

Univerza v Ljubljani
Fakulteta za matematiko in fiziko



Klasifikacija zvezdnih spektrov

Avtor: Simon Perovnik

Predavatelj: prof. dr. Borut Paul Kerševan

Asistent: Gregor Traven

Druga naloga pri Praktikumu strojnega učenja v fiziki

Ljubljana, november 2022

1 Naloga

Pomembna metoda astronomskega raziskovanja vesolja je t.i. astronomski spektroskopija pri kateri preučujemo spekture svetlobe nebesnih teles. Za opazovanje so tako zelo zanimive zvezde, katerih spektri so si lahko zaradi mnogih fizikalnih vplivov in pa samih lastnosti zvezd precej različni. V nalogi bomo analizirali podatke, pridobljene v sklopu projekta GALAH, analizirali pa bomo vidni del spektrov 10000 zvezd.

Napoved spektra poljubne zvezde je v splošnem zelo zahtevna naloga, saj je veliko dejavnikov, ki vpliva na karakteristike spektra. Najpomembnejši izmed teh so: temperatura zvezde, kovinskost, gravitacijski pospešek na površju zvezde, radialna hitrost, rotacijska hitrost, turbulanca, magnetna polja, plin okoli zvezde, vezanost v sistem dvojne (ali trojne) zvezde in absorpcija svetlobe v medzvezdnem prostoru ter v Zemljini atmosferi.

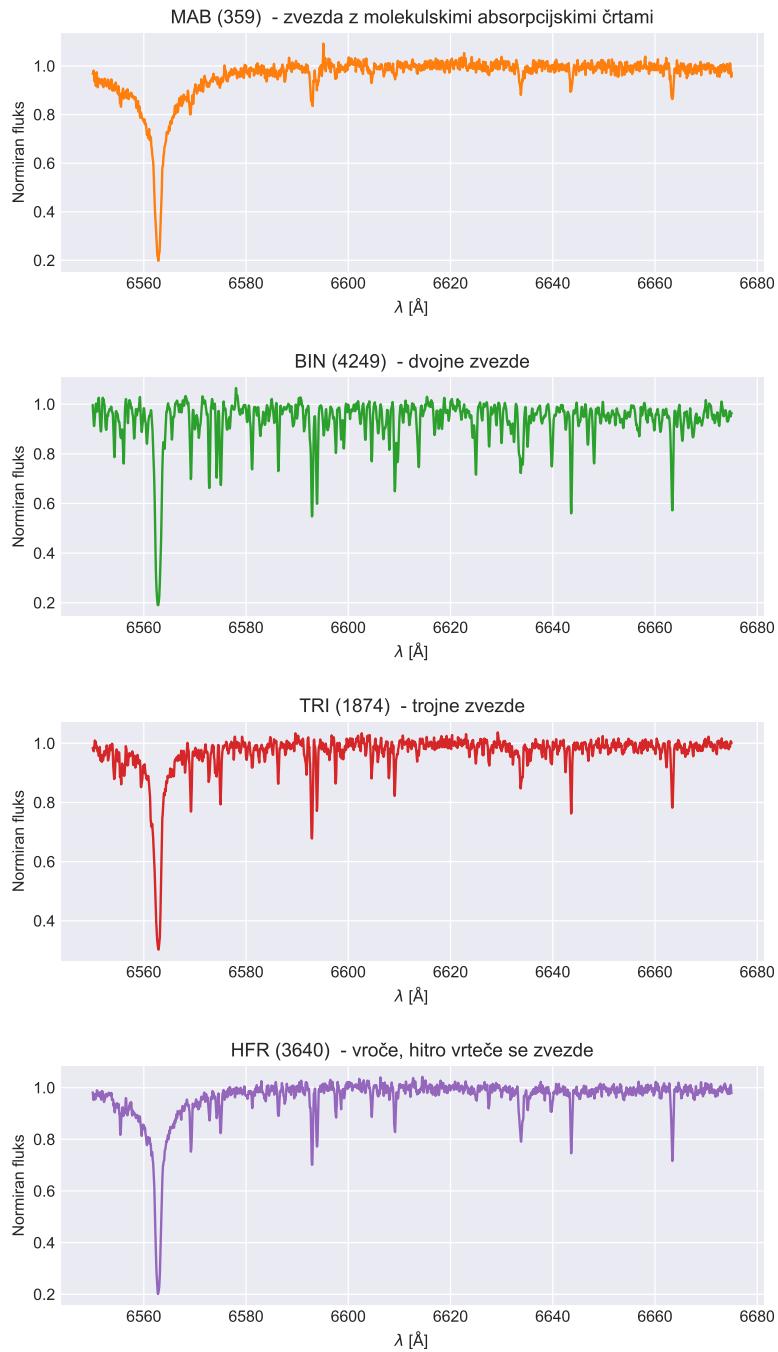
V nalogi bomo s pomočjo metod strojnega učenja analizirali spekture zvezd in jih poskusili kategorizirati med že poznane tipe zvezd, ki so nam podani v navodilu. Pravtako bomo poskusili poiskati kakšne nove gruče podobnih si zvezd in napovedati, kaj bi bil vzrok njihove podobnosti oziroma za kakšne zvezde gre.

2 Podatkovni set

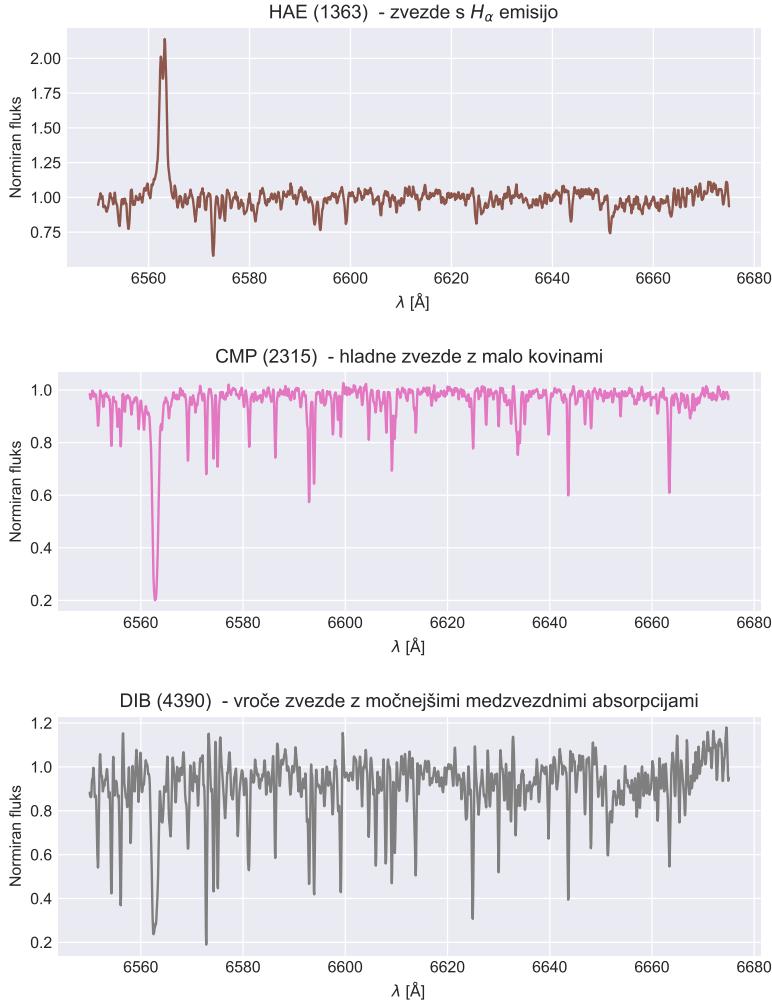
Spektri projekta GALAH so nam podani kot .txt datoteke s 2084 vrsticami oziroma vrednosti fluksa pri različnih valovnih dolžinah. Podatki so že normirani (vrednost kontinuma je 1) in popravljeni glede na Dopplerjev premik zaradi radialnih hitrosti zvezd. Poleg spektrov imamo na voljo tudi seznam že kategoriziranih spektrov, kjer imamo za 57 zvezd že podano klasifikacijsko geslo. V tem naboru med sabo ločimo:

- MAB - zvezde z molekulskimi absorpcijskimi črtami.
- BIN - zvezde vezane v sistem dvojne zvezde.
- TRI - zvezde vezane v sistem trojne zvezde.
- HFR - vroče, hitro vrteče se zvezde.
- HAE - zvezde z emisijo H_{α} .
- CMP - hladne zvezde z malo kovinami.
- DIB - vroče zvezde z močnejšimi medzvezdnimi absorpcijami.

Poglejmo si nekaj najbolj tipičnih predstavnikov posamezne kategorije.



Slika 1: Že kategorizirani spektri.



Slika 2: Že kategorizirani spektri.

3 Analiza poglavitnih komponent (PCA)

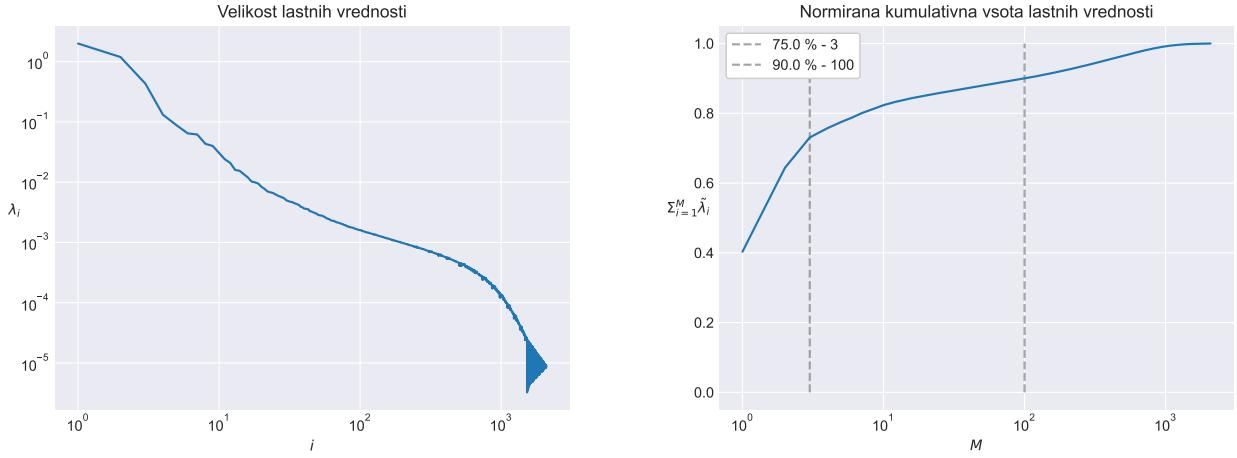
Metoda glavnih komponent (Principal component analysis – PCA) je ena od osnovnih metod zmanjševanja števila dimenzij. S to metodo poiščemo lastne vektorje in lastne vrednosti našega podatkovnega seta. Podatkovni set lahko nato predstavimo kot projekcijo na prostor, ki ga napenjajo lastni vektorji s pripadajočimi najbolj zastopanimi lastnimi vrednostmi kovariančne matrike

$$C = \frac{1}{n-1} B^* B. \quad (1)$$

Matrika B je matrika centriranih podatkov in ima dimenzijs $n \times p$, kjer je n število spektrov (10000), p pa število točk spektra (2084). Pri projekciji spektrov na nižje dimenzijs bi se radi v čim večji meri znebili šuma, pri tem pa obdržali večino koristnih informacij. Ta meja je lahko zelo nejasna, zato ne vemo točno, kakšno število lastnih vektorjev moramo upoštevati pri projekciji na nižje dimenzionalni prostor, da bo rezultat najboljši. Navodila nam kot možno rešitev ponujajo vpeljavo t. i. energije, ki je definirana kot

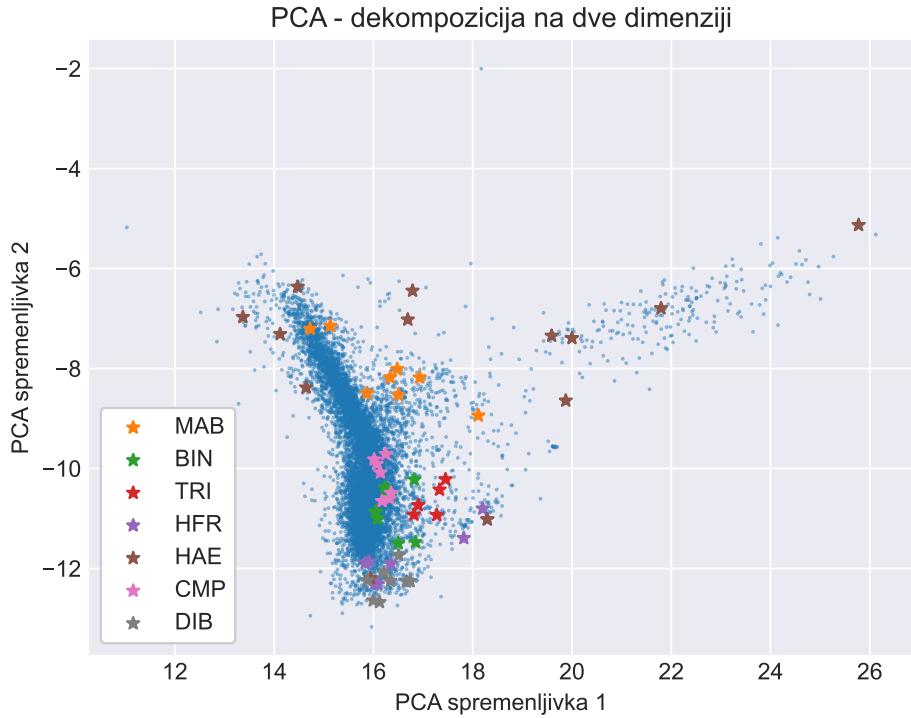
$$g_j = \sum_{k=1}^j \lambda_k. \quad (2)$$

V tem kontekstu lahko raziščemo, kako velike lastne vrednosti dobimo po razcepnu, in koliko jih potrebujemo, da ohranimo npr. 90% energije.



Slika 3: Velikosti lastnih vrednosti (levo) in normirana kumulativna vsota glede na število lastnih vrednosti.

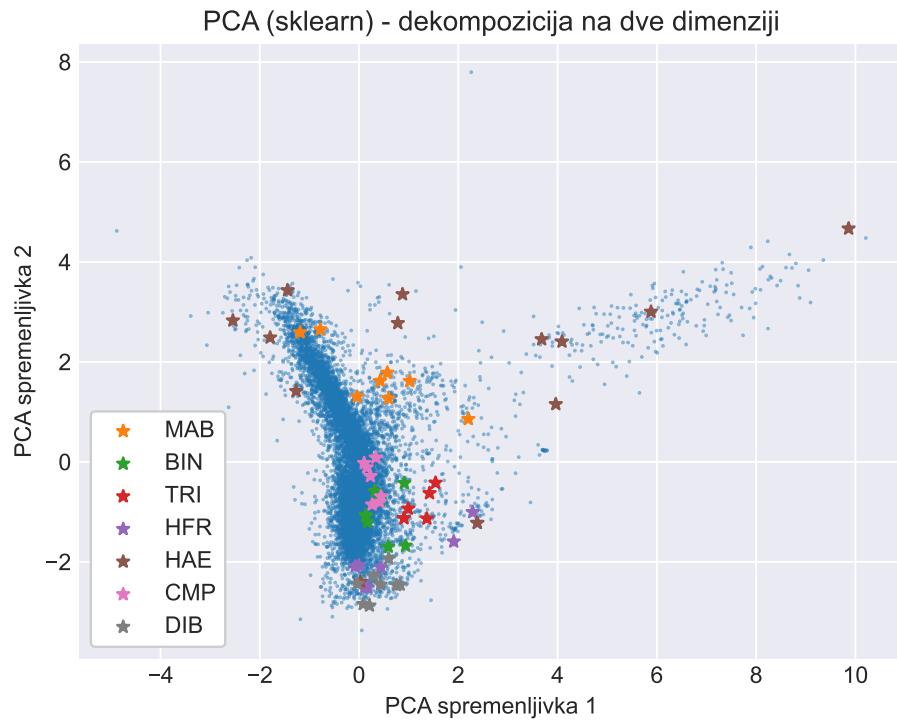
Velikost lastnih vrednosti razmeroma hitro znatno pada, saj je že enajsta po vrsti kar dva velikostna reda manjša od prve. Pri zadnjih lastnih vrednostih lahko opazimo efekt 'iznihanja', saj so začele le te sunkovito oscilirati. Z desnega grafa lahko vidimo, da že z zgolj tremi lastnimi vrednostmi zaobjamemo 75% energije. V nadaljevanju si najprej poglejmo, kakšen razcep dobimo, če reduciramo sistem na zgolj dve dimenziji (tako ga bomo lahko enostavno vizualizirali).



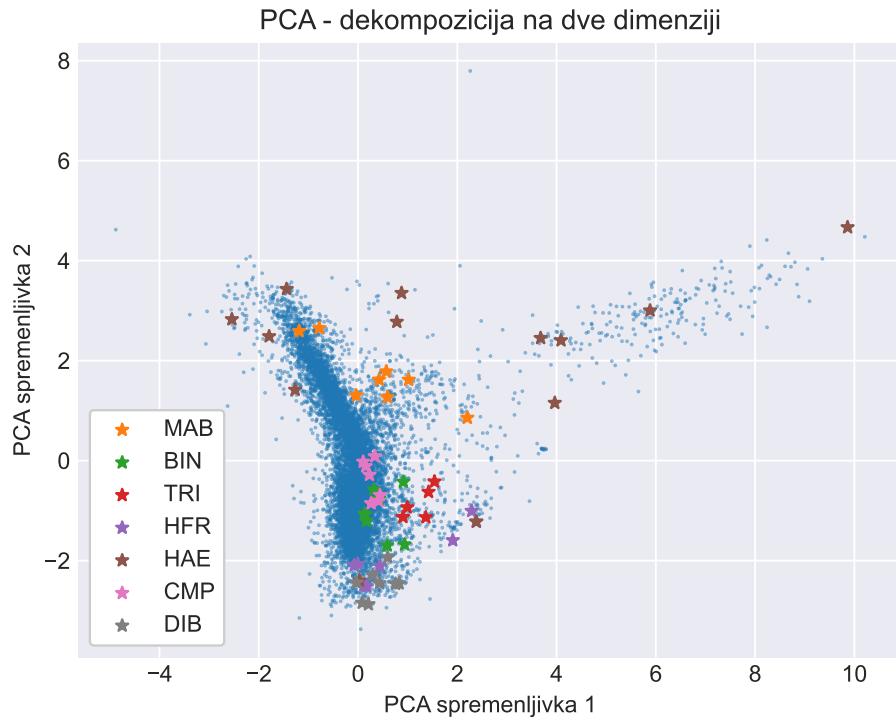
Slika 4: Dekompozicija spektrov na zgolj dve komponenti - ročna implementacija, glej dodatek.

3.1 Primerjava z implementacijama iz knjižnice

Da bi preverili, ali naša ročno implementirana metoda pravilno deluje jo lahko primerjamo z dvema implementacijama iz knjižnice **scikit learn** - **sklearn.decomposition.PCA** in **sklearn.decomposition.KernelPCA**.



Slika 5: Implementacija PCA iz knjižnice Scikit learn.

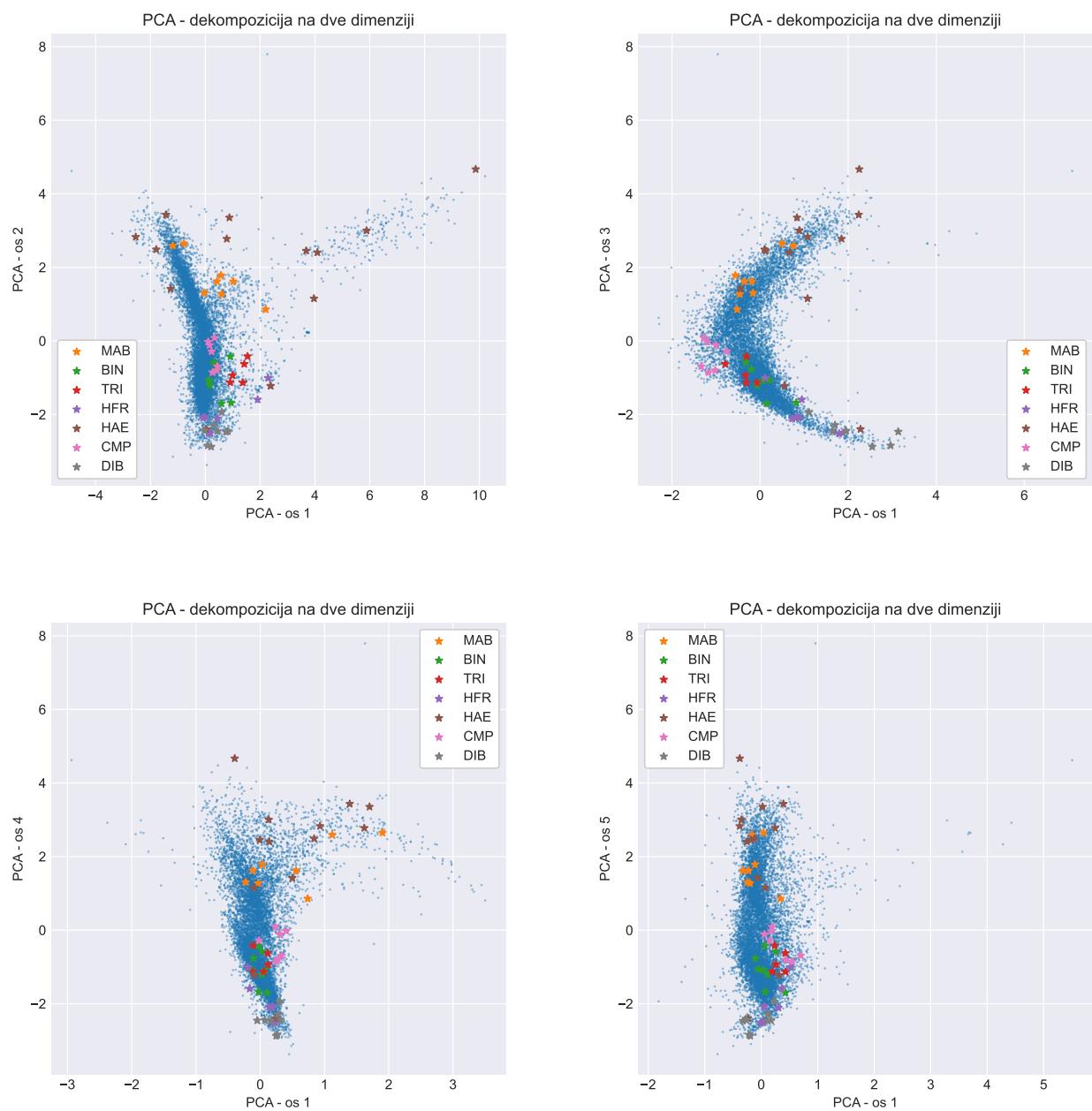


Slika 6: Implementacija KernelPCA iz knjižnice Scikit learn.

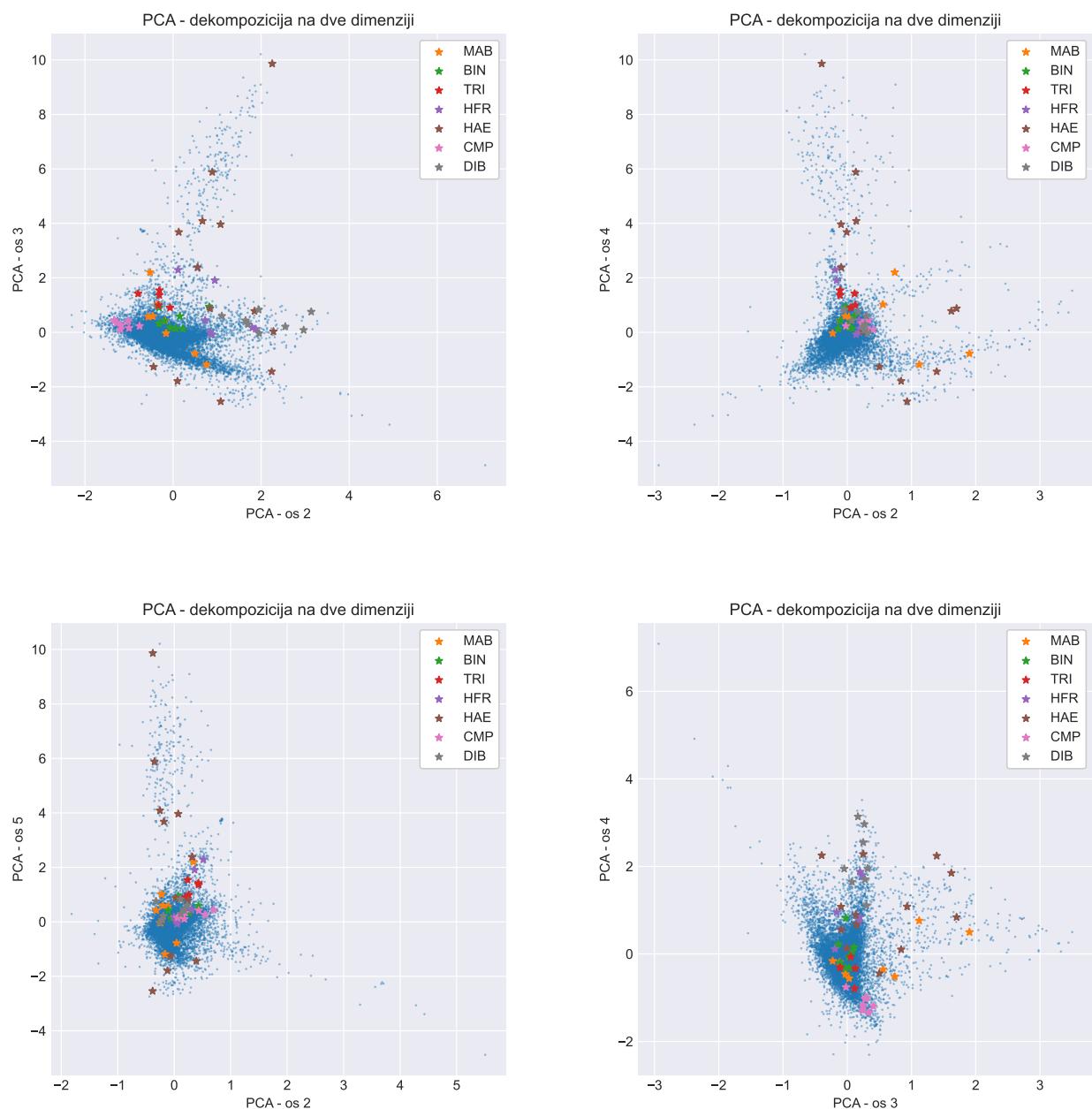
Vse tri metode nam dajo povsem identičen rezultat, kar je malce presenetljivo, saj bi pričakovali, da bomo zardi optimizacijskih metod dobili pri implementacijah iz knjižnice scikit learn dobili vsaj nekoliko drugačne rezultate.

3.2 Projiciranje v več dimenzij

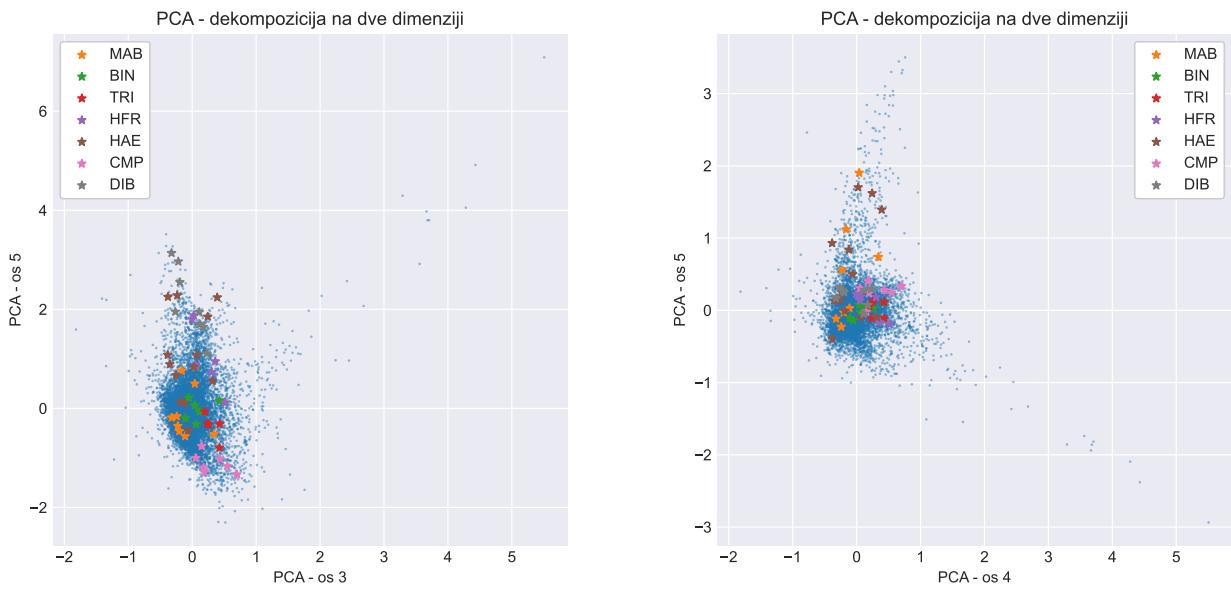
V želji, da bi dobili močnejšo gručenje med posameznimi kategorijami zvezd, si lahko pogledamo nekaj rezultatov redukcije na 5 najpomembnejših lastnih osi. Vizualizirali jih bomo na način parskih projekcij na posamezne osi.



Slika 7: Projiciranje pri dekompoziciji na 5 glavnih osi (1/3).



Slika 8: Projiciranje pri dekompoziciji na 5 glavnih osi (2/3).

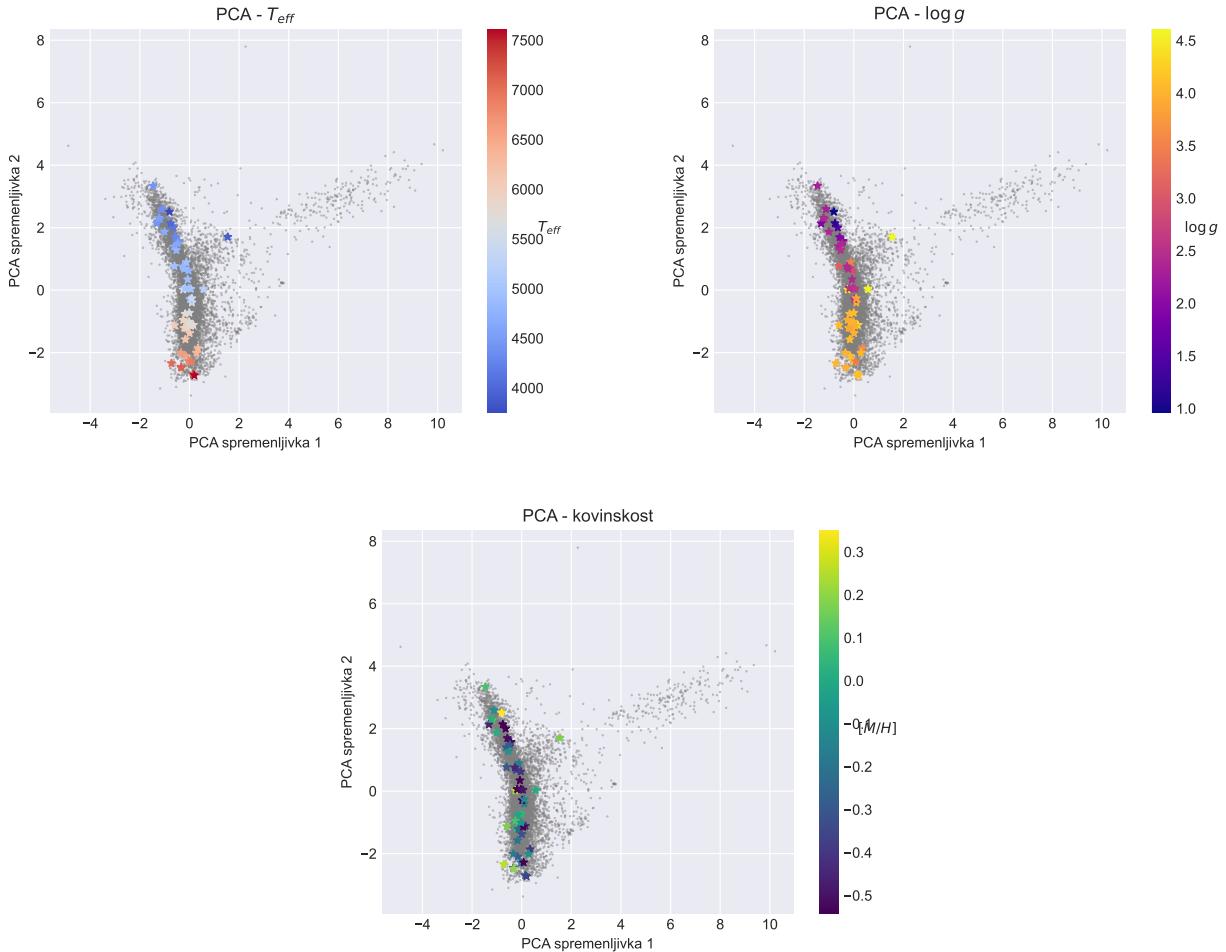


Slika 9: Projiciranje pri dekompoziciji na 5 glavnih osi (3/3).

- Večino projekcij si je kvalitativno zelo podobnih - po večini ni opaziti nobenega močnega gručenja. Še najbolj zanimive rezultate lahko opazimo pri projekciji na osi 1 in 5, kjer se različne kategorije zvezd nekoliko poravnajo v premico in se po njej razklopijo, a tudi tu prihaja do močnega prekrivanja med posameznimi gručami.
- Projekcije med osmi višjega indeksa nam podajo manj informacij o morebitnih gručah, kar se sklada s predpostavko, da lahko z nekaj glavnimi osmi dobro opišemo varianco sistema.
- Točke so v vseh projekcijah tako nagnetene skupaj, da ne moremo prepoznati nobenega trenda ali gručenja, ki bi nakazovalo na kakšno še ne opisano kategorijo zvezd.

3.3 Vpliv zvezdnih parametrov pri končni dekompoziciji.

V uvodnem delu naloge smo definirali nekaj fizikalnih lastnosti, ki dobro definirajo spekter posamezne zvezde. Raziščemo lahko, ali je kakšen od teh parametrov pomemben (ima veliko varianco) tudi v močno reduciranim sistemu - za enostavnost vizualizacije znova reduciramo na dve dimenziji.



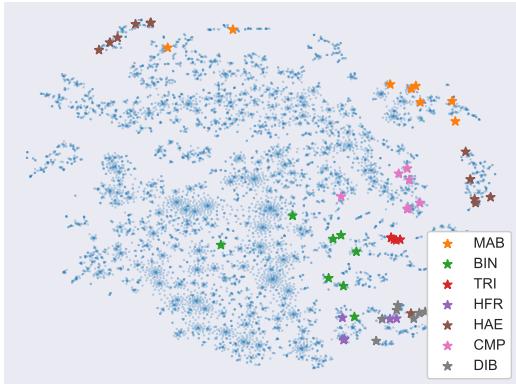
Slika 10: Porazdeljenost zvezdnih parametrov T_{eff} (efektivna temperatura), $\log g$ (gravitacijski pospešek) in $[M/H]$ (kovinskost).

- Efektivna temperatura je močno povezana s PCA spremenljivko 2.
- Vpliv gravitacijskega pospeška na površju se še vedno znatno opazi v končni dekompoziciji, a ima raztros manj jasen trend kot temperatura.
- Zdi se, da informacija o kovinskosti ne nastopa v končni dekompoziciji, saj ni mogoče opaziti nobenega trenda v projekcijski ravnini.

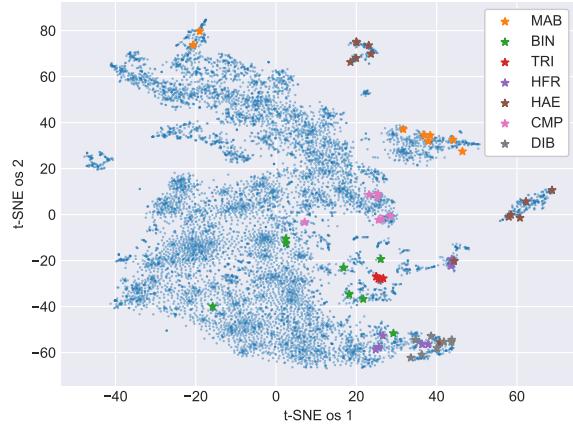
4 Dekompozicija t-SNE

T-SNE je metoda, ki je zelo učinkovita pri vizualizaciji visoko dimenzionalnih podatkov. Ker je t-SNE bolj učinkovita od PCA pri vizualizaciji, bomo podatke zreducirali kar na dve dimenziji. Pri 2084 dimenzijah je ta algoritmom še kar zamuden, zato bomo s PCA najprej podatke zreducirali na npr. 100 dimenzij nadalje pa z metodo t-SNE na 2 dimenziji. Izkaže se, da sta parametra ki najbolj vplivata na končno sliko 'perplexity' (p) in hitrost učenja. Če nastavimo preveliko hitrost učenja so točke na koncu preveč razpršene, zato je težje opaziti gruče spektrov, ki bi lahko spadali v isto kategorijo. Parameter perplexity pa določa, koliko najbližjih sosedov (najbolj podobnih spektrov) algoritom upošteva pri izračunu.

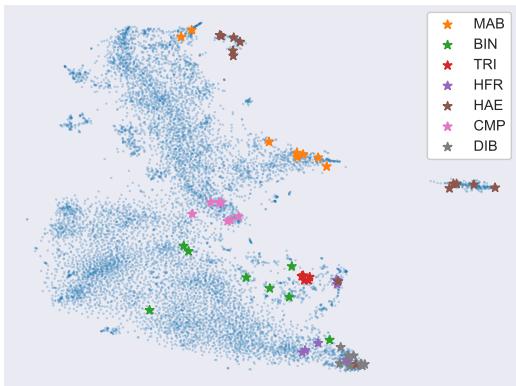
t-SNE - dekompozicija na dve dimenziji - KL = 2.556
preplexity = 5



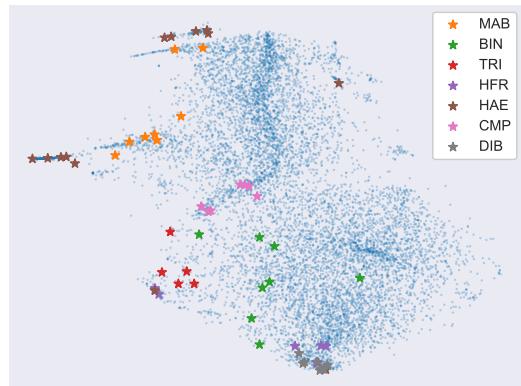
t-SNE - dekompozicija na dve dimenziji - KL = 2.144
preplexity = 20



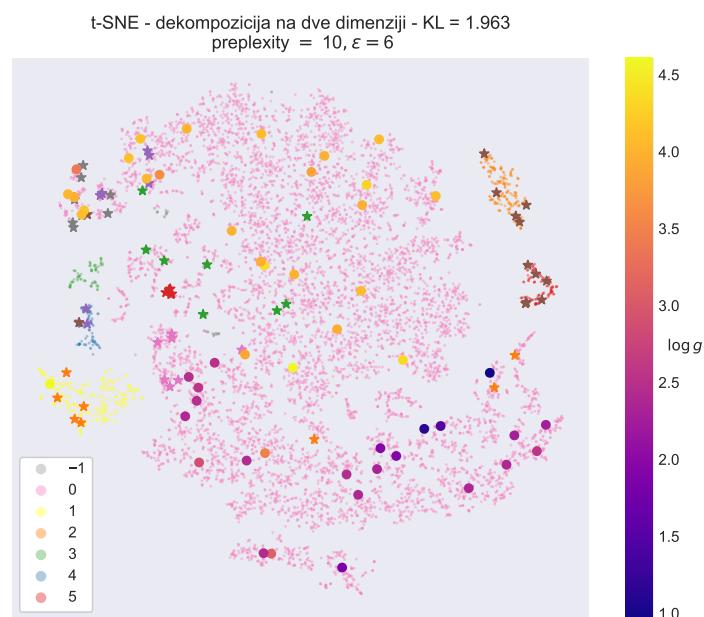
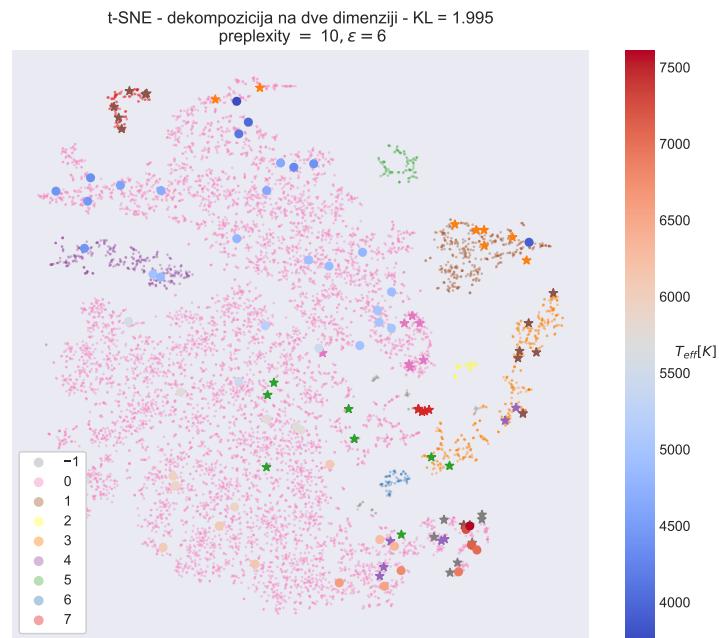
t-SNE - dekompozicija na dve dimenziji - KL = 1.463
preplexity = 100



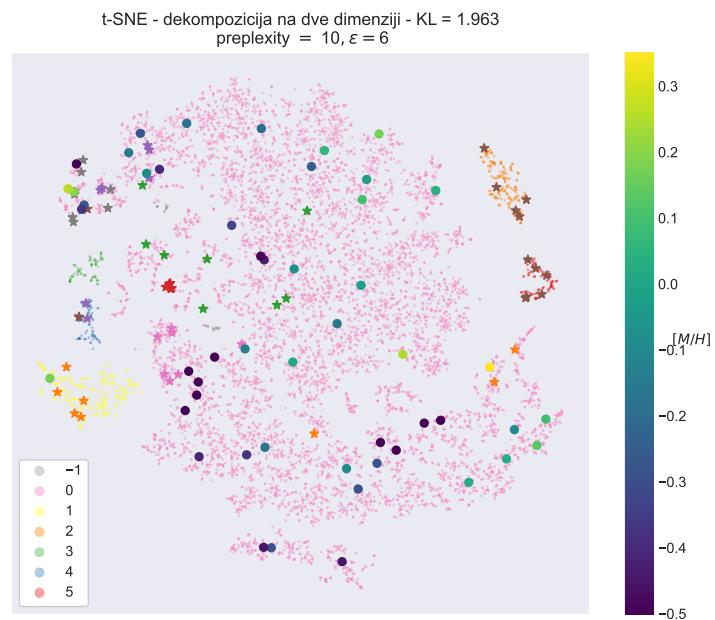
t-SNE - dekompozicija na dve dimenziji - KL = 0.511
preplexity = 1000



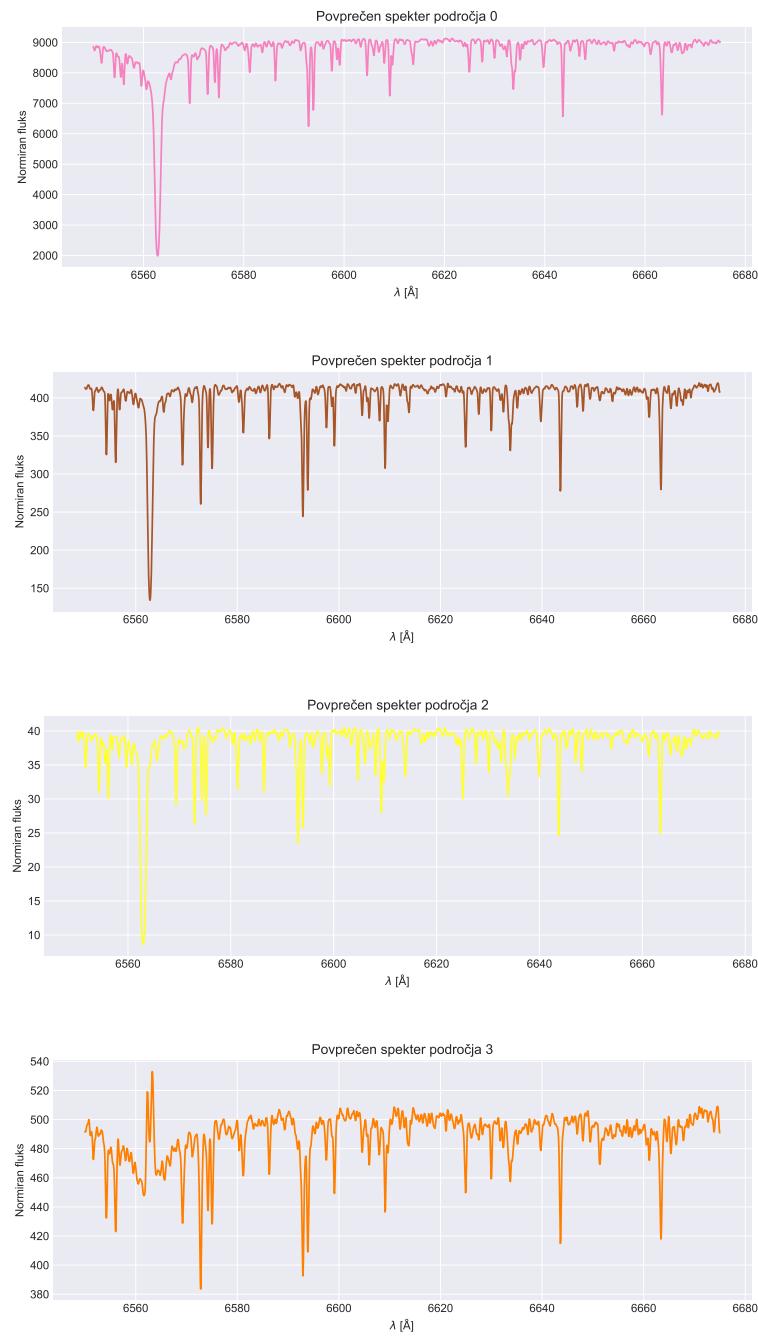
Slika 11: Dekompozicija z metodo t-SNE pri različnih vrednostih parametra preplexity (p).



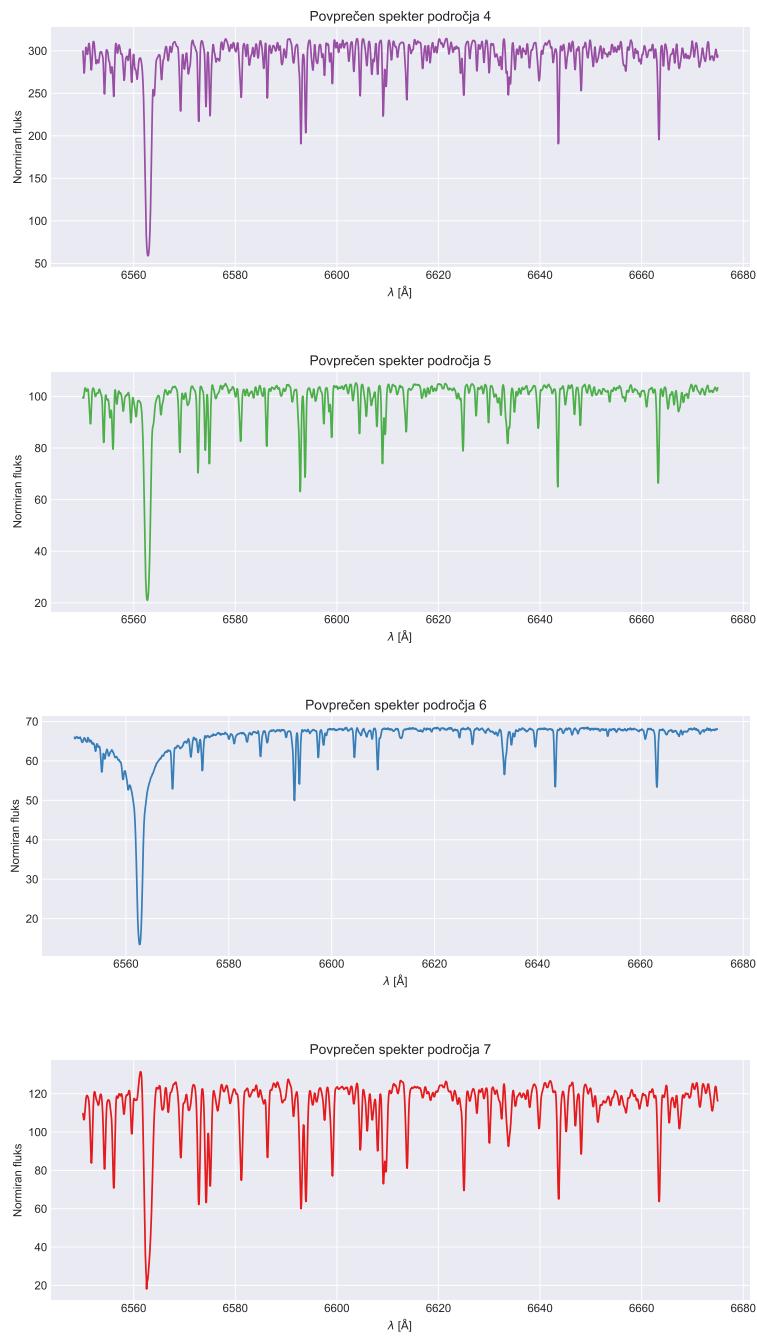
Slika 12: Gručenje z uporabo algoritma DBSCAN. Na levi so prikazane tudi točke z znano efektivno temperaturo, na desno pa z znanim gravitacijskim pospeškom na površju.



Slika 13: Gručenje z uporabo algoritma DBSCAN. Prikazane so tudi kovinskosti nekaterih zvezd.



Slika 14: Spektri pripadajočim gručam dobljeni z algoritmom DBSCAN (1/2).

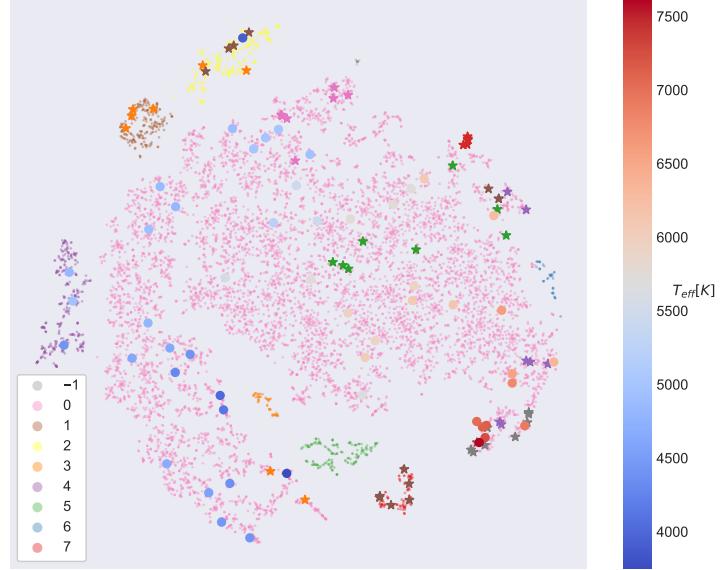


Slika 15: Spektri pripadajočim gručam dobljeni z algoritmom DBSCAN (2/2).

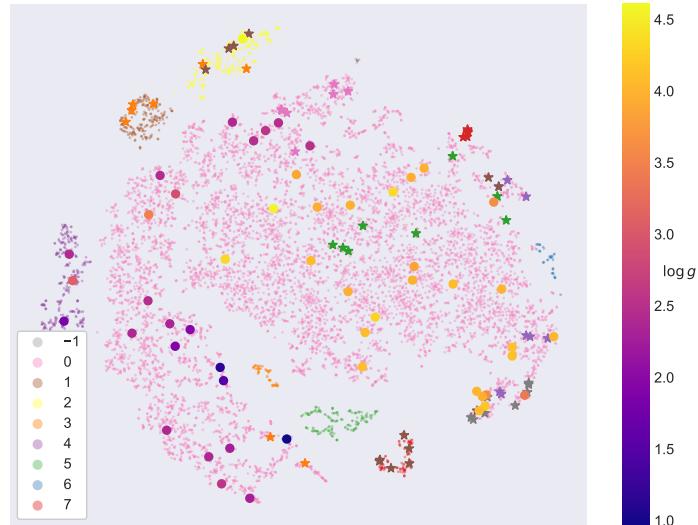
5 Analiza okrnjenega spektra

Za konec si oglejmo še kategorizacijo zvezd, če vsem spektrom odrežemo prvih 240 točk in na ta način ne zajamemo značilne absorpcijske črte H_{α} .

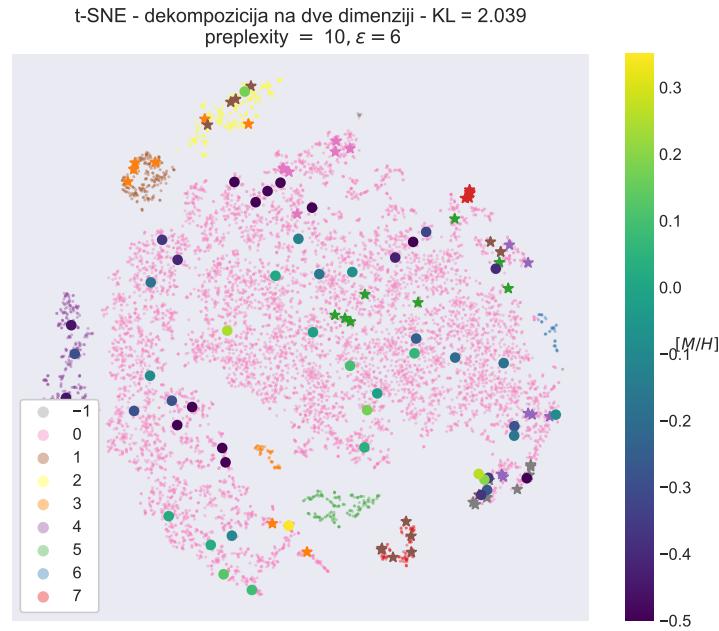
Okrnjen spekter - t-SNE - dekompozicija na dve dimenziji - $KL = 2.039$
preplexity = 10, $\varepsilon = 6$



t-SNE - dekompozicija na dve dimenziji - $KL = 2.039$
preplexity = 10, $\varepsilon = 6$



Slika 16: Gručenje z uporabo algoritma DBSCAN pri analizi okrnjenih podatkov. Na levi so prikazane tudi točke z znano efektivno temperaturo, na desno pa z znanim gravitacijskim pospeškom na površju.



Slika 17: Gručenje z uporabo algoritma DBSCAN pri analizi okrnjenih podatkov. Prikazane so tudi kovinskosti nekaterih zvezd.

6 Dodatek

```

1 import scipy.sparse.linalg as linalg
2 import numpy as np
3
4 def PCA(X, 1):
5     N = len(X)
6     B = X - np.mean(X, axis=0)           # centering data
7     C = 1/(N-1) * np.dot(B.T, B)         # covariance matrix
8     D, V = linalg.eigs(C, 1)             # find 1 highest eigenvalues
9     T = np.dot(X, V)                   # project data to 1 dimensions
10    return T

```

Slika 18: Implementacija analize poglavitnih komponent v Python-u.