

Projekat iz predmeta Mašinsko učenje 2024/25

Projekat šaljete na mail: abolic@pmf.unsa.ba. Obavezno u naslovu naglasiti "ML projekat". Projekat se može braniti u dva termina:

- 26.06.2025. (četvrtak) – u tom slučaju rok za slanje projekta je 23.06. do 00:00h
- 10.07.2025. (četvrtak) – u tom slučaju rok za slanje projekta je 07.07. do 00:00h

Kašnjenje sa slanjem projekta se neće tolerisati!

U svim projektima potrebno je uraditi sljedeće cjeline:

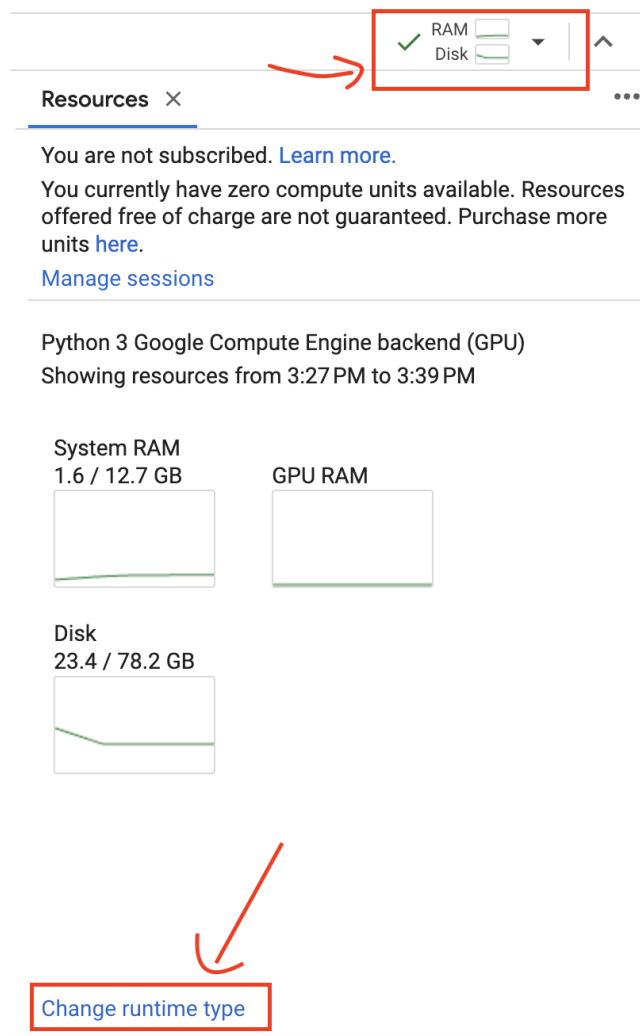
1. Eksploracija podataka - slično kao što smo radili na vježbama, učitavanje podataka, eventualno čišćenje podataka, podjelu na train/test/validation skupove, istraživanje koje featurese dataset ima, koji bi featuri mogli biti bitni, a koji ne, korelaciju između featura, razni grafici, selekcija featuresa za treniranje, itd. itd. U drugoj temi ovaj dio treba sadržati i informacije (ili kod) o načinu skupljanja podataka
2. Treniranje i upoređivanje modela - trebate istrenirati bar **dvije** različite arhitekture modela (linearni model, polinomijalni modeli, SVM, neuralne mreže ili koji god drugi model mislite da ima smisla koristiti, s tim da ako koristite neki model, morate bar okvirno znati kako funkcionise) na vašem datasetu. Ovaj dio treba sadržavati kod za treniranje modela, praćenje preciznosti i funkcije gubitka modela i šta god drugo smatrate važnim za vaš model
3. Evaluacija modela - ovaj dio sadrži razne evaluacije modela nakog njegovog treniranja, npr. preciznost i vrijednost funkcije gubitka na svim skupovima podataka koje imate, confusion matricu, da li je došlo do overfittinga/underfittinga i generalno diskusiju o rezultatima modela, kako bi se mogao eventualno model poboljšati, zašto ne radi dobro na nekim podacima itd.

Ako kod pišete u Notebooku, ne zahtijevam pisanje dokumentacije u posebnom fajlu, nego to možete raditi u unutar notebooka pomoću *markdown* ćelija. Ako ne koristite Notebook, napišite dokumentaciju u posebnom fajlu.

Treniranje i učitavanje podataka

Za treniranje modela možete koristiti svoj računar/laptop, s tim da to može biti jako sporo ili čak nemoguće ukoliko imate slabiji laptop. Također, većina laptopa nema dobru grafičku karticu na kojoj možete trenirati veće modele sa puno podataka. Pored svog laptopa, modele možete trenirati i na Google Colab (kojeg smo dosta koristili i na vježbama) koji nudi opciju besplatnog treniranja na njihovim GPU-ovima (naravno resursi su limitirani te je moguće da nekad nemaju na raspolaganju neki GPU, mada se to rijetko dešava). Jedna mana je što morate paziti da ne izgubite konekciju sa Colab, jer će se u tom slučaju treniranje prekinuti. Ovaj problem možete riješiti tako što često spašavate model, pa i ako se treniranje prekine, možete nastaviti treniranje istog modela (naravno prije toga morate učitati taj

model). Izvršavanje notebook-a na GPU možete uključiti tako što kliknete na ikonicu za resurse, a zatim u donjem lijevom uglu na “Change runtime type” (pogledati sliku ispod).



The screenshot shows the 'Resources' panel in Google Colab. At the top, there is a header with a green checkmark, 'RAM', 'Disk', and a dropdown arrow. A red arrow points to this header. Below the header, the text reads: 'Resources X', 'You are not subscribed. [Learn more.](#)', 'You currently have zero compute units available. Resources offered free of charge are not guaranteed. Purchase more units [here.](#)', and '[Manage sessions](#)'. Below this, it says 'Python 3 Google Compute Engine backend (GPU)' and 'Showing resources from 3:27 PM to 3:39 PM'. There are three graphs: 'System RAM 1.6 / 12.7 GB', 'GPU RAM', and 'Disk 23.4 / 78.2 GB'. At the bottom, a red arrow points to a button labeled 'Change runtime type'.

Resources X

You are not subscribed. [Learn more.](#)

You currently have zero compute units available. Resources offered free of charge are not guaranteed. Purchase more units [here.](#)

[Manage sessions](#)

Python 3 Google Compute Engine backend (GPU)

Showing resources from 3:27 PM to 3:39 PM

System RAM
1.6 / 12.7 GB

GPU RAM

Disk
23.4 / 78.2 GB

[Change runtime type](#)

Nakon toga promijenite hardware accelerator na GPU i selektujte neki od ponuđenih GPU-ova.

Notebook settings

Hardware accelerator
GPU ⌵ ?

GPU type
T4 ⌵

Want access to premium GPUs? [Purchase additional compute units](#)

☐ Omit code cell output when saving this notebook

Cancel Save

Da ne biste morali svaki put uploadati dataset na Colab (a pogotovo u slučaju datasea u projektu 1) možete se preko notebook-a spojiti na Google Drive i na taj način učitavati podatke direktno sa drive (pri tome ćete morati jednom podatke uploadati na drive). Uspostavljanje veze i čitanje nekog fajla sa drive je demonstrirano u kodu ispod:

```

▶ from google.colab import drive
  drive.mount('/content/drive')

  with open('/content/drive/MyDrive/testni folder/test.txt', 'r') as f:
    print(f.readlines())

```

Mounted at /content/drive
['Test!']

Na sličan način možete i spašavati fajlove direktno na drive (istražite na internetu kako se to radi, ukoliko vam to treba). Ako ne možete pronaći putanju do nekog fajla, možete pomoću “glob” biblioteke saznati koji se sve fajlovi nalaze u nekom folderu i na taj način saznati putanju do nekog fajla, kao u sljedećem kodu:

```

✓ 0s ▶ import glob
      import os.path as osp
      glob.glob(osp.join('/content/drive/MyDrive/testni folder', '*'))

```

['/content/drive/MyDrive/testni folder/rad.gsheets',
'/content/drive/MyDrive/testni folder/coco_test.sample.json',
'/content/drive/MyDrive/testni folder/test.txt']

Tema 1 (max 25b)

U ovoj temi radit ćete sa medicinskim skupom podataka koji sadrži informacije o pacijentima, uključujući demografske podatke, kliničke nalaze i životne navike. Vaš zadatak je da izgradite modele mašinskog učenja koji će predviđati da li je pacijent pod rizikom od srčanog zastoja.

Ciljna kolona je HeartDisease, koja predstavlja binarnu varijablu (0 – nema srčane bolesti, 1 – ima srčanu bolest). Na osnovu ostalih atributa, vaš zadatak je da napravite **klasifikacijske modele** koji predviđaju ovu varijablu. Uporedite performanse više modela i evaluirajte ih korištenjem odgovarajućih metrika (npr. accuracy, precision, recall, F1-score, ROC-AUC, ...).

Podaci se nalaze na linku: https://drive.google.com/file/d/1QeQ-e_ubkb-xcXXI22QLwhf65I2-giuO/view?usp=sharing

Dataset sadrži jedan fajl *heart.csv*, a kolone unutar tabele su sljedeće:

- Age (float): godine pacijenta
- Sex (int): spol pacijenta (0 – ženski, 1 – muški)
- ChestPainType (string): tip bola u grudima (TA – tipična angina, ATA – atipična angina, NAP – neanginalni bol, ASY – asimptomatski)
- RestingBP (int): krvni pritisak u mirovanju (mm Hg)
- Cholesterol (int): nivo holesterola u krvi (mm/dl)
- FastingBS (int): da li je nivo šećera u krvi “na prazan stomak” > 120 mg/dl (0 – ne, 1 – da)
- RestingECG (string): rezultat EKG-a u mirovanju (Normal, ST, LVH)
- MaxHR (int): maksimalni broj otkucaja srca tokom testa
- ExerciseAngina (string): da li pacijent ima anginu izazvanu vježbanjem (Y – da, N – ne)
- Oldpeak (float): ST depresija izazvana vježbanjem u odnosu na odmor
- ST_Slope (string): nagib ST segmenta (Up, Flat, Down)
- HeartDisease (int): ciljna varijabla – 1 ako pacijent ima srčanu bolest, 0 ako nema

Na vama je da napravite odgovarajući **split podataka** (obavezno train/test podjela) i da izvršite **predprocesiranje** podataka gdje je to potrebno (npr. kodiranje kategoričkih varijabli, skaliranje numeričkih vrijednosti, itd).

Tema 2 (max 30b, može se raditi u parovima od po 2 studenta, ili sami)

Cilj je napraviti model koji vrši predikciju cijene polovnih mobitela. Podatke je potrebno sam skupiti, npr. sa stranice olx.ba (ili nekih drugih stranih ili lokalnih stranica za prodaju polovnih stvari). Za to ćete morati istražiti kako funkcionišu query-i za traženje artikala na odgovarajućoj stranici (bit će vam korisna “Inspect” opcija i “Network” tab u Google Chrome pretraživaču npr.). Također, mogu vam pomoći sljedeći kodovi (koji su vjerovatno stari i ne rade više, ali mogu pomoći sa osnovnom logikom kako napraviti Python program koji kupi informacije sa stranice):

- <https://github.com/DustTheory/olxba-py>
- <https://github.com/fajicbenjamin/olxba-search-scheduler>
- <https://github.com/daholino/OLXbaSurfer>

Sami odlučite koje parametre koristiti za treniranje, te razdvojite podatke na skup za treniranje i testiranje. Imajte na umu da ne smije doći do ponavljanja podataka u trening i testnom skupu, te da trebati imati slične distribucije po parametrima u oba skupa.

Tema 3 (slobodna tema, max broj bodova ovisi u težini problema)

U sklopu ove teme možete raditi neki ML problem koji ste sami smislili, našli na internetu i slično. Pri tome, prije nego krenete raditi na projektu, trebate dobiti odobrenje od predmetnog asistenta da li je projekat ok i informaciju koliko maksimalno bodova možete dobiti na tom projektu. Ako odaberete ovu temu, javite mi se na teamsu, gdje ćemo se dalje dogovoriti.