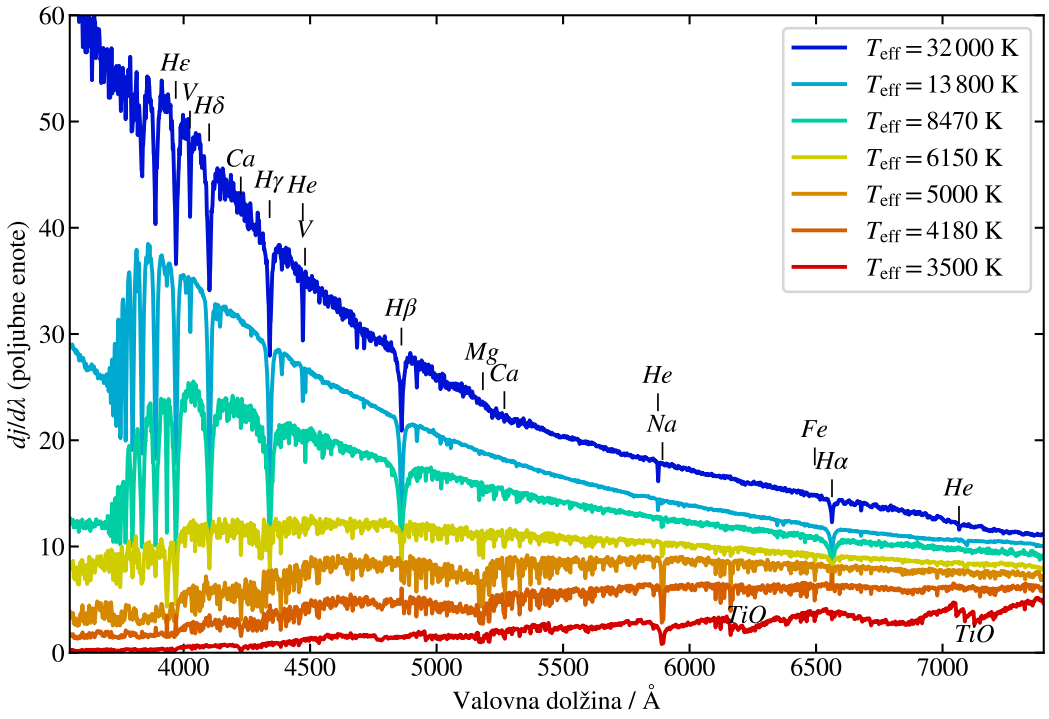
2. naloga: Klasifikacija zvezdnih spektrov

1 Uvod

Astronomska spektroskopija je področje, ki nam daje daleč največ opazovalnih informacij o vesolju in skoraj vseh komponentah, ki ga sestavljajo. V tej nalogi se bomo ukvarjali s spektri zvezd v vidni svetlobi, metodologija in aplikacije pa so podobne v ostalih valovnih dolžinah od gama žarkov do radijske svetlobe.

Spekter predstavimo kot graf jakosti svetlobe pri določeni valovni dolžini v odvisnosti od valovne dolžine. Prva lastnost, ki jo opazimo, je, da so spektri zvezd v grobem črna telesa s Planckovim spektrom (slika 6). Bolj vroče zvezde imajo vrh spektra pri krajših valovnih dolžinah, hladne pri daljših. Hkrati so vroče zvezde tudi svetlejše (Štefanov zakon, celotna gostota svetlobnega toka je večja). S to lastnostjo zvezd se v tej nalogi ne bomo ukvarjali. Tako bodo naši spektri v nadaljevanju podani kot normalizirani spektri (deljeni s Planckovo ovojnico). Zanimale nas bodo predvsem spektralne črte in njihove oblike, kjer leži največ informacije. Na sliki 6 so označene nekatere najmočnejše absorpcijske črte.



Slika 1: Spekter sedmih različno vročih zvezd. Označene so nekatere močnejše spektralne črte.

Iz slike 6 hitro opazimo nekaj pomembnih karakteristik spektralnih črt. Na primr: vroče zvezde imajo močne vodikove in helijeve črte, hladne zvezde pa gosto posejane črte titanovega oksida. Prav tako so črte v vročih zvezdah široke, v hladnih pa ozke. Poglejmo si nekaj fizikalnih procesov, ki vplivajo na obliko spektralnih črt, tudi takih, ki jih na sliki 6 ne moremo opaziti:

- Temperatura (navadno podana kot efektivna temperatura T_{eff}) je glavno vodilo za obliko spektralnih črt. Od temperature je odvisno kakšna je stopnja ionizacije in vzbujenosti

atomov ter molekul v fotosferi (spodnjem delu "atmosfere" zvezde, kjer absorpcijske črte nastanejo) in katere črte zato lahko nastanejo. Od temperature je odvisna tudi oblika črt, ki se zaradi termičnega gibanja atomov razširijo. Temperature med 3000 in 30 000 K so običajne.

- Zastopanost različnih elementov (podana kot kovinskost $[M/H]$, številsko razmerje kovin (v astronomiji elementov težjih od helija) napram vodiku, logaritmirano in normirano na vrednost za Sonce. Kovinskost Sonca je tako $[M/H] = 0,0$, zvezde z desetkrat manj kovinami pa imajo $[M/H] = -1,0$. Podamo lahko tudi zastopanosti posameznih elementov ali njihova razmerja, na primer [Ca/H], [Mg/Fe]). Kovinskost ali zastopanost elementov vpliva na moč posameznih spektralnih črt.

- Gravitacijski pospešek na površju zvezde (podan kot $\log g$, desetiški logaritem gravitacijskega pospeška v $\mathrm{cm\,s^{-2}}$) vpliva na tlačno razširitev črt. Gravitacijski pospeški z $0,5 < \log g < 5,0$ so običajni.

- Radialna hitrost (v_r, podana v $\mathrm{km\,s^{-1}}$) meri kako hitro se zvezda giblje proti ali stran od nas. Glede na to hitrost so valovne dolžine vseh črt premaknjene kot diktira Dopplerjev pojav. Hitrosti do nekaj 100 $\mathrm{km\,s^{-1}}$ so običajne za zvezde v naši galaksiji. Radialna hitrost ne vpliva na obliko spektralnih črt.

- Rotacijska hitrost (spet podana v $\mathrm{km\,s^{-1}}$) razširi črte, saj imajo različni deli zvezde različno hitrost zaradi projekcije rotacijske hitrosti na smer proti opazovalcu. Zvezde se lahko vrtijo z nekaj 100 $\mathrm{km\,s^{-1}}$ na ekvatorju. Hitrosti do okoli 20 $\mathrm{km\,s^{-1}}$ pa so običajne.

- Turbulenca je prisotna na površju vsake zvezde in se najbolj izraža kot konvekcijsko gibanje vročih in hladnih celic plazme. Rahlo vpliva na obliko črt, spet zaradi različnih projekcij hitrosti v konvekcijskih celicah. Tipične hitrosti so nekaj $\mathrm{km\,s^{-1}}$.

- Magnetna polja lahko ustvarijo žepe ali plasti izredno vročega plina nad površjem zvezde, ki seva emisijske črte. Šibke emisijske črte spremenijo le obliko minimuma spektralnih črt s katerimi se seštejejo. Močne emisijske črte lahko segajo daleč nad kontinuum.

- Plin okoli zvezde je lahko segret veliko bolj kot površje zvezde in seva močne emisijske črte. Plin dlje od zvezde je lahko hladen in taiste črte absorbira, kar močno zakomplicira zvezdni spekter.

- Dvojne zvezde so sistemi kjer dve zvezdi krožita ena okoli druge. Zvezd v sistemu je lahko tudi več, vendar so dvojni sistemi najbolj pogosti. Če sta zvezdi preblizu druga drugi, da bi ju lahko vizuano ločili, imamo v spektru dva seta spektralnih črt, ki pripadata dvema zvezdama z načeloma različnimi parametri.

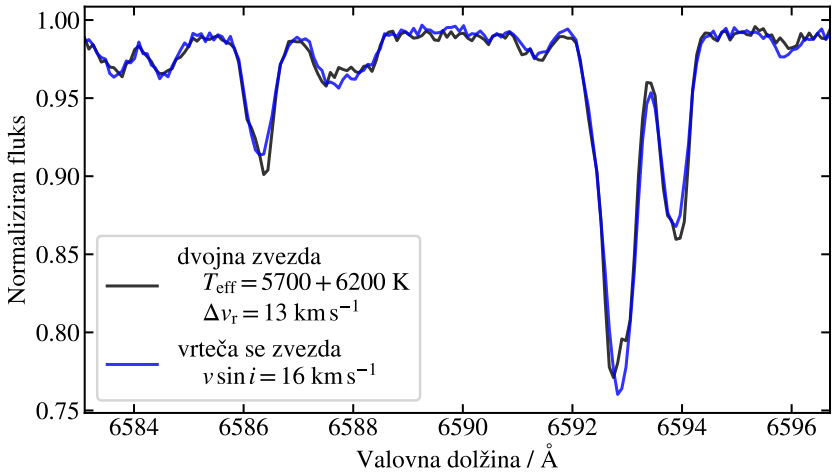
- Inštrumentalne napake in omejitve se kažejo v omejeni resolucijski moči spektrov (podano kot $R = \lambda/\Delta\lambda$, razmerje valovne dolžine in najmanjše razlike valovnih dolžin, ki jo še razločimo) in v artefaktih ter nepoissonskem šumu.

- Spektralne črte lahko nastanejo tudi v medzvezdnem mediju in v Zemljini atmosferi, kar pomeni, da nekatere črte, ki jih opazimo, niso odvisne od parametrov zvezdne atmosfere. Seveda pa lahko popačijo izmerjene oblike zvezdnih črt.

Zgornje lastnosti lahko direktno izmerimo le v izrednih primerih. Tipično moramo modelirati obliko zvezdnega spektra in vse proste parametre prilagajati dokler se modelski spekter ne ujema z opazovanim spektrom. V praksi je to izredno težka naloga, saj je 3D magnetohidrodinamika

(brez termičnega ravnovesja) računsko in konceptualno zahtevna. Navadno variiramo le nekaj osnovnih parametrov, za ostale pa uporabimo umeritvene relacije, ki jih dobimo ali iz teoretičnih študij ali iz opazovanj. Moderni pregledi neba tekom let posnamejo spektre milijonov zvezd in tudi za zgolj osnovno modeliranje takega števila spektrov potrebujemo superračunalnike. Če želimo izmeriti lasnosti zvezd hitreje, se moramo poslužiti metod strojnega učenja, ki se povsem ognejo fizikalnemu modeliranju oblike zvezdnih spektrov.

Poglejmo si nekaj izzivov, ki nas pri tem čakajo. Glavni razlog, da lastnosti zvezd ne moremo izmeriti direktno je, da so oblike spektralnih črt degenerirane – črte zelo podobne oblike lahko nastanejo na različne načine. Na sliki 2 sta narisana spektra dvojnega zvezdnega sistema in zvezde, ki se hitro vrti. Očitno je, da iz širin ali globin črt ne moremo izmeriti za kateri primer gre. Prav tako nam na videz malo pomagajo oblike črt. A če imamo črt dovolj (na sliki je le majhen del spektra s kakršnimi boste delali), je na voljo dovolj informacije, da oba primera med sabo jasno ločimo. To je primer uporabe klasifikacije spektrov, saj tipično zvezda pade le v enega od obeh razredov. S primerno klasifikacijsko metodo oba razreda ločimo brez modeliranja fizikalnih parametrov.



Slika 2: Spekter dveh povsem različnih zvezd oziroma zvezdnih sistemov. Vidimo, da ne moremo enostavno ločiti med dvojnim zvezdnim sistemom in samostojno zvezdo, ki se hitro vrti. Brez pravilne klasifikacije se ne moremo lotiti računanja zvezdnih parametrov.

Degeneracija se odraža tudi bolj subtilno. Na globino črt v nekem območju parametrov lahko vplivata zastopanost elementov v zvezdi, ali pa temperatura, ki spremeni deleže vzbujenih stanj ter ionizacije vsakega atoma. Tako lahko natančno izmerimo kombinacijo temperature in kovinskosti, ne moremo pa natančno izmeriti obeh parametrov posebej. Tega primera nam klasifikacija ne reši, saj ima zvezda lahko skoraj poljubno kombinacijo temperature in kovinskosti. Kljub temu nam metode iz te naloge lahko pomagajo najti ekstremne primere (na primer zvezde z zelo malo kovinami) ali pa ločijo vroče zvezde od hladnih, če lahko s kakšnim razlogom ignoriramo tiste vmes.

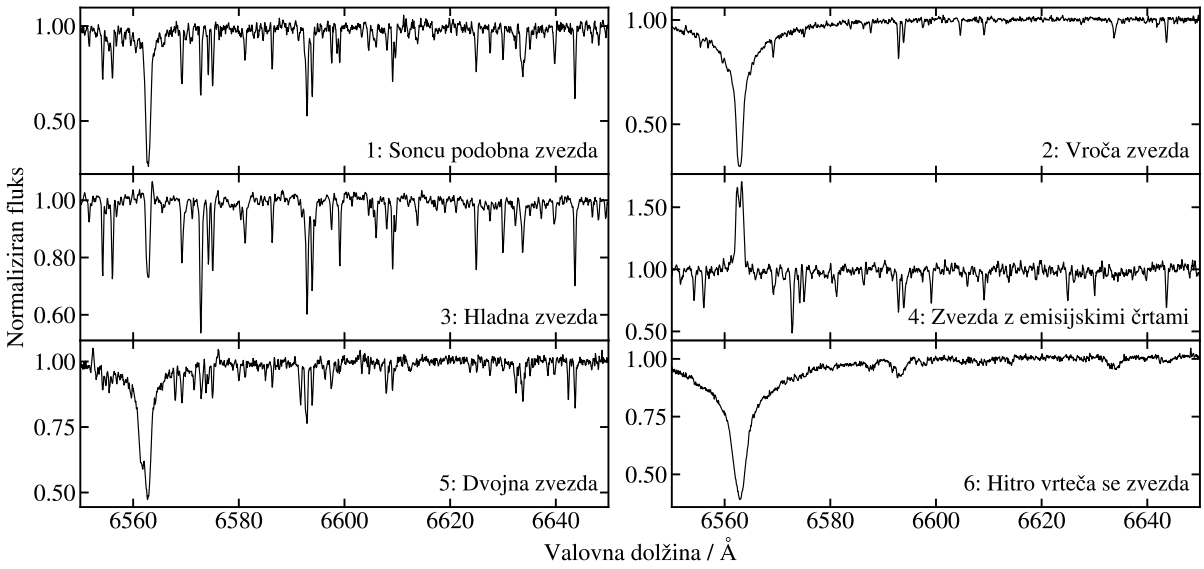
2 Podatkovni set

Podatki, ki jih imate na voljo pri tej vaji predstavljajo majhen del zbirke podatkov iz pregleda neba GALAH. GALAH je pregled neba na Anglo-avstralskem teleskopu na observatoriju Siding Spring v Avstraliji. Opazuje s teleskopom premera 3,9 m. Svetlobo zvezd v gorišču teleskopa zajame 400 optičnih vlaken, ki peljejo v spektroskop, ki lahko svetlobo iz 400-ih zvezd simul-

tano razkloni v spektre. Spektroskop opazuje v treh območjih v vidni svetlobi in enem v bližnji infrardeči. V vseh štirih kanalih skupaj pokrije okoli 1000 Å širok del spektra vsake zvezde. Valovne dolžine, ki jih opazuje, so skrbno izbrane, tako da lahko merimo zastopanosti 31 kemičnih elementov v zvezdah. Cilj projekta GALAH je prvič natančno izmeriti zastopanosti elementov za več kot milijon zvezd in preko tega rekonstruirati kemo-dinamične procese, ki so oblikovali našo galaksijo in naše osončje.

Podatkovni set za to vajo pokriva le 125 Å širok del spektrov (med 6550 Å in 6675 Å), na voljo pa imate 10 000 spektrov. Vsi spektri so normalizirani (vrednost kontinuuma je 1, spektralne črte torej predstavljajo relativno moč absorpcije pri vsaki valovni dolžini) in popravljeni za radialne hitrosti zvezd (vsi spektri so dopplersko premaknjeni tako, da so spektralne črte pri istih valovnih dolžinah kot bi bile v laboratoriju).

Vsak spekter je spravljen v svojo tekstovno datoteko. Ime datoteke je "ime" spektra. V prvi vrstici, ki je zakomentirana z "#", piše kaj je v datoteki. Potem sledi en stolpec številk, ki predstavljajo točke (fluks) spektra. Valovna dolžina vsake točke v Ångstromih je podana v datoteki `val.dat`. Na sliki 3 je narisanih nekaj spektrov.



Slika 3: Šest spektrov kaže bogatost zvezdnih spektrov, čeprav v danem podatkovnem setu najdete tudi bolj ekstremne primerke. Najbolj izrazita črta pri 6563 Å je vodikova črta Hα. Le v hladnih zvezdah so kovinske črte bolj izrazite. V spektru dvojne zvezde so vse črte podvojene. Spekter vroče zvezde kaže le malo kovinskih črt. Črte v spektru hitro vrteče se zvezde pa so razširjene v obliko črke "U".

Poleg spektrov imate na voljo manjši učni set. Ta set je v pomoč predvsem vam, da boste razumeli kaj različni spektri predstavljajo. V datoteki `ucni_set_parametri.txt` je seznam spektrov (1. stolpec) in njihovih temperatur v K (2. stolpec), gravitacijskega pospeška (3. stolpec) ter kovinskosti (4. stolpec). V datoteki `ucni_set_tipi.txt` pa je seznam spektrov (1. stolpec) in njihova klasifikacija (2. stolpec) podana s kraticami:

`MAB`: zvezde z molekulskimi absorpcijskimi črtami,

`BIN`: dvojne zvezde,

`TRI`: trojne zvezde,

`HFR`: vroče, hitro vrteče se zvezde,

`HAE`: zvezde s Hα emisijo,

`CMP`: hladne zvezde z malo kovinami,

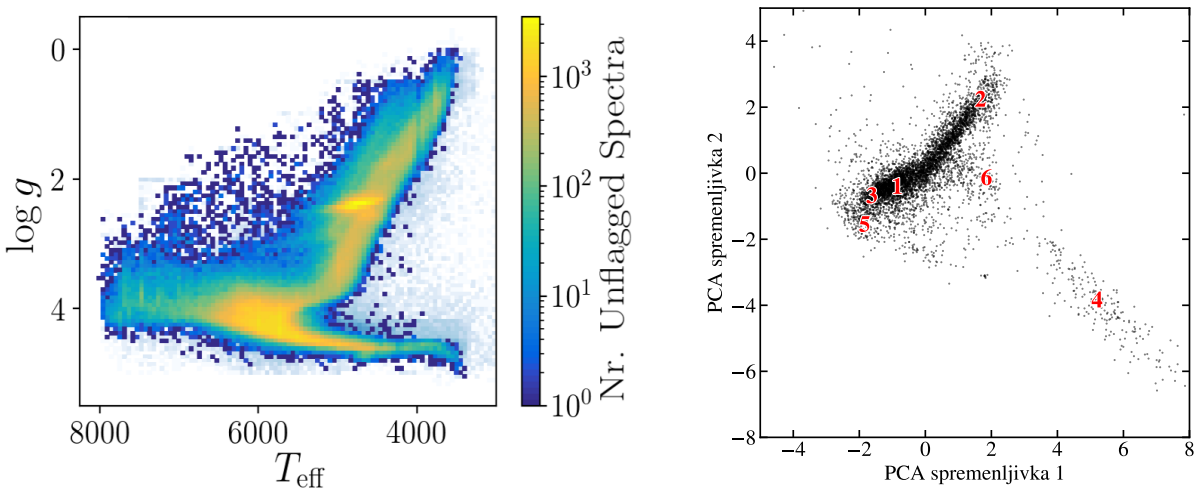
`DIB`: vroče zvezde z močnejšimi medzvezdnimi absorpcijami.

Podatke dobite na `marvin:/data/PSUF_naloge/2-naloga` ali na učilnici.

3 Zmanjševanje dimenzij kot metoda za klasifikacijo spektrov

Klasifikacije spektrov se lotimo z algoritmi, ki identificirajo zvezde s podobnimi spektri. V praksi želimo, da so si zvezde podobne po fizikalnih lastnostih. Na primer, ločiti želimo dvojne zvede od enojnih, vroče od hladnih, itd. V tej nalogi si bomo konkretneje pogledali algoritme, ki pri klasifikaciji uporabljajo zmanjševanje dimenzij.

Imejmo podatkovni set, ki je urejen v n objektov, vsak objekt pa naj bo opisan s p spremenljivkami. Radi bi zmanjšali dimenzijo našega podatkovnega seta, tako da bo n objektov čim bolj verodostojno opisanih z l spremenljivkami, kjer je $l < p$. Na primer: imamo 100 oseb, vsako osebo opišemo z nekaj lastnostmi (višina, spol, dolžina las, barva las, barva oči). 100 oseb lahko dobro opišemo z le štirimi od petih prejšnjih lastnosti, če ugotovimo, da modra barva oči navadno pomeni svetlo barvo las. Morda tako ne bomo perfektno opisali vseh 100 oseb, bomo pa pri tem uporabili manj spremenljivk. Te štiri spremenljivke niso nujno podmnožica prejšnjih petih spremenljivk. Ena od spremenljivk lahko na primer postane "povprečje" barve las in oči.



Slika 4: Levo: idealna projekcija spektrov v dve dimenziji, kjer vsaka dimenzija predstavlja en fizikalni parameter. Prikazane so vse zvezde iz pregleda neba GALAH (vir: Buder et al., 2021). Desno: graf dveh spremenljivk dobljenih s PCA. Vsaka točka predstavlja en spekter/zvezdo. S številkami so označeni spektri iz slike 3. Nekateri razredi spektrov so zbrani v ločenih skupinah. Z metodo t-SNE želimo izrazitost skupin še izboljšati. Pozor: strukture na levi in desni sliki si v ničemer niso analogne.

Spremenljivke niso nujno fizikalne lastnosti. V našem primeru bodo spremenljivke kar vrednosti fluksa za vsako točko spektra. Vsak od n spektrov je tako opisan s $p = 2084$ spremenljivkami in lahko ga predstavimo kot vektor s 2084 vrednostmi, torej kot vektor v 2084 dimenzijah. Teh 2084 dimenzij bi radi zmanjšali na recimo dve dimenziji, za kateri upamo, da bosta predstavljali neki fizikalni lastnosti, na primer temperaturo in gravitacijski pospešek na zvezdi (slika 4 levo). Za tako smiselno redukcijo dimenzij bi potrebovali izredno zapleten algoritem. V praksi

dve dimenziji redko predstavljata dve fizikalni količini, predstavljata pa lahko kombinacijo nekaj fizikalnih količin. To je pogosto dovolj, da med sabo ločimo (kasificiramo) različne zvezde, ni pa dovolj, da bi natančno izmerili same fizikalne količine (slika 4 desno). Posamezne skupine zvezd lahko nato analiziramo z drugimi metodami, kjer ne rabimo več misliti na določene fizikalne procese (vemo na primer, da naše zvezde niso dvojne, ali pa uporabimo metode, ki dobro delujejo na vročih zvezdah, ker vemo da v določeni podskupini zvezd ni nobene hladne zvezde).

Da vemo kaj različne skupine zvezd predstavljajo, moramo analizirati nekaj zvezd v vsaki skupini. Za potrebe te vaje imate pripravljena dva manjša učna seta.

V tej nalogi si bomo pogledali dve metodi za zmanjševanje dimenzij. PCA je linearna metoda, ki jo boste spisali sami, t-SNE pa je bolj napredna nelinearna metoda, ki vam je dosegljiva preko različnih knjižnic. Uporabljali boste tudi metodo za gručenje, DBSCAN, ki jo prav tako najdete v programskih knjižnicah.

3.1 PCA

Metoda glavnih komponent (Principal Component Analysis – PCA) je ena od osnovnih (linearnih) metod zmanjševanja dimenzij. S to metodo naredimo dekompozicijo našega podatkovnega seta v lastne vektorje in lastne vrednosti. Pri tem so dimenzije dobljenih lastnih vektorjev linearna kombinacija vhodnih dimenzij. Lastne vektorje lahko rangiramo od najbolj do najmanj pomembnih v podatkovnem setu, kjer najbolj pomemben (prvi) glavni vektor opisuje "smer" največje razpršenosti (variance) podatkov v večdimenzionalnem prostoru. Cilj PCA je poiskati nekaj prvih lastnih vektorjev, ki kar najbolje pojasnjujejo večji del razpršenosti podatkovnega seta, ki ga nato lahko predstavimo z linearno kombinacijo teh najbolj pomembnih lastnih vektorjev.

Imamo n spektrov, vsak od spektrov pa ima p točk (dimenzij). Tak set podatkov predstavimo z matriko $\mathbf{X}$ dimenzije $n \times p$. Najprej izračunamo odstopanja od povprečja v vsakem stolpcu. Ustvarimo vektor $\mathbf{u}$ s povprečji vseh stolpcev

$$u_j = \frac{1}{n}\sum_{i=1}^{n} X_{ij} \qquad j = 1, 2, \ldots, p.$$

Vektor $\mathbf{u}^T$ odštejemo od vsakega stolpca matrike $\mathbf{X}$, da dobimo matriko $\mathbf{B}$

$$\mathbf{B} = \mathbf{X} - \mathbf{h}\mathbf{u}^T,$$

kjer je $h_i = 1$, za $i = 1, 2, \ldots, n$ vektor enic. Temu rečemo, da smo naše originalne podatke centrirali. V naslednjem koraku izračunamo kovariančno matriko matrike $\mathbf{B}$ po definiciji

$$\mathbf{C} = \frac{1}{n-1}\mathbf{B}^*\mathbf{B}.$$

Matrika $\mathbf{B}$ je v našem primeru realna, kar pomeni, da je $\mathbf{B}^* = \mathbf{B}^{\mathrm{T}}$. Sedaj moramo izračunati lastne vektorje in lastne vrednosti kovariančne matrike. Iščemo matriko lastnih vektorjev $\mathbf{V}$ in diagonalno matriko (na diagonali so lastne vrednosti) $\mathbf{D}$, da velja

$$\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D}.$$

Pri tem lahko uporabimo poljubno metodo za izračun lastnih vektorjev in lastnih vrednosti. Rezultat je matrika $\mathbf{D}$ dimenzije $p \times p$, kjer je $D_{kl} = \lambda_k$ za vsak $k = l$, vrednosti pri $k \neq l$ pa so enake 0. λ_k je k-ta lastna vrednost. Matrika $\mathbf{V}$ je prav tako dimenzij $p \times p$, kjer je vsak stolpec lastni vektor. k-temu lastnemu vektorju odgovarja k-ta lastna vrednost. Večja kot je lastna vrednost, bolj pomembno komponento podatkov predstavlja njen lastni vektor. Sedaj

lahko uredimo stolpce matrike $\mathbf{V}$ po padajoči velikosti pripadajočih lastnih vrednosti. Hkrati na enak način uredimo matriko $\mathbf{D}$, ki naj ostane diagonalna, pri tem pa še vedno k-temu lastnemu vektorju odgovarja k-ta lastna vrednost.

Sedaj se moramo odločiti na koliko dimenzij želimo zmanjšati naš podatkovni set. Navadno izberemo toliko dimenzij, da z njimi še vedno dobro opišemo originalen set. Število potrebnih dimenzij lahko izračunamo iz lastnih vrednosti. Izračunamo lahko koliko "energije" skupaj vsebujejo vsi lastni vektorji do j-tega lastnega vektorja

$$g_j = \sum_{k=1}^{j} D_{kk} \qquad za \qquad j = 1, 2, \ldots, p,$$

kjer je g_j torej kumulativna (skupna) "energija" do j-tega lastnega vektorja. Tako lahko določimo število q, tako da prvih q lastnih vektorjev predstavlja na primer 90% vse energije. Spomnimo se, da so lastni vektorji in lastne vrednosti že rangirane. V matriko $\mathbf{W}$ spravimo q stolpcev matrike $\mathbf{V}$

$$W_{kl} = V_{kl} \qquad za \qquad k = 1, 2, \ldots, p \qquad in \qquad l = 1, 2, \ldots, q.$$

V zadnjem koraku podatkovni set $\mathbf{X}$ projiciramo na novo bazo $\mathbf{W}$, da dobimo podatkovni set z zmanjšanim številom dimenzij $\mathbf{T}$:

$$\mathbf{T} = \mathbf{X} \cdot \mathbf{W}.$$

$\mathbf{T}$ ima dimenzije $n \times q$. Vsak spekter tako opišemo, namesto s 2084 spremenljivkami, s q spremenljivkami. Pri tem se moramo zavedati, da rezultat niso spektri "interpolirani" iz 2084 točk na q točk, ampak vsak stolpec matrike $\mathbf{T}$ predstavlja nek nov parameter, ki je lahko v idealnem primeru sorazmeren s kakšno fizikalno količino. Ker so naši vhodni parametri brez enot (relativen fluks), so brez enot tudi izhodni parametri. Če bi fluks imel enote, bi morali začetni podatkovni set poleg centriranja še normalizirati (centriranju in normalizaciji skupaj rečemo standardizacija).

3.2 t-SNE

t-SNE (t-distributed Stochastic Neighbour Embedding) je močno nelinearna metoda. V glavnem se uporablja za vizualizacijo visokodimenzionalnih podatkov, mi pa jo bomo uporabili kot algoritem za klasifikacijo zvezd. V našem primeru zvezdnih spektrov je t-SNE bolj primerna izbira kot PCA, saj je zveza med fluksom ter pomembnimi fizikalnimi lastnostmi spektra lahko zelo nelinearna.

Čeprav lahko t-SNE število dimenzij zreducira poljubno, bomo v naših primerih vedno zreducirali set podatkov na le dve dimenziji, ki zadostujeta za potrebe vizualizacije in klasifikacije. Projekcija v dve dimenziji ima enostaven grafični prikaz; t-SNE vsak spekter zreducira na točko v 2D grafu. Spektri, ki so si podobni, bodo predstavljeni kot točke, ki so si blizu ena druge, zelo različni spektri pa bodo predstavljeni kot točke, ki so daleč narazen ali v različnih skupinah. Močna tendenca metode t-SNE, da podatke razdeli v skupine je lastnost, ki jo bomo v tem delu naloge izkoristili.

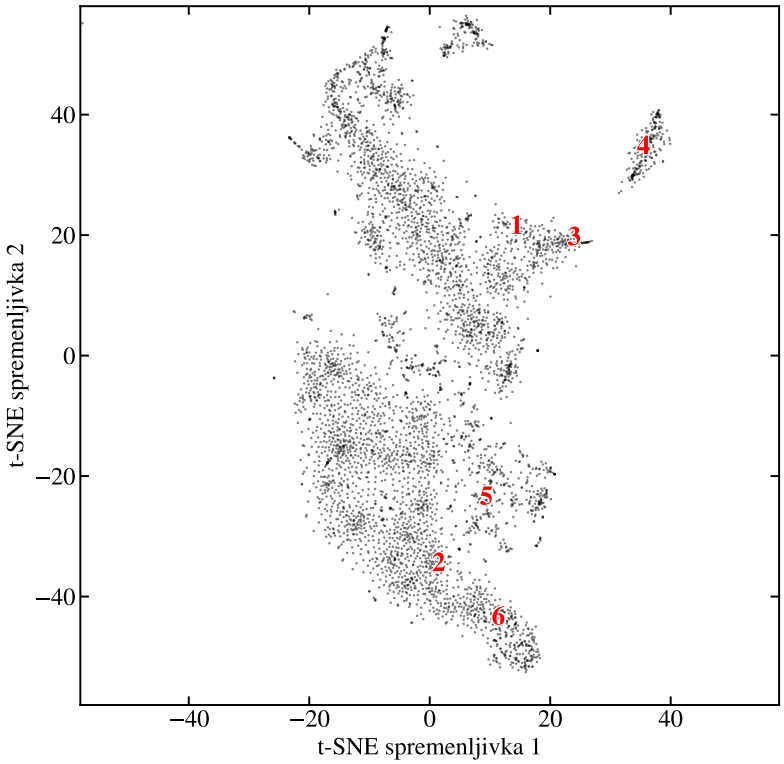
Naš podatkovni set predstavimo kot spektre $\mathbf{x}_i$, kjer je i indeks spektra, torej $i = 1, 2, \ldots, N$, če imamo N spektrov. V prvem koraku t-SNE izračuna podobnost p_{ij} med vsakim parom spektrov.

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N},$$

kjer je

$$p_{j|i} = \frac{\exp\left(-||\mathbf{x}_i - \mathbf{x}_j||^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-||\mathbf{x}_i - \mathbf{x}_k||^2 / 2\sigma_i^2\right)}.$$

Parameter σ_i se izračuna avtomatsko, tako da metoda isto dobro deluje za gosto ali redko poseljen del N-dimenzionalnega prostora. Pomembne bodo torej tudi razlike med zelo podobnimi spektri. Kako goste ali redke dele podatkov bo metoda učinkovito tretirala, narekuje parameter z imenom "perplexity", ki ga lahko v večini implementacij nastavi uporabnik.



Slika 5: Primer projekcije t-SNE z označenimi zvezdami iz slike 3. t-SNE projekcija izgleda drugače po vsakem zagonu algoritma. Kljub temu morajo biti otoki zvezd vedno podobni.

V naslednjem koraku t-SNE naredi prvo predpostavko za rezultat. Rezultat so vektorji $\mathbf{y}_i$, ki imajo dimenzijo 2. Vhodni vektorji $\mathbf{x}_i$ so imeli dimenzijo 2084. Podobno tudi za vektorje $\mathbf{y}_i$ izračunamo medsebojne podobnosti q_{ij}, tokrat po formuli

$$q_{ij} = \frac{(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2)^{-1}}{\sum_{k \neq i} (1 + ||\mathbf{y}_i - \mathbf{y}_k||^2)^{-1}}.$$

Da najdemo optimalne vrednosti za $\mathbf{y}_i$, je potrebno minimizirati sledečo mero (imenovano Kullback–Leiblerjeva divergenca)

$$KL(P, Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

Metoda t-SNE je preveč kompliciranа, da bi jo v okviru te vaje sprogramirali sami. Dosegljiva je preko knjižnic `scikit-learn`, `bhtsne`, `umap` v Pythonu ali mnogih drugih programskih jezikih. Tipično tudi ne uporabljamo originalne implementacije algoritma t-SNE, saj ima računsko zahtevnost $\mathcal{O}(n^2)$. Uporabljamo implementacije z Barnes-Hut aproksimacijo, ki imajo zahtevnost $\mathcal{O}(n \log n)$.

3.3 Vizualizacija

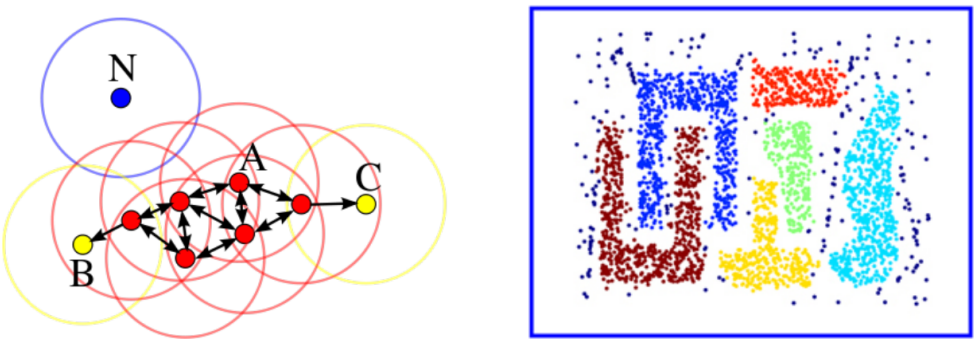
Običajne spektre predstavimo kot graf fluksa v odvisnosti od valovne dolžine. Tega ne moremo narediti z rezultati PCA ali t-SNE analize, saj po projekciji spremenljivke ne predstavljajo več fluksa pri nekih valovnih dolžinah. Spreminjanje novih spremenljivk z valovno dolžino nas niti ne zanima več, saj je bil eden od ciljev zmanjševanja dimenzij znebiti se nepotrebnih parametrov. Ker so spektri za to vajo že postavljeni v isti sistem valovnih dolžin (vse spektralne črte so vedno pri isti valovni dolžini), lahko umeritev valovnih dolžin tretiramo kot parameter, ki je identičen za vse spektre. V bistvu smo za točke spektra podali valovne dolžine le zato, da si spektre lažje predstavljate v naravnih enotah.

Zanima nas predvsem kako dobljene spremenljivke opišejo razlike med danimi spektri. Vizualizacija rezultatov je enostavna v primeru projekcije t-SNE, kjer je vsak spekter opisan z dvema spremenljivkama. Vse spektre lahko zato predstavimo v ravnini, kjer je vsaka spremenljivka ena od koordinat. Tako je narisana slika 5 – vsaka točka predstavlja en spekter. V primeru PCA, kjer je vsak spekter predstavljen z recimo nekaj deset spremenljivkami, moramo biti bolj iznajdljivi. Na sliki 4 (desno) smo izbrali dve spremenljivki, ki smo ju narisali na isti način kot v prejšnjem primeru. Narisali bi lahko tudi 3D graf, ki pa bi bil že manj pregleden. Še eno dimenzijo bi lahko narisali kot spremenljivo barvo točk, itd. Alternativa je uporaba kotnega grafa (angleško *corner plot*), kjer narišemo kombinacijo vsake spremenljivke proti vsaki drugi spremenljivki z množico 2D grafov. Pri tem nam je lahko v pomoč knjižnica `corner`, če uporabljamo Python.

Ker nas zanimajo le relacije med skupinami zvezd, ne glede na to kakšno vizualizacijo izberemo, so brezpredmetne tudi enote na oseh na slikah 5 in 4 (desno). Nič ni narobe, če enote in celotno mrežo v vizualizaciji kar izpustimo.

3.4 DBSCAN

DBSCAN je algoritem za prepoznavo gruč (clustrov) podatkovnih točk v N-dimenzionalnem prostoru. Kot vsak drug algoritem ima svoje dobre in slabe lastnosti, o čemer si lahko več preberete v literaturi, tukaj pa je pomembno to, da je sposoben zaznati gruče poljubnih oblik, kar je idealno za našo 2D t-SNE projekcijo. Poleg tega ima metoda le dva parametra ki ju poljubno nastavimo da dobimo željeno prepoznavo gruč za dan podatkovni set, in na istem setu lahko uporabimo tudi več kombinacij teh dveh parametrov da dobimo željen rezultat, npr. da DBSCAN zazna veliko manjših gruč ali pa manjše število večjih.



Slika 6: Shematičen prikaz delovanja metode DBSCAN. Za to da ta algoritem najde poljubne oblike gruč mu moramo podatki dva parametra, najmanjše število točk v poljubni zaznani gruči (minPts) in pa največjo razdaljo med osrednjimi točkami vsake gruče (ε).

DBSCAN lahko uporabimo da nam računalnik avtomatsko zazna gruče točk iz 2D t-SNE projekcije in jih izpiše v seznam, kjer potem za vsako gručo določimo klasifikacijsko skupino (kategorijo).

4 Naloga in vprašanja

1. Spoznajte se s podatkovnim setom. Najdite in narišite nekaj spektrov, ki najbolj odstopajo od povprečja (vizualno, neka metrika, ...). To so lahko učni spektri, pa tudi spektri, ki jih nekako poiščete sami.

2. Sprogramirajte metodo PCA in kodo priložite kot tekst na koncu v poročilo.

3. Vizualizirajte podatkovni set po redukciji dimenzij s PCA.

4. Opišite kje in zakaj v tej vizualizaciji ležijo spektri iz 1. točke?

5. Najmanj koliko dimenzij potrebujete, da dobro opišete dani set spektrov? Dober opis pomeni da zajamemo bistveno večino variance podatkov. V tem smislu tudi raziščite katera fizikalna količina je odgovorna za največ variance in koliko PCA komponent potrebujemo da opazimo trende vseh najbolj pomembnih fizikalnih količin (T_{eff}, $[M/H]$, $\log g$).

6. S pomočjo spektrov iz 1. točke ugotovite, ali kakšna skupina zvezd v PCA projekciji predstavlja logično zaključen razred zvezd (ki mu pripada neka unikatna fizikalna značilnost ali več le-teh). Ali v delu spektrov, ki ne paše v nobeno izolirano skupino, oziroma tvori največjo skupino, opazite kakšne trende?

7. Uporabnost PCA primerjajte z variacijo metode po imenu kernel PCA, ki jo dobite v npr. python knjižnicah.

8. Uporabite t-SNE, da naredite projekcijo podatkov v dve dimenziji. Preučite kako prosti parametri (še posebej "perplexity") te metode spremenijo projekcijo. Enako kot za PCA, s pomočjo spektrov iz 1. točke ugotovite kje se nahajajo in zakaj, ter identificirajte skupine zvezd v t-SNE projekciji, ki predstavljajo logično zaključene razrede zvezd.

9. V t-SNE projekciji poskusite najti kakšen logično zaključen razred zvezd o katerem nismo eksplicitno govorili. Takšne skupine zvezd lahko poskusite razložiti fizikalno (kaj povzroča posebnosti, ki jih opazite v spektrih) ali pa morfološko (npr. profil neke absorpcijske črte je tak in tak in bistveno odstopa od profila pri ostalih zvezdah). Tu lahko z neko avtomatsko označitvijo (glej spodaj) posameznih skupin ustvarite povprečne spektre itn.

10. Ena najbolj izstopajočih skupin so emisijske zvezde. S pomočjo t-SNE raziščite v katere skupine se razporedijo te zvezde, če pri redukciji dimenzij ne upoštevamo širšega območja črte Hα, kjer je prisotna najmočnejša emisija?

11. **DODATNO (za plus točke)**: Vseh 10 tisoč spektrov v vaši najboljši t-SNE projekciji avtomatsko klasificirajte oziroma označite s kategorijami (imeni skupin) po lastni presoji. Za avtomatsko označevanje, lahko pa tudi za pregled posameznih skupin oziroma otokov točk, uporabite metodo DBSCAN. Prikažite kakšne rešitve poda DBSCAN pri različnih vrednostih vhodnih parametrov in ali je uporaben za avtomatsko označevanje identificiranih razredov zvezd. Seznam vseh 10000 spektrov, kjer ima vsak svojo klasifikacijsko oznako (kategorijo), priložite kot datoteko k poročilu.

Literatura

1. Baron, D.: Machine Learning in Astronomy: a practical overview, arXiv (2019) (https://ui.adsabs.harvard.edu/abs/2019arXiv190407248B/abstract)
2. Guide to Spectroscopy and Spectral Lines (Astrobites) https://astrobites.org/guides/spectroscopy-and-spectral-lines/
3. Hollow, R.: Spectroscopy: Unlocking the Secrets of Star Light (https://www.sydney.edu.au/science/physics/pdfs/foundation/STW2006/hollow2.pdf)
4. PCA (Wikipedia) https://en.wikipedia.org/wiki/Principal_component_analysis
5. van der Maaten, L. & Hinton, G.: Visualizing Data using t-SNE, Journal of Machine Learning Research, 1 (2008) (https://jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf)
6. van der Maaten, L.: Barnes-Hut-SNE, arXiv (2013) (https://ui.adsabs.harvard.edu/abs/2013arXiv1301.3342V/abstract)
7. t-SNE (Wikipedia) https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding
8. t-SNE interaktivne simulacije https://distill.pub/2016/misread-tsne/
9. scikit-learn – Machine Learning in Python (vključuje Barnes-Hut t-SNE) https://scikit-learn.org/stable/
10. Barnes-Hut-SNE implementacija za Python https://github.com/dominiek/python-bhtsne
11. Implementacije t-SNE v drugih jezikih https://lvdmaaten.github.io/tsne/
12. DBSCAN (Wikipedia) https://en.wikipedia.org/wiki/DBSCAN
13. Ester, M. et al.: DBSCAN – A density-based algorithm for discovering clusters in large spatial databases with noise (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.71.1980)

5 Dodatek: prekletstvo dimenzij

Mogoče se vam je pri tej nalogi postavilo vprašanje zakaj kompliciratи, če lahko podobne spektre najdemo na bolj preprost način, na primer s χ^2 testom? Problem na katerega bi naleteli se imenuje prekletstvo dimenzij (angleško curse of dimensionality). Visokodimenzionalni prostor je izredno prazen. To pomeni, da so vse točke daleč stran druga od druge. Posledično se je težko odločiti kaj sploh pomeni, da sta si dve točki blizu. Naše spektre smo predstavili kot točke v 2084 dimenzionalnem prostoru. Tudi po projekciji imamo pogosto opravka z več 10 dimenzionalnimi parametričnimi prostori.

Da je velikodimenzionalen prostor zelo prazen, si zlahka predstavljamo. Za začetek recimo, da imamo enodimenzionalen prostor (omejen med 0 in 1), ki ga napolnimo s točkami. Recimo, da če vanj enakomerno razporedimo 10 točk, je prostor gosto poseljen s točkami. Da bi enako gosto poselili dvodimenzionalen prostor, bi rabili $100 = 10^2$ točk. Za tridimenzionalni prostor bi jih rabili $1000 = 10^3$, za 30 dimenzionalni bi jih rabili 10^{30}. Nihče na tem svetu nima podatkovnega seta velikega 10^{30} točk!

6 Dodatek: odpiranje spektrov s Pythonom

En spekter lahko odprete in narišete na sledeč način:

```
import numpy as np
import matplotlib.pyplot as plt

# load wavelengths file
wav = np.loadtxt('spektri/val.dat', comments='#')

# load one spectrum (spectrum number 123, for example)
flux = np.loadtxt('spektri/123.dat', comments='#')

# plot spectrum
plt.plot(wav, flux, 'k-')
plt.xlabel(r'Wavelength / $\mathrm{\AA}$')
plt.ylabel(r'Normalized flux')
plt.show()
```

Spektre lahko spravite v numpy array (matriko). Matrika je potem shranjena v spominu, tako da ni več potrebno odpirati posameznih spektrov:

```
import numpy as np
import matplotlib.pyplot as plt

# load wavelengths file
wav = np.loadtxt('spektri/val.dat', comments='#')

# load spectra in a loop
spectra_array=[]
for spectrum in range(1,5750):
    flux = np.loadtxt('spektri/%s.dat' % spectrum, comments='#')
    spectra_array.append(flux)

# create an array
spectra_array = np.array(spectra_array)

# plot spectrum
plt.plot(wav, spectra_array[1234], 'k-')
plt.xlabel(r'Wavelength / $\mathrm{\AA}$')
plt.ylabel(r'Normalized flux')
plt.show()
```

7 Dodatek: nekaj napotkov za pisanje poročila

- Ime datoteke poročila naj vsebuje ime, priimek ter datum
- Uvod naj ima pol do ene strani, to je povzetek navodila/vsebine naloge in ne kopiranje teksta
- V poročilu ne pisati podrobne izpeljave PCA ali t-SNE

- Vsak svoj postopek podrobno opišite
- V besedilu pazite na slog svojih trditev oziroma ugotovitev (bodite previdni/ponižni/nežni)
- Pazite na slovenski jezik!
- Pazite na velikost označb na grafih oziroma vseh elementov grafik ter grafik samih
- Pri risanju gostih območij točk odstranite rob točk ter uporabljajte transparenco (ali konture ali 2D histogram)
- Pazite na pravilno označevanje in sklicevanje na grafe/slike/itn.
- Vsa grafika naj bo taka da izgleda v redu ko poročilo natisneš in ne preveilka v smislu končne velikosti datoteke poročila (pazite tudi na vektorski/rasterski zapis slik)
- Pazite na pravilen vrstni red stvari: seznam spektrov v datotekah ter npr. PCA komponent
- V seznamu DBSCAN klasifikacije ne uporabljajte številk gruč ali kaj podobnega ampak imena kategorij
- Napišite če odkrijete da v učnem setu kakšen spekter ni pravilno klasificiran