SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Rajan Zejnuni

SEMANTIČKO INDEKSIRANJE DOKUMENATA I ITERATIVNO PRETRAŽIVANJE

Diplomski rad

Voditelj rada:
doc. dr. sc. Pavle Goldstein

Zagreb, veljača 2019

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Zahvaljujem se obitelji i prijateljima koji su mi dugogodišnja podrška u svakom koraku koji sam napravio kako bih došao do ovog cilja. Zahvaljujem se mentoru Pavlu Goldsteinu na neograničenom strpljenju i profesionalnosti, kolegama, profesorima te Matematičkom odsjeku Prirodoslovno-matematičkog fakulteta.

Diplomski rad posvećujem svojoj obitelji, prijateljima i svima koji nažalost nisu imali priliku studirati. Najveća motivacija tijekom studiranja mi je bio prijatelj koji nas je prerano napustio. Bio mi je motivacija i potpora s neba jer smo imali slične ciljeve i planove za život. Ovaj rad posvećujem tebi, Freza. ***12/10***

Sadržaj

Uvod

Donošenje odluka je proces koji je prisutan tijekom cijelog života svakog čovjeka. Odluke donosimo svakodnevno, ali ipak, neke odluke imaju veću važnost te ponekad zahtijevaju detaljniju analizu.
Koliko detaljnu? Ovisi o važnosti posljedica naših odluka. Takve odluke, naprimjer, donosimo kada biramo obrazovanje, posao ili osobno prijevozno sredstvo. Prije donošenja konačne odluke često stvaramo popis s prednostima i nedostacima za sve moguće izbore te onaj izbor koji ima najviše pozitivnih ocjena ili pluseva je naš konačan odabir. Najčešće nam takva metoda odlučivanja pomaže samo savjetodavno te u konačnici donosimo odluku prema prethodnim osobnim iskustvima, prema nekoj tradiciji ili jednostavno, po osjećaju. Međutim, ako problem promatramo s istraživačke strane onda je rezultat takvog odlučivanja egzaktan. Očito je da se radi o jednostavnoj klasifikaciji tijekom odabira automobila ako imamo prikazane sve prednosti i nedostatke nekoliko modela. Konačna odluka neće značajno utjecati na naš život ali nam je i ta odluka bitna. No, što ako radimo u medicinskom laboratoriju te želimo znati je li tumor malignan ili benignan? Što ako želimo odrediti karakteristike proteinskih familija i klasificirati proteine?
Odgovore na ova i mnoga druga pitanja nam daju klasifikacijske i regresijske statističke metode, strojno učenje *(eng. Machine learning)* te općenito rudarenje podacima *(eng. Data Mining)*. U radu koji slijedi upoznat ćemo vas s klasifikacijskim problemom na skupu podataka proteinskih familija. Razlog odabira klasifikacije proteina u proteinske familije je nepostojanje rječnika prema kojem možemo jednostavno odlučiti o kojem se proteinu radi te sam zbog toga isti morao osmisliti. Nakon osmišljavanja rječnika proteina upoznat ćemo se s klasifikacijskim algoritmom potpornih vektora (*eng. Support Vector Machine*) kojeg ćemo koristiti u nastavku rada. Zbog utjecaja broja elemenata rječnika, dimenzija modela će postati prevelika te ćemo predstaviti algoritam glavnih komponenti (*eng. Principal Component Analysis*) kojim ću reducirati dimenziju modela. Na kraju rada ću predstaviti rezultate te konačan zaključak.
Proteinske familije možemo klasificirati pomoću karakterističnih motiva a njihovo traženje je iterativno pretraživanje. Problem karakterističnih motiva ću zamjeniti proteinskim rječnicima n-grama te umjesto iterativnog pretraživanja karakterističnih motiva, provest ću klasificiju proteina u proteinske familije pomoću proteinskih rječnika.

Poglavlje 1

Pojmovi i definicije

1.1 Vektorski prostor

[3] Neka je V neprazan skup na kojem su zadane binarna operacija zbrajanja $+ : V \times V \to V$ i operacija množenja skalarima iz polja $\mathbb{F}$, $\cdot : \mathbb{F} \times V \to V$. Kažemo da je uređena trojka $(V, +, \cdot)$ vektorski prostor nad poljem $\mathbb{F}$ ako vrijedi:

(a) $a + (b + c) = (a + b) + c, \forall a, b, c \in V$

(b) postoji $0 \in V$ sa svojstvom $a + 0 = 0 + a = a, \forall a \in V$

(c) za svaki $a \in V$ postoji $-a \in \mathbb{V}$ tako da je $a + (-a) = -a + a = 0$

(d) $a + b = b + a, \forall a, b, \in V$

(e) $\alpha(\beta a) = (\alpha\beta)a, \forall \alpha, \beta \in \mathbb{F}, \forall a \in V$

(f) $\alpha(a + b) = \alpha a + \alpha b, \forall \alpha \in \mathbb{F}, \forall a, b \in V$

(g) $1 \cdot a = a, \forall a \in V$

1.2 Euklidska udaljenost dviju točaka

Neka su $A, B \in V^n, A = (a_1, a_2, \ldots, a_n), B = (b_1, b_2, \ldots, b_n)$ točke dane svojim pravokutnim koordinatama. Tada je euklidska udaljenost od A do B dana sa

$$d(A, B) = \sqrt{\sum_{i=1}^{n} (b_i - a_i)^2}$$

1.3 Ravnina

[1] Neka je T_0 točka n-dimenzionalnog afinog prostora A^n, V^n njemu pridruženi vektorski prostor i W^k k-dimenzionalni potprostor od V^n. Skup svih točaka $T \in A^n$ za koje je $\overrightarrow{T_0T} \in W^k$ naziva se k-dimenzionalna ravnina (k-ravnina) točkom T_0 sa smjerom W^k i označava sa Π^k, tj. ravnina Π^k točkom T_0 sa smjerom W^k je skup

$$\left\{T \in A^n \mid \overrightarrow{T_0T} \in W^k\right\}$$

1.4 Linearni operatori

[5]Neka su V i W vektorski prostori nad poljem $\mathbb{F}$. Preslikavanje $A : V \to W$ se zove linearan operator ako vrijedi

$$A(\alpha x + \beta y) = \alpha Ax + \beta Ay, \forall x, y \in V, \forall \alpha, \beta \in \mathbb{F}.$$

1.5 Svojstvene vrijednosti i svojstveni vektori

[4] Neka je $A \in L(V)$. Ako je $Av = \lambda v$ za neki $v \neq 0$, onda kažemo da je $\lambda \in K$ svojstvena vrijednost operatora A i da je v svojstveni vektor operatora A (za svojstvenu vrijednost λ). Skup svih svojstvenih vrijednosti operatora A zovemo spektrom operatora A i označavamo sa

$$\sigma_A = \{\lambda \in K : \lambda \ je \ svojstvena \ vrijednost \ od \ A\}.$$

Pri čemu je K polje realnih brojeva $\mathbb{R}$ ili polje kompleksnih brojeva $\mathbb{C}$.

1.6 Varijanca uzorka

Varijanca uzorka je prosječno kvadratno odstupanje od aritmetičke sredine uzorka te ju označavamo sa s^2 i računamo

$$s^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})$$

Pri tome su x_i vrijednosti uzorka za $i = 1, 2, ..., n$, a $\overline{x}$ aritmetička sredina uzorka.

Poglavlje 2

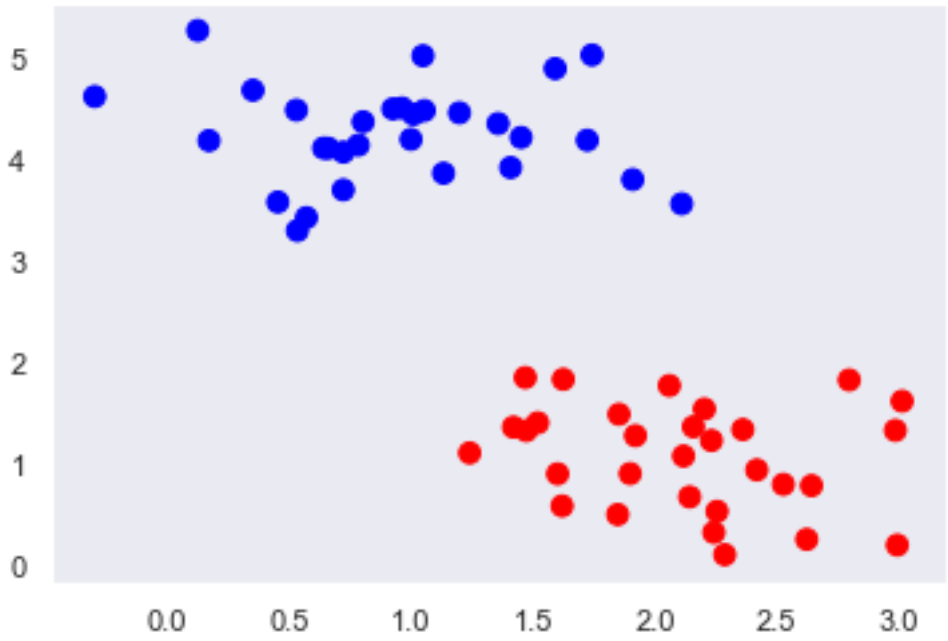
Algoritam potpornih vektora

Algoritam potpornih vektora (*eng. Support Vector Machine*) je klasifikacijski algoritam koji se najčešće koristi u modelima velike dimenzionalnosti, što i je naš slučaj. Algoritam se temelji na intuitivnim matematičkim pojmovima kao što su euklidska udaljenost, odnos dvaju (i više) vektora te pojam hiperravnine. Težini algoritma pridonose visokodimenzionalni prostori u kojima računamo udaljenosti vektora od hiperravnina, određujemo hiperravnine ili računamo projekcije.

Podaci korišteni u sljedećem primjeru su generirani "slučajno" sa zadanim krajnjim vrijednostima. Dakle, podaci nisu preuzeti niti s jednog javnog izvora, služe kako bismo jednostavno definirali klasifikaciju podataka i optimalnu razdvajajuću hiperravninu te ih nećemo koristiti u nastavku rada.

Da bismo najjednostavnije objasnili princip SVM-a koristit ćemo se jednostavnim primjerom u dvodimenzionalnom vektorskom prostoru.

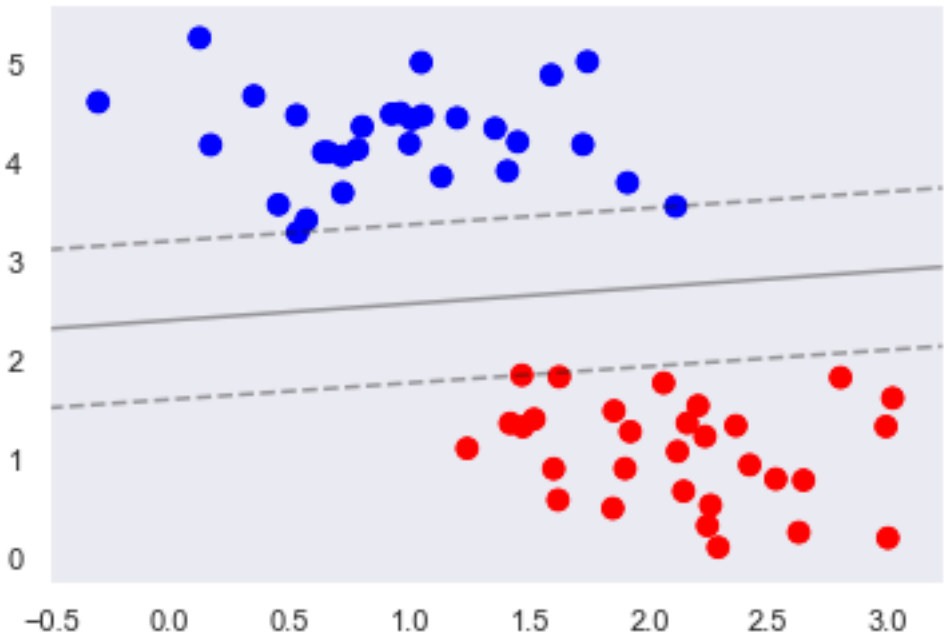
Primjer 1:
Neka su A i B skupovi točaka u dvodimenzionalnom vektorskom prostoru (*Slika 2.1*). Točke skupa A su označene plavom a točke skupa B crvenom bojom. Kao što smo ranije naveli, cilj je pronaći razdvajajuću hiperravninu. Tražena hiperravnina nije jedinstvena te nas zanima optimalna razdvajajuća hiperravnina.



Slika 2.1: Skupovi podataka A i B

Nakon definiranja razdvajajuće hiperravnine možemo definirati potporne vektore.

Potporni vektori su točke koje su najbliže našoj razdvajajućoj hiperravnini te one imaju najveći utjecaj na položaj i orijentaciju razdvajajuće hiperravnine. Ako odaberemo jednu plavu točku skupa A koja leži na isprekidanom pravcu (*Slika 2.2*) te ju pokušamo zamisliti bliže setu B u smjeru okomitom na razdvajajuću hiperravninu, možemo zaključiti da će se smanjiti margina. Isto tako, translacijom točaka u različitim smjerovima možemo promijeniti smjer razdvajajuće hiperravnine.



Slika 2.2: Optimalna razdvajajuća hiperravnina

Nakon određivanja optimalne razdvajajuće hiperravnine, točke validacijskog seta podataka se klasificiraju u dva skupa s obzirom na razdvajajuću hiperravninu. Dakle, sve točke "ispod" hiperravnine će pripadati skupu B, a sve točke "iznad" hiperavnine će pripadati skupu A. Svaka točka validacijskog seta se može prikazat u vektorskom prostoru te pozicija točke u odnosu na razdvajajuću hiperravninu determinira kojem skupu podataka pripada.
Primjenu i detaljniju raščlambu klasifikacije ću opisati u poglavlju **Primjena: Algoritam potpornih vektora** na skupu podataka proteinskih familija.

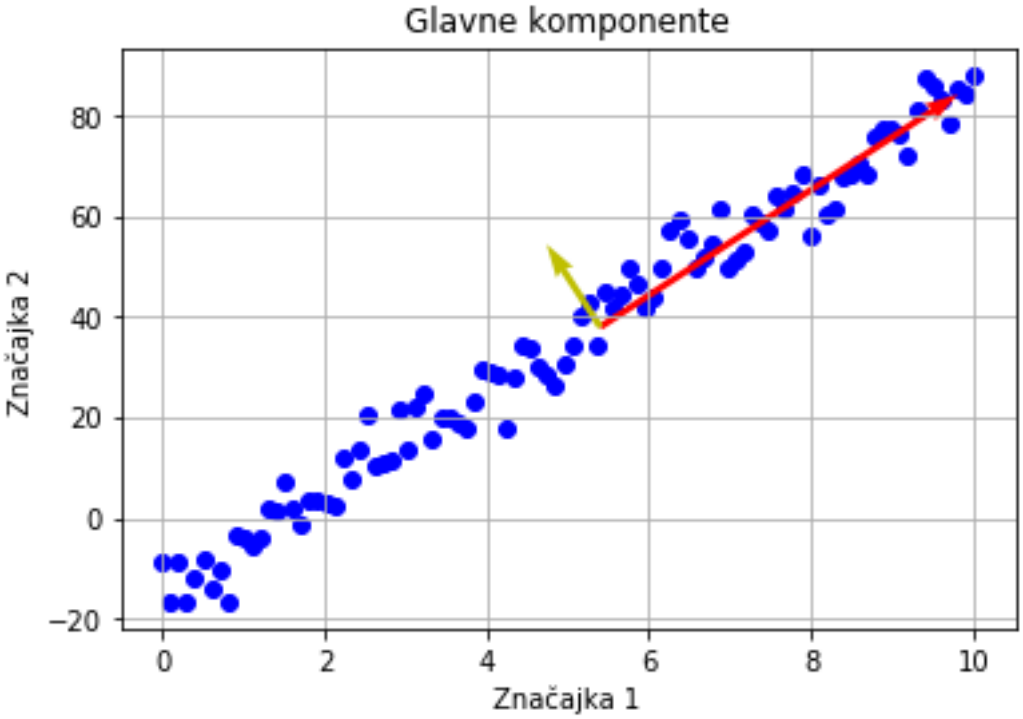
Poglavlje 3

Analiza glavnih komponenti

Neka su $Y = [y_1, y_2, \ldots, y_m]$ zavisna kategorijska varijabla, $X = [x_{ij}],\ i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n$ matrica prediktora čiji su stupci $X_i,\ i = 1, 2, ..., n$ značajke.
Analiza glavnih komponenti *(eng. Principal Component Analysis)* je statistička metoda koja se temelji na pojmovima linearne algebre kao što su vektori, matrice, svojstvene vrijednosti te operatori i operacije na operatorima. Početni skup prediktora $X_1, X_2, \ldots X_n$ se transformira u novi, nekorelirani skup prediktora. Dobiveni prediktori su nekorelirani, te sortirani tako da prvi novi prediktor opisuje najviše varijabilnosti podataka u odnosu na Y a zadnji prediktor najmanje. Odnos trasformiranih prediktora i glavnih komponenti detaljno ćemo objasniti u poglavlju **Primjena: Analiza glavnih komponenti**.

Glavne komponente su (jedinični) ortogonalni vektori koji razapinju n-dimenzionalni prostor u kojem su naši podaci smješteni. Na pravcu kojeg određuje prvi vektor tj. prva glavna komponenta je disperzirano najviše podataka, svaki sljedeći vektor zadovoljava kriterije ortogonalnosti (i normiranosti) te se disperzija podataka smanjuje sa svakom sljedećom glavnom komponentom. Nekoreliranost glavnih komponenti rezultirana je ortogonalnošću razapinjajućih vektora te možemo reći da su nekoreliranost i ortogonalnost u ovom slučaju u $1:1$ odnosu.
Kao što sam već naveo, prva glavna komponenta opisuje najviše varijabilnosti te je najviše točaka disperzirano na pravcu kojeg određuje prva glavna komponenta. Radi lakšeg razumijevanja, možemo parametrizirati naš problem te reći da se podaci nalaze u elipsoidu. Osi elipsoida su vektori glavnih komponenti te se podaci, u tom kontekstu, raspršuju po osima elipse u prostoru V^2, elipsoida u prostoru V^3, odnosno hiperelipsoida u prostorima V^n gdje je $n > 3$.
Pojmove u zagradama *jedinični* i *normirani* sam naveo opcionalno jer matrica prediktora može i ne mora biti skalirana/normirana. U nastavku slijedi detaljnije objašnjenje.
Dana "parametrizacija" pojmova nam pomaže da razumijemo opisanu varijabilnost poje-

dine glavne komponente. Duljina raspršenosti podataka po pravcima kojeg određuju osi eliposida i opisana varijabilnost podataka su u odnosu 1 : 1.
Dakle, što je raspršenost podataka „duža“ po pravcu kojeg određuje glavna komponenta, opisana varijabilnost podataka modela je veća.
U grafičkom prikazu ispod su prikazane glavne komponente. Ako promotrimo crveni vektor (prvu glavnu komponentu) zaključujemo kako su podaci disperzirani na pravcu kojeg određuje vektor označen crvenom bojom.



Slika 3.1: Glavne komponente

U analizi koja slijedi će nas zanimati pravci na kojima je naviše podataka disperzirano. Navedenom definicijom smo kreirali grafički kriterij na osnovu kojega odbacujemo ili zadržavamo glavne komponente. Navedenim slijedom možemo odbaciti sve glavne komponente čiji podaci na pripadnim osima nisu dovoljno „dugi“. Zadovoljavajuća duljina podataka se određuje empirijski, najćešće uvjetovana visinom opisane kumulativne varijabilnosti. Npr. želimo da naše glavne komponente opisuju 80% varijabilnosti modela.

Odbacivanjem glavnih komponenti pozivajući se na neki od kriterija koje ćemo predstaviti i primijeniti u ovom radu, reduciramo dimenziju prostora što će nam biti od iznimne

važnosti u nastavku. Koliko je potrebno smanjiti dimenziju modela određeno je našim pretpostavkama i predefiniranim očekivanjima. Stoga, ako želimo zavisnu varijablu Y što bolje opisati, zadržat ćemo što više glavnih komponenti. Međutim, ako nam je primarni cilj smanjiti dimenzionalnost onda nam je ponekad i jedna glavna komponenta dovoljna za model. Najćešće se želi postići "zlatna sredina" te nam je bitna i opisana varijabilnost i redukcija dimenzije.
Model koji ćemo kreirati, zbog prirode podataka u vidu visoke dimenzionalnosti te zbog prirode autora u vidu težnje za boljim rezultatima, zahtijeva redukciju dimenzije uz uvjet da se opisana varijabilnost značajno ne smanji. O redukciji dimenzije i opisanoj varijabilnosti detaljnije ćemo govoriti u sljedećim poglavljima.
U sljedećih nekoliko koraka ćemo opisati postupak računanja glavnih komponenti te njihovu redukciju.

3.1 Matrica prediktora

Promotrimo li stupce dolje prikazane matrice, svaki stupac X_i predstavlja jednu nezavisnu varijablu ili značajku (ne znači nužno da varijable nisu korelirane) pomoću kojih želimo opisati našu zavisnu varijablu Y.

$$X = \begin{bmatrix} x_{11} & \mathrm{x}_{12} & \dots & \mathrm{x}_{1n} \\ \mathrm{x}_{21} & \mathrm{x}_{22} & \dots & \mathrm{x}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{x}_{m1} & \mathrm{x}_{m2} & \dots & \mathrm{x}_{mn} \end{bmatrix}$$

3.2 Normalizacija podataka

U većini slučajeva strojnog učenja varijable $X_i,\ i = 1, 2, ..., n; n \in \mathbb{N}$, kojima opisujemo zavisnu varijablu Y, imaju jako veliku varijabilnost te moramo podatke normalizirati ili skalirati. Dakle, mora vrijediti $\overline{X_i} = 0, s^2 = 1$ ili $x^*_{ij} = \frac{x_{ij}}{\overline{X_i}}, i = 1, 2, ..., n; j = 1, 2, ..., m$ respektivno.

Pretpostavljamo da je dovoljno skalirati stupce matrice X te računamo prosječne vrijednosti $\overline{X_i}$ svakog stupca matrice X:

$$\overline{X_i} = \frac{1}{m} \sum_{j=1}^{m} x_{ji} \; i = 1, 2, ..., n$$

$$R = \begin{bmatrix} \frac{x_{11}}{\overline{X_1}} & \frac{x_{12}}{\overline{X_2}} & \cdots & \frac{x_{1n}}{\overline{X_n}} \\ \frac{x_{21}}{\overline{X_1}} & \frac{x_{22}}{\overline{X_2}} & \cdots & \frac{x_{2n}}{\overline{X_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{x_{m1}}{\overline{X_1}} & \frac{x_{m2}}{\overline{X_2}} & \cdots & \frac{x_{mn}}{\overline{X_n}} \end{bmatrix}$$

3.3 Svojstvene vrijednosti matrice L

Možemo primijetiti kako matrica R nije uvijek kvadratna matrica, odnosno $m \neq n$ vrijedi u većini slučajeva. Jako je rijedak slučaj da imamo jednak broj opservacija i značajki. Međutim, jednostavnom transformacijom možemo doći do kvadratne matrice.

Primjer 2:
Množenjem matrice A_{mn} i pripadajuće transponirane matrice A^T_{nm} dobivamo kvadratnu matricu A_{mm}.

Determinanta je funkcija definirana na skupu svih kvadratnih matrica i poprima vrijednosti na skupu skalara. Shodno tome, možemo izračunati determinantu svake kvadratne matrice pa tako i matrice $L := RR^T$. Nas ipak zanimaju svojstvene vrijednosti i svojstveni vektori matrice L kako bismo došli do glavnog cilja ovog poglavlja, glavnih komponenti. Računamo svojstvene vrijednosti i svojstvene vektore matrice $L = R^T R$.

$$Lv = \lambda v, \text{ za neki } v \neq 0$$

- Svojstvene vrijednosti: $\lambda_1, \lambda_2 \ldots, \lambda_m$

- Svojstveni vektori: $v_1, v_2 \ldots, v_m$

Dijagonalna matrica sa svojstvenim vrijednostima $\lambda_1, \lambda_2, ..., \lambda_m$ na dijagonali.

$$D = \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_m \end{bmatrix}$$

Pozivajući se na kolegij Linearna algebra, možemo izračunati dekompoziciju matrice L te ju prikazati na sljedeći način:
$L = VDV^T$ pri čemu je V matrica čiji su stupci svojstveni vektori a D dijagonalna matrica.

3.4 Glavne komponente

Konačno, računamo matricu $G := VR$ pri čemu je V matrica svojstvenih vrijednosti a R normalizirana matrica X.
Ortogonalna baza prostora u kojem se nalaze naši podaci je usko povezana s vektor-stupcima matrice R. U primjeni ćemo detaljno objasniti povezanost vektor-stupaca matrice R, navedene ortogonalne baze i glavnih komponenti.

Poglavlje 4

Kriteriji zadržavanja glavnih komponenti

4.1 Metoda disperzije podataka

Neka su $\lambda_1, \lambda_2, ..., \lambda_m$ svojstvene vrijednosti matrice L te pretpostavimo da naš model opisuje udio varijabilnosti podataka c, gdje je $c \in [0, 1]$. Kriterij *Metoda disperzije podataka* je najpouzdaniji kriterij odbacivanja, odnosno zadržavanja glavnih komponenti te ga provodimo na sljedeći način.

Za svaki $C_l,\ l = 1, 2, ..., m$ provjeravamo istinitost nejednakosti $C_l > c$.

Pri čemu je $C_l := \frac{p_j}{f}$

$$p_j = \sum_{i=1}^{l} \lambda_i,\ j = 1, 2, ..., m; l = 1, 2, ..., m$$

$$f = \sum_{i=1}^{m} \lambda_i$$

4.2 Kaiser-Guttman kriterij

Kaiser-Guttman kriterij je jednostavan kriterij zadržavanja glavnih komponenti prema kojem trebamo zadržati sve glavne komponente čije su pripadne svojstvene vrijednosti $\lambda_i >= \overline{\lambda},\ i = 1, 2, ..., m$, pri čemu je

$$\overline{\lambda} = \frac{1}{n} \sum_{i=1}^{n} \lambda_i$$

4.3 Metoda lakta

Metoda lakta *(eng. Broken Stick)* je grafička metoda koja se koristi kao kriterij zadržavanja, odnosno odbacivanja glavnih komponenti. Često je nepouzdana te ćemo je u ovom radu koristiti savjetodavno a ne kao strogi kriterij. Uz Kaiser-Guttmanov kriterij, metoda lakta je najčešće spominjana metoda u literaturi no nerijetko u kontekstu nepouzdane metode.

Poglavlje 5

Primjena: Algoritam potpornih vektora

5.1 Baza podataka

Podaci na kojima ćemo primijeniti sljedeće metode i algoritme su biomedicinske prirode te kao takvi mogu biti jako zahtjevni za manipulaciju. Kao što sam ranije naveo, radi se o bazi podataka 9 proteinskih familija preuzetih s javne baze podataka PFAM koju možete pronaći putem poveznice: *https://pfam.xfam.org/*

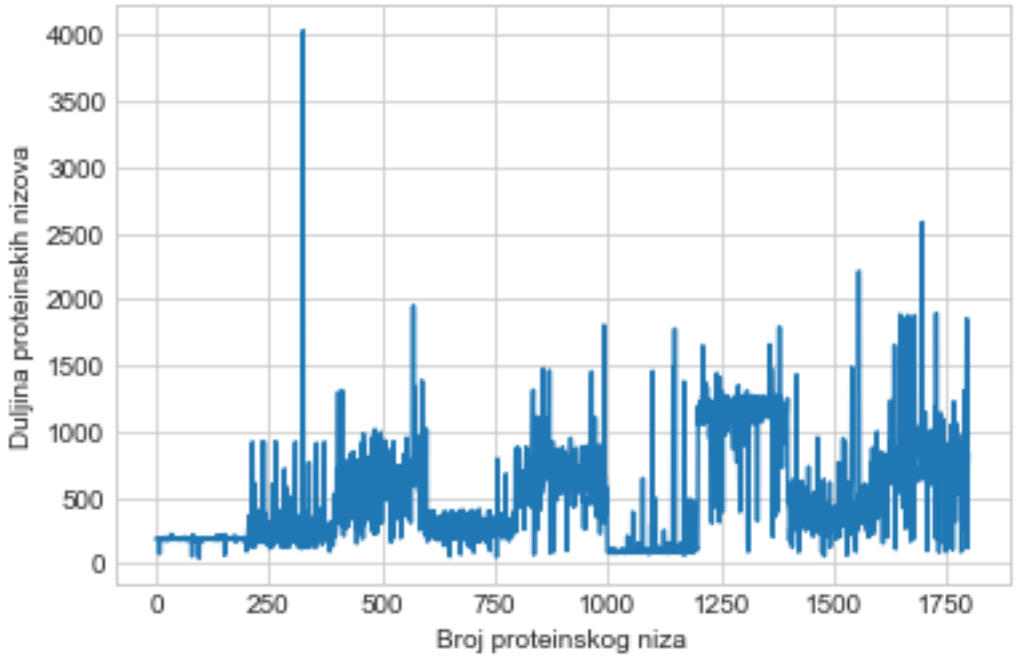
5.2 Proteinske familije

Svaka od proteinskih familija se sastoji od 200 proteinskih nizova a svaki proteinski niz se sastoji od kombinacije 20 poznatih aminokiselina i aminokiseline **X**, kada ona nije poznata.

Oznaka	**Aminokiselina**	**Oznaka**	**Aminikiselina**
G	Glycine	**P**	Proline
A	Alanine	**V**	Valine
L	Leucine	**I**	Isoleucine
M	Methionine	**C**	Cysteine
F	Phenylalanine	**Y**	Tyrosine
W	Tryptophan	**H**	Histidine
K	Lysine	**R**	Arginine
Q	Glutamine	**R**	Asparagine
E	Glutamic Acid	**D**	Aspartic Acid
S	Serine	**T**	Threonine

Tablica 5.1: Popis aminokiselina

U tablici u prethodnom prikazu je prikazano 20 najpoznatijih (najčešćih) aminokiselina i oznaka aminokiselina.
Broj aminokiselina u svakom od 1800 proteina nije fiksiran pa tako varira od nekoliko desetina do nekoliko tisuća aminokiselina. Grafički prikaz ispod prikazuje varijabilnost duljine pojedinih proteina.



Slika 5.1: Duljina proteinskih nizova

Familija proteina započinje s proteinom koji se sastoji od 189 aminokiselina, kulminira s proteinom od 4035 aminokiselina te završava s proteinom koji se sastoji od 833 aminokiselina. U sljedećem prikazu možemo vidjeti 50 prvih aminokiselina prvog i posljednjeg proteina. Prikaz je ograničen zbog duljine proteinskih nizova te za detaljniji uvid u podatke pogledajte priložene podatke u elektroničkom obliku.

→ *MNQVLKDALEDNPIIVAIKDDAGLQKCKESESRIIFILYGDLLNIADIVD...*
→ *MEAMEGWVALLLLMYHTQQWQTVATRGQDSIKSHIFYAVEMEGGSPAARA...*

Training set se sastoji od 200 primjera proteina svake od devet proteinskih familija, ukupno 1800 primjera. Prvih 200 primjera pripadaju familiji proteina označenih brojem 1, sljedećih 200 primjera pripadaju familiji proteina označenih brojem 2 itd. Točne nazive prethodno numeriziranih proteinskih familija možemo vidjeti u sljedećoj tablici.

Br.	Naziv proteinske familije
1 ←	Glycerol-3-phosphate responsive antiterminator
2 ←	Chromatin modification-related protein EAF7
3 ←	c-SKI Smad4 binding domain
4 ←	Plasmid replication region DNA-binding N-term
5 ←	Cell-cycle sustaining, positive selection
6 ←	Dynactin
7 ←	Up-frameshift suppressor 2
8 ←	UcrQ family
9 ←	Peptidase S8 pro-domain

Tablica 5.2: Proteinske familije

Validacijski set se sastoji od 100 proteina svake familije, odnosno 900 proteina ukupno.

Sada kada su trening set i validacijski set predstavljeni potrebno je predstaviti proteinske rječnike na kojima će se temeljiti klasifikacija i redukcija dimenzije modela.

5.3 Rječnik proteinskih familija

U prvom stupcu u sljedeće dvije tablice se nalaze elementi rječnika trojki, odnosno rječnika petorki. U preostalim stupcima se nalaze frekvencije elemenata trojki, odnosno petorki u svakoj od 9 familija proteina.
Da bismo lakše razumjeli osnivanje rječnika trojki i petorki, poslužit ćemo se s primjerom kojeg smo nešto ranije naveli.

→ *MNQVLKDALEDNPIIVAIKDDAGLQKCKESESRIIFILYGDLLNIADIVD...*

Ako promotrimo prve tri aminokiseline (MNQ) navedenog proteina i usporedimo ih s prvom vrijednosti u rječniku proteina, uočit ćemo podudaranje. Primijetimo da se drugi element (NQV) rječnika trojki sastoji od druge, treće i četvrte aminokiseline u gore navedenom nizu. Analogno slijedi i za ostale elemente rječnika, kako trojki tako i petorki.

Rječnik trojki

Br.	**Riječ**	**F1**	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**	**F8**	**F9**
1.	**MNQ**	7	1	5	2	0	0	5	6	3
2.	**NQV**	2	0	83	8	3	10	8	17	13
3.	**QVL**	5	5	71	16	26	52	59	18	26
4.	**VLK**	8	7	55	11	34	5	175	38	31
5.	**LKD**	13	15	39	22	38	3	172	11	33
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
8107.	**XSK**	0	0	0	0	0	0	0	0	1

Tablica 5.3: Rječnik trojki

Broj elemenata rječnika trojki je 8107, dok rječnik petorki broji 356, 892 elemenata.

Ako broj aminokiselina označimo s n a duljinu riječi s r, jednostavnim računom dolazimo do ukupnog broja trojki, odnosno petorki.

$n = 20 + 1$
$r = 3$
$B_3 = 21^3 = 9261$

Naravno, radi se o broju svih trojki ali nas zanimaju različite trojke te redukcijom dolazimo do broja elemenata rječnika trojki i on iznosi 8107 elemenata.

Rječnik petorki

Br.	Riječ	F1	F2	F3	F4	F5	F6	F7	F8	F9
1.	**MNQVL**	1	0	0	0	0	0	0	0	0
2.	**NQVLK**	1	0	0	1	0	0	0	0	0
3.	**QVLKD**	1	0	0	0	0	0	0	0	0
4.	**VLKDA**	1	0	0	0	0	0	0	0	0
5.	**LKDAL**	1	0	0	0	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
356,892.	**RKEIA**	0	0	0	0	0	0	0	0	1

Tablica 5.4: Rječnik petorki

Istim postupkom dolazimo do broja petorki:

$n = 21$
$r = 5$
$B_5 = 21^5 = 4,084,101$

Broj svih kombinacija petorki je nešto veći od četiri milijuna no nakon redukcije dolazimo do $356,892$ elemenata rječnika petorki.

5.4 Vektorizacija trening seta

U ovom potpoglavlju ćemo vektorizirati i smjestiti naše podatke u vektorski prostor.

Nakon definiranja rječnika proteina te određivanja frekvencije elemenata rječnika za svaki od devet familija proteina možemo definirati matricu prediktora. Matrica prediktora se sastoji od vektor stupaca a svaki vektor stupac je prostorna interpretacija svake od 8107 riječi u odnosu na trening set i to na sljedeći način. Odaberemo jednu riječ, npr. "MNQ", prođemo kroz svih 1800 proteinskih nizova trening seta te u prvi stupac zapisujemo redom;

1.	Broj ponavljanja "MNQ" u 1. proteinskom nizu
2.	Broj ponavljanja "MNQ" u 2. proteinskom nizu
⋮	⋮
1800.	Broj ponavljanja "MNQ" u 1800. proteinskom nizu

Isti postupak provedemo za preostalih 8106 proteinskih riječi te dobivene vektore smjestimo u stupce matrice prediktora X, $X \in M^{1800\times 8107}$.

5.5 Vektorizacija validacijskog seta

Isti postupak provedemo za 900 proteinskih nizova te dobijemo validacijsku matricu W dimenzije 900×8107. Svaki vektor redak predstavlja jedan proteinski niz u 8107 dimenzionalnom prostoru. Uobičajeno je označavati koordinatne osi s X, Y, Z no u našem slučaju, nazivi koordinata su elementi rječnika. Zbog jednostavnosti neću označavati koordinatne osi no kada bi to bilo potrebno, naše koordinante osi bi nosile nazive MNQ, NQV, ..., XSK.

5.6 Optimalne razdvajajuće hiperravnine

Pomoću Python paketa *numpy, sklearn i scipy* pronalazimo optimalne razdvajajuće hiperravnine. Ukupan broj hiperravnina je 8 jer želimo podijeliti vektorski prostor na 9 dijelova. U poglavlju **Algoritam potpornih vektora** u općem primjeru u dvodimenzionalnom prostoru smo prikazali dva seta podataka i razdvajajuću hiperravninu (pravac) no zbog dimenzionalnosti vektorskog prostora u kojem su smješteni naši podaci ne možemo grafički prikazati odnos familija i hiperravnina.
Sljedeći korak je odrediti poziciju svakih od 900 vektor-redaka validacijskog seta s obzirom na razdvajajuće hiperravnine. Pozicija vektora u vektorskom prostoru određuje proteinsku familiju kojoj pripada. Jako je intuitivno i jednostavno razumjeti na koji se način klasificiraju proteinski nizovi validacijskog seta.

- Svaki proteinski niz ima vektorski prikaz te ga možemo prikazat kao točku u 8107 dimenzionalnom vektorskom prostoru.
- Svakoj točki možemo odrediti položaj u odnosu na hiperravninu.
- Položajem točke u odnosu na hiperravnine određujemo proteinsku familiju kojoj ta točka (vektorizirani proteinski niz) pripada.

Na sreću, nismo morali računati ručno položaj svake od 900 točaka u 8107 dimenzionalnom vektorskom prostoru jer su to za nas napravili algoritmi koji su sadržani u gore navedenim Python paketima.

5.7 Klasifikacija trening seta

U ovom potpoglavlju ću predstaviti rezultate klasifikacije proteinskih nizova s obzirom na rječnik trojki te rječnik petorki.

Rezultati klasifikacije-trojke

Primjenom klasifikacijskog algoritma *Algoritam potpornih vektora* na podacima s pripadnim rječnikom trojki dobili smo sljedeće rezultate.

Familija	**Točnost klasifikacije %**
Familija 1:	100%
Familija 2:	95%
Familija 3:	98%
Familija 4:	95%
Familija 5:	96%
Familija 6:	98%
Familija 7:	92%
Familija 8:	96%
Familija 9:	95%
Ukupno:	96.11%

Tablica 5.5: Rezultati modela trojki

Rezultat klasifikacije-petorke

Primjenom klasifikacijskog algoritma na podacima s pripadnim rječnikom petorki dobili smo sljedeće rezultate.

Familija	**Točnost klasifikacije %**
Familija 1:	99%
Familija 2:	88%
Familija 3:	97%
Familija 4:	98%
Familija 5:	93%
Familija 6:	100%
Familija 7:	83%
Familija 8:	89%
Familija 9:	91%
Ukupno:	93.11%

Tablica 5.6: Rezultati modela petorki

Ako usporedimo ukupnu točnost klasifikacije modela na rječnicima trojki i petorki, možemo zaključiti da je prvi model značajno bolji.
Možemo primijetiti da je familija 1 u prvom modelu najbolje klasificirana, dok je u drugom modelu isti slučaj kod familije 9. Isto tako, ako promotrimo raspon rezultata modela trojki i modela petorki možemo zaključiti kako je prvi model puno stabilniji $[92\%, 100\%]$ od drugog modela $[83\%, 100\%]$ zbog manjeg raspona između najlošije klasificirane familije i one najbolje.

Poglavlje 6

Primjena: Analiza glavnih komponenti

U prethodnom poglavlju smo klasificirali proteine u devet proteinskih familija te smo dobili jako dobre rezultate klasifikacije na rječnicima trojki i petorki. Međutim, kako su rezultati klasifikacije na rječniku trojki nešto bolji te je vrijeme klasifikacije podataka na prvom modelu kraće, daljnju analizu ćemo nastaviti s rječnikom trojki.

U poglavlju **Analiza glavnih komponenti** smo definirali korake algoritma glavnih komponenti te ćemo sada, slijedeći istu strukturu, primijeniti algoritam na podacima devet proteinskih familija.

6.1 Matrica prediktora

Zbog prevelike dimenzionalnosti matrice prediktora $X \in M^{1800\times 8107}$, prikaz iste u cijelosti nije moguće realizirati. Međutim, prikazat ćemo elemente koji se nalaze u rubnim pozicijama matrice X.

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}$$

6.2 Normalizacija podataka

Normalizacija vektor-stupaca matrice prediktora X je uobičajen dio postupka te se primjenjuje gotovo uvijek. Međutim, taj korak sam svjesno izostavio zbog male varijabilnosti vrijednosti stupaca matrice prediktora X.

Dakle, u nastavku rada vrijedi $R := X$.

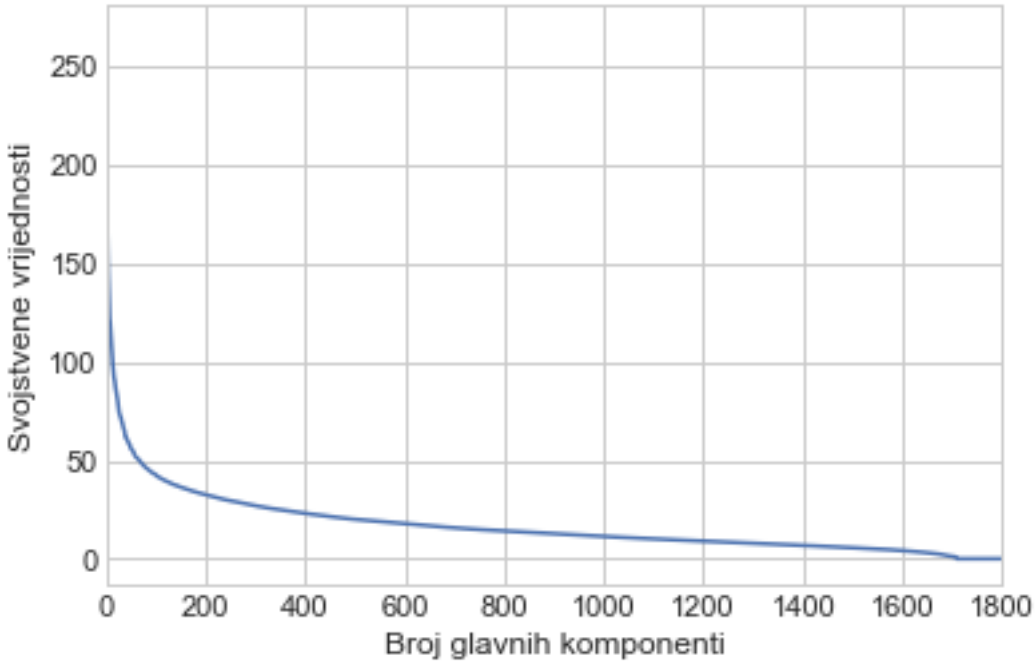
6.3 Svojstvene vrijednosti matrice L

Pozivajući se na prethodno potpoglavlje, vektori matrice X nisu normalizirani zbog jako male varijabilnosti podataka te vrijedi $R = X$. Svojstvene vrijednosti matrice $L = RR^T$, ($R \in M^{1800\times8107}$, $R^T \in M^{8107\times1800}$) su prikazane u sljedećoj tablici:

Naziv	**Vrijednost**
λ_1	$2.680\,785\,82 \times 10^2$
λ_2	$1.739\,003\,03 \times 10^2$
λ_3	$1.555\,697\,63 \times 10^2$
$\vdots$	$\vdots$
λ_{1798}	$4.580\,501\,40 \times 10^{-15}$
λ_{1799}	$4.553\,355\,95 \times 10^{-15}$
λ_{1800}	$4.392\,400\,53 \times 10^{-15}$

Tablica 6.1: Svojstvene vrijednosti

Iz prethodne tablice i grafičkog prikaza ispod možemo jasno vidjeti da su svojstvene vrijednosti sortirane silazno. Broj svojstvenih vrijednosti je 1800.



Slika 6.1: Svojstvene vrijednosti

6.4 Glavne komponente

U prethodnom poglavlju smo naveli kako je poredak svojstvenih vrijednosti bitan te znamo da svojstvene vrijednosti određuju svojstvene vektore. Svojstveni vektori, stupci matrice V, su međusobno ortogonalni, odnosno korelacija vektor-stupaca matrice V je 0. Matrica $G = VR = VX$ se sastoji od vektor-stupaca koji su nam jako bitni za osnivanje pojma glavnih komponenti. Pozivajući se na prethodnu karakterizaciju matrice V, znamo da su vektor-stupci matrice G međusobno ortogonalni što odgovara definiciji glavnih komponenti.
Međutim, svaki stupac matrice G je jedna točka u 1800 dimenzionalnom vektorskom prostoru te će nam pomoći pri osnivanju pojma glavnih komponenti. Kako su navedeni vektori međusobno ortogonalni, točke su linearno nezavisne. Odnosno, niti jedan par točaka ne leži u istoj ravnini.

Promotrimo sljedeće analogije:

- Da bismo odredili jedinstven pravac, potrebno nam je ishodište O i bilo koja druga

točka T, $T \neq O$.

- Da bismo odredili ravninu u dvodimenzionalnom prostoru potrebno nam je ishodište O i dvije točke T_1 i T_2 takve da T_2 ne leži na pravcu p određenog točkama O i T_1.
- Da bismo konstruirali N-dimenzionalnu ravninu potrebno nam je ishodište O i N točaka tako da su vektori $\overrightarrow{ON_i}$ međusobno nezavisni za svaki $i = 1, 2, ..., n; n \in \mathbb{N}$.

Na početku rada smo definirali da glavne komponente razapinju vektorski prostor u kojem su dani podaci disperzirani. Pozivajući se na karakteristike matrice G te na prethodnu konstrukciju n-dimenzionalne hiperravnine, stupci matrice G (ukupno 1800) razapinju 1799 dimenzionalni prostor a svaka od 1799 osi tako konstruiranog vektorskog prostora predstavlja jednu glavnu komponentu.

U matrici G_K su prikazane glavne komponente. Glavna vizualna razlika matrice G i matrice G_K je ta što matrica glavnih komponenti sadrži jedan vektor-stupac manje od matrice G. Gdje je nestao jedan stupac?

Kako su vektori matrice G međusobno ortogonalni te razapinju vektorski prostor u kojem se naši podaci nalaze, jedan vektor-stupac je ishodište (O) a ostali stupci su transformirani u glavne komponente (vektor-stupce matrice G_K).

$$G_K = \begin{bmatrix} 9.5940 \times 10^{-4} & 4.1734 \times 10^{-3} & \dots & 2.1337 \times 10^{-5} \\ 3.3058 \times 10^{-4} & 1.0760 \times 10^{-2} & \dots & 1.5866 \times 10^{-4} \\ 2.4263 \times 10^{-3} & 1.0462 \times 10^{-2} & \dots & -5.6480 \times 10^{-5} \\ \vdots & \vdots & \ddots & \vdots \\ 2.0730 \times 10^{-2} & -7.6216 \times 10^{-3} & \dots & 9.0988 \times 10^{-4} \\ -4.4667 \times 10^{-2} & 1.0407 \times 10^{-2} & \dots & -1.9435 \times 10^{-3} \\ -1.7915 \times 10^{-2} & 1.3216 \times 10^{-2} & \dots & -8.5816 \times 10^{-4} \end{bmatrix}$$

Konačno smo i matematički konstruirali pojam glavnih komponenti te nam preostaje odrediti broj glavnih komponenti koje opisuju dovoljno varijabilnosti podataka u modelu.

Poglavlje 7

Kriteriji zadržavanja glavnih komponenti

7.1 Metoda disperzije podataka

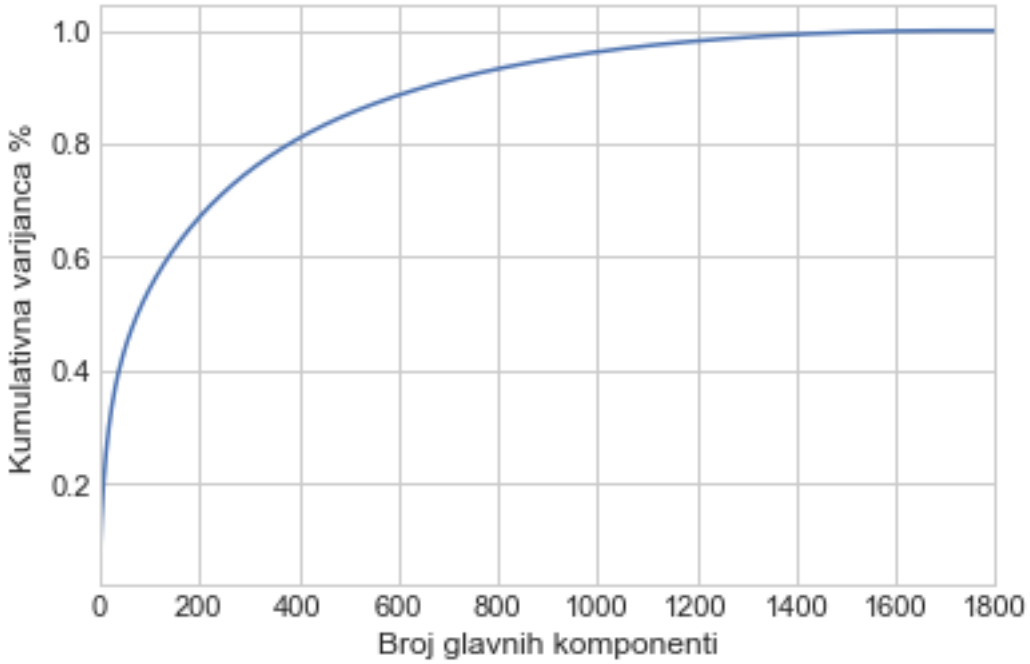
Suma svih svojstvenih vrijednosti matrice $L = RR^T$ predstavlja ukupnu opisanu varijabilnost podataka našeg modela. Nije teško pretpostaviti zašto su svojstvene vrijednosti sortirane od najveće do najmanje, što smo i prikazali u potpoglavlju **6.3 Svojstvene vrijednosti matrice L**. Ako zadržimo samo prvu glavnu komponentu $\lambda_1 = 268$ tada je postotak objašnjene raspršenosti podataka 0.87%. U tablici ispod su navedene određene svojstvene vrijednosti, postotak disperzije i kumulativna varijanca podataka. Postotak raspršenosti podataka se odnosi na modele koji sadrže glavne komponente zaključno s onom čija je svojstvena vrijednost prikazana u prvom stupcu. Dakle, model u kojem zadržavamo prvih 85 glavnih komponenti opisuje 20.14% raspršenosti podataka. Analogno vrijedi i za ostale vrijednosti.

λ_i	**Svojstvene vrijednosti**	**Raspršenost podataka %**	**Kumulativna varijanca %**
λ_1	= 268	0.87%	6.73%
λ_{10}	= 112	4.82%	22.31%
λ_{50}	= 56	14.46%	43.46%
λ_{85}	= 45.3	20.14%	51.61%
λ_{100}	= 42.6	22.28%	54.29%
λ_{200}	= 32.5	34.23%	66.94%
λ_{400}	= 23	51.97%	80.92%
λ_{800}	= 14.1	75.38%	93.22%
λ_{1600}	= 4.17	99.00%	99.91%
λ_{1799}	$= 4.39 \times 10^{-15}$	100.00%	100.00%

Tablica 7.1: Udio raspršenosti podataka

7.2 Kaiser-Guttman kriterij

U modelima gdje je matrica prediktora X skalirana ili normalizirana, Kaiser-Guttman kriterij bi vrijedio kao što smo definirali u potpoglavlju **4.2 Kaiser-Guttman kriterij**. Dakle, zadržali bismo sve glavne komponente čije su pripadne svojstvene vrijednosti veće od 1. (U takvim modelima, prosjek svojstvenih vrijednosti je 1).
Kako podaci u modelu kojeg sam predstavio nisu skalirani, prema Kaiser-Guttmann kriteriju bismo trebali zadržati sve glavne komponente čije su pripadne svojstvene vrijednosti veće od prosjeka svih svojstvenih vrijednosti.
Prosjek svih svojstvenih vrijednosti je $p = 17.03$ a broj glavnih komponenti čije su svojstvene vrijednosti veće od p je 631. Pozivajući se na prethodni kriterij, ukupna opisana varijabilnost podataka opisanih sa 631 komponentom je 89.42%.

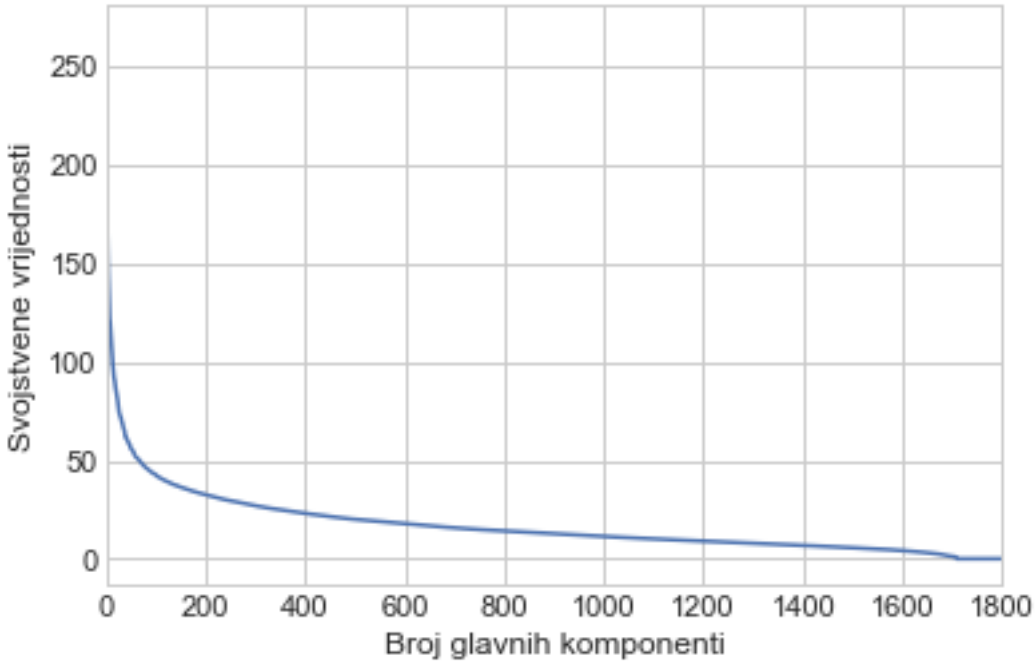


Slika 7.1: Kaiser-Guttman kriterij

7.3 Metoda lakta

Posljednji kriterij zadržavanja glavnih komponenti, kojeg smo ranije okarakterizirali kao nepouzdan, koristit ćemo samo kako bismo rezultate promotrili iz druge perspektive.
Ranije smo naveli kako vrijednosti svojstvenih vrijednosti određuju raspršenost podataka te shodno navedenom, cilj je zadržati sve svojstvene vrijednosti koje određuju najviše raspršenosti. Metoda lakta nalaže da odbacimo sve svojstvene vrijednosti nakon "lakta", mjesta na grafu gdje se događa najveći pad u vrijednostima svojstvenih vrijednosti.

Ako promotrimo graf ispod, možemo zaključiti kako je teško odrediti gdje se točno nalazi "lakat". Lakat se sigurno nalazi između λ_1 i λ_{200} te je to najbolja aproksimacija.



Slika 7.2: Metoda lakta

Poglavlje 8

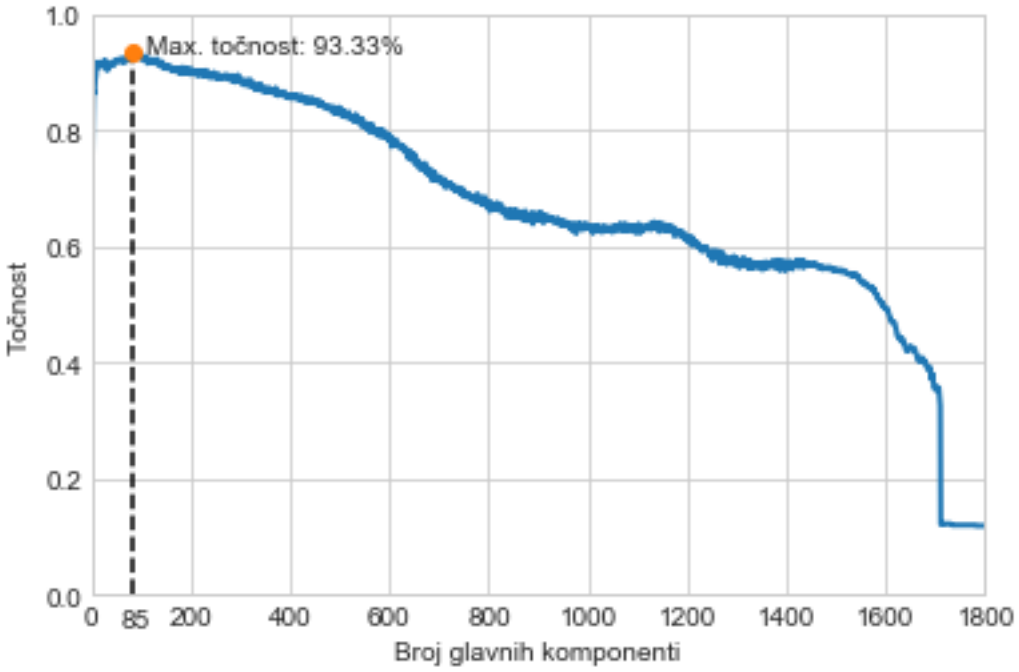
Rezultati nakon redukcije

Posljednji dio analize ovog rada je empirijske naravi. Kao što sam na početku rada naveo, točnost klasifikacije i redukcija dimenzionalnosti su nam od primarne važnosti. Pozivajući se na kriterije iz prethodnog poglavlja sljedeći broj komponenti moramo zadržati.

- Prema metodi disperzije podataka moramo zadržati 382 glavne komponente kako bismo opisali 80% varijabilnosti podataka.
- Prema Kaiser-Guttman kriteriju bismo trebali zadržati 631 glavnu komponentu te bismo u tom slučaju opisali 66.86% varijabilnosti podataka.
- Prema metodi lakta bismo trebali zadržati između 1 i 200 glavnih komponenti te bismo u tom slučaju maksimalno opisali 66.93% varijabilnosti podataka.

Koliko glavnih komponenti moramo zadržati kako bismo postigli maksimalnu točnost klasifikacije te koji kriterij pri tome odabrati?
Znamo da možemo zadržati jednu glavnu komponentu, nekoliko ili sve glavne komponente. Međutim, za koji broj glavnih komponenti postižemo najbolju točnost klasifikacije?

Na grafu ispod je prikazani rezultati klasifikacije za modele u kojima smo zadržali od 1 do 1799 glavnih komponenti.



Slika 8.1: Broj glavnih komponenti i točnost klasifikacije

Rezultati empirije nam sugeriraju kako bismo trebali zadržati 85 komponenti da bismo postigli maksimalnu točnost klasifikacije. Točnost klasifikacije za model koji zadržava prvih 85 glavnih komponenti je 93.33%. Na prethodnom grafu možemo vidjeti odnos točnosti klasifikacije u odnosu na broj glavnih komponenti u modelu. Možemo uočiti kako točnost klasifikacije raste te postiže maksimum za model koji zadržava 85 glavnih komponenti. Nakon toga, uz nekoliko iznimki, pada te minimum postiže za model koji zadržava 1799 glavnih komponenti. Povećanjem broja komponenti rastu opisana varijabilnost i disperzija podataka ali zašto točnost klasifikacije pada već nakon modela s 85 glavnih komponenti? Pretpostavimo da na prvih 85 glavnih komponanti imamo najbolje ("najduže") disperzije podataka. Ako dodamo 86. glavnu komponentu u naš model čiji udio objašnjene disperzije nije značajan (ali je najveći od preostalih), povećavamo dimenziju prostora u kojem se nalaze naši podaci. Kako nam je cilj bio postići maksimalan učinak klasifikacije uz što manju dimenzionalnost prostora, odlučio sam se za maksimalan učinak klasifikacije!

Napomena: *Vodeći se predhodnom analogijom, krivulja na grafu bi trebala padati nakon maksimuma. Razlog zašto krivulja raste u određenim modelima su nenormalizirani stupci matrice prediktora.*

Poglavlje 9

Zaključak

Provedenu analizu klasifikacije i redukcije dimenzije bih okarakterizirao kao uspješnu. Krenuli smo od problema semantičkog indeksiranja i iterativnog pretraživanja dokumenata te rekonstruirali problem u klasifikacijski. Odabrao sam familije proteina za koje ne postoji rječnik po kojem možemo odrediti strukturu iste te mi je zbog toga problem postao još izazovniji. Umjesto karakterističnih motiva pomoću metode iterativnog pretraživanja, konstruirao sam rječnike n-grama te odlučio se za klasifikaciju pomoću algoritma *Algoritam potpornih vektora*. Rezultati klasifikacije proteina u proteinske familije su se pokazali boljima od očekivanja te sam se odlučio za korak dalje. *Analizom glavnih komponenti* sam reducirao dimenziju prostora u kojem su smješteni dani podaci te ponovnom klasifikacijom na prostoru značajno manjih dimenzija postigao odlične rezultate. Dakle, kao što smo i priželjkivali, značajno smo reducirali dimenziju prostora ($8107 \rightarrow 85$) a da pritom nismo značajno izgubili na točnosti klasifikacije ($96.11\% \rightarrow 93.33\%$).

Tema je opširna te kada bih nastavio s daljnjom analizom posvetio bih vremena karakterizaciji svake pojedine proteinske familije. Pokušao bih pronaći ključne karakteristične motive koji su nužni i dovoljni za opisivanje familije a da pritom promatram svaku familiju odvojeno.

Bibliografija

[1] M. Polonijo, D. Crnković, M. Bombardelli, T. Ban Kirigin, Z. Franušić, R. Sušanj, Z. Iljauović, *Euklidski prostori*, 2008.

[2] Ž. Milin Šipuš, M. Bombardelli, *Analitička geometrija*, 2016.

[3] D. Bakić, *Linearna algebra*, 2008.

[4] G. Muić, M. Primc, *Vektorski prostori*

[5] S. Miličić, *Linearna algebra 2*, 2004.

[6] N. Cristianini, J. Shawe-Taylor, *An introduction to Support Vector Machines*, Cambridge University Press, London, 2000.

[7] P. Kumar Kolluru, *SVM Based Dimensionality Reduction and Classification of Hyperspectral Data*, 2013.

[8] A. Li, J. Zhang, Z. Zhou, *PLEK: a tool for predicting long non-coding RNAs and messenger RNAs based on an improved k-mer scheme*, BMC Bioinformatics 15:311, 2014.

[9] G. Salton, C. Buckley, *Term-weighting approaches in automatic text retrieval, In Information Processing Management Volume 24, Issue 5*, 1988.

[10] Y Li, CY. Chen, WW Wasserman, *Deep Feature Selection: Theory and Application to Identify Enhancers and Promoters*, 2016.

Sažetak

Cilj ovog rada je bio pronaći glavne karakteristične motive devet proteinskih familija metodom iterativnog pretraživanja. Problem sam transformirao u klasifikacijski te karakteristične motive zamijenio s n-gramima. Problemi s kojima sam se susreo u ovom radu su određivali smjer u kojem sam se kretao kako bih došao do konačnog željenog rezultata. Pronašao sam proteinske rječnike koji predstavljaju glavne karakteristike proteinskih familija, klasificirao proteinske nizove te reducirao dimenziju prostora. Konačno, rezultati nakon redukcije su se pokazali izrazito kvalitetnim jer sam značajno smanjio dimenziju prostora a da se pritom točnost klasifikacije nije značajno promijenila.

Summary

The purpose of this thesis was to find the main distinctive motifs of the nine protein families using the iterative method. I transformed the problem into one of a classification nature and replaced the distinctive motifs with n-grams. The problems I faced throughout my research determined the direction in which I proceeded so that I could get to the final and wanted result. I determined the protein dictionary which represent the main characteristics of protein families, after which I classified the protein sequences and reduced the space dimension. Finally, the results after the reduction proved to be of a tremendously great quality as I have significantly reduced the space dimension without significantly changing the accuracy of the classification.

Životopis

Rođen sam 22. 09. 1993. godine u Vinkovcima. Osnovnu školu sam pohađao u Osnovnoj Školi Antun i Stjepan Radić Gunja u općini Gunja. U Gunji sam proveo cijelo djetinjstvo te svoje srednjoškolsko obrazovanje nastavio u Općoj Gimnaziji Županja u gradu Županja. U konačnici upisujem preddiplomski sveučilišni studij Matematika, nastavnički smjer na Prirodoslovno-matematičkom fakultetu u Zagrebu te paralelno pohađam civilno-vojni program Kadet u sklopu Oružanih snaga Repulike Hrvatske. Nakon uspješno završenog preddiplomskog studija i vojnog programa Kadet odlučio sam se za diplomski studij Matematička statistika na istom fakultetu, kojeg upravo završavam. Na prvoj godini diplomskog studija sam počeo raditi u Razvojnoj Agenciji Zagreb (ZICER) na poziciji analitičara te nakon 14 mjeseci uspješne suradnje svoju karijeru nastavljam u Schur Flexibles Holding GesmbH na poziciji Junior Business Intelligence Specialist u Austriji gdje trenutno živim i planiram nastaviti svoju karijeru...