: 2207.01848,* https://arxiv.org/abs/2207.01848
[17] Camović Melida, Rokša Amina i Ahmatović Hadija. “Ispitivanje TabPFN-a na nepoznatim podacima”. VI projekat, https://colab.research.google.com/drive/1kEkbPyTKBY4m7nJHqgaL9hb3nAShqwty?usp=sharing