Matematički fakultet,
Univerzitet u Beogradu

Fakultetski projekat

Analiza društvenih mreža i spektralno grupisanje u grafovima i mrežama korišćenjem R programskog jezika

---

Natalija Lazić
Dejana Miladinović
Staša Tonić

decembar 2022.

Sadržaj

1 Analiza društvenih mreža **3**

2 Teorija grafova - Uvod i osnovni pojmovi **5**
- 2.0.1 Geometrijska reprezentacija grafa . . . 5
- 2.0.2 Stepen čvora . . . 5
- 2.0.3 Povezanost grafova . . . 6
- 2.0.4 Matrično prikazivanje grafova . . . 7
- 2.1 Prikaz grafova u *R*-u . . . 7
  - 2.1.1 Full Graph . . . 7
  - 2.1.2 Ring Graph . . . 8
  - 2.1.3 Star Graph . . . 9
  - 2.1.4 Sample Graphs . . . 10
- 2.2 Mreža . . . 13
  - 2.2.1 Primer mreže . . . 13

3 Centralnost čvora u mreži **17**
- 3.1 Centralnost stepena . . . 17
  - 3.1.1 Normalizacija vrednosti centralnosti stepena . . . 18
- 3.2 Centralnost između . . . 18
- 3.3 Centralnost bliskosti . . . 20
- 3.4 Centralnost sopstvenog vektora . . . 21
  - 3.4.1 Matrica povezanosti . . . 21
  - 3.4.2 Sopstveni vektori . . . 21
  - 3.4.3 Važnost i upotreba metode . . . 22
- 3.5 Vizualizacija važnih čvorova u grafu . . . 23

4 Analiza klastera **25**
- 4.1 Definicija i osnovni pojmovi . . . 25
  - 4.1.1 Merenje grupisanja u klaster analizi . . . 25
- 4.2 Tipovi algoritama u analizi klastera . . . 26
- 4.3 Proces primene analize klastera na podacima . . . 27

5 Spektralno grupisanje **27**
- 5.1 Matematički prikaz spektralnog grupisanja . . . 28
- 5.2 Implementacija spektralnog gupisanja . . . 28
- 5.3 Matrica afiniteta . . . 28
  - 5.3.1 Izračunavanje matrice afiniteta korišćenjem *The k-nearest neighbor (KNN)* algoritma . . . 29
  - 5.3.2 Izračunavanje matrice afiniteta korišćenjem *Gausovog metoda jezdra* . . . 30
- 5.4 Laplasova matrica . . . 31
  - 5.4.1 Nenormalizovana Laplasova matrica . . . 31
  - 5.4.2 Normalizovana Laplasova matrica . . . 32
- 5.5 Sopstveni vektori Laplasove matrice . . . 32
- 5.6 Grupisanje dobijenih podataka - *K-means clustering* . . . 32
  - 5.6.1 Metode odabira broja k - Pravilo lakta . . . 33

5.6.2 Metode odabira broja k - Analiza silueta 34
5.7 Spektralno grupisanje - prednosti i mane 34
5.8 Primer spektralnog klasterovanja 35

6 Zaključak **41**

1 Analiza društvenih mreža

Social network analysis, tj. analiza društvenih mreža (konekcija) je podoblast sociologije i antropologije koja se fokusira na proučavanje društvenih mreža i obrazaca odnosa koji postoje unutar njih. To uključuje upotrebu različitih alata i tehnika za analizu i vizuelizaciju ovih odnosa.

Neke uobičajene primene analize društvenih mreža uključuju:

- Proučavanje širenja ideja ili ponašanja unutar društvene mreže, kao što je usvajanje nove tehnologije ili širenje bolesti.
- Analiziranje formiranja i evolucije društvenih grupa ili zajednica, kao što je formiranje grupa prijateljstva ili pojava liderskih uloga.
- Razumevanje obrazaca komunikacije ili saradnje unutar društvene mreže, kao što je tok informacija ili resursa unutar mreže.
- Identifikovanje ključnih igrača ili važnih čvorova u društvenoj mreži, kao što su lideri mišljenja ili najuticajniji pojedinci.

Analiza društvenih mreža uključuje upotrebu širokog spektra matematičkih koncepata i tehnika za razumevanje i analizu obrazaca odnosa unutar društvenih mreža.

Jedan od ključnih alata u analizi društvenih mreža je teorija grafova, grana matematike koja proučava grafove i njihova svojstva. Graf je matematički prikaz skupa objekata i odnosa između njih. U kontekstu analize društvenih mreža, objekti su tipično pojedinci ili organizacije, a odnosi između njih predstavljaju neki oblik društvene veze ili interakcije. Teorija grafova pruža okvir za predstavljanje i analizu strukture društvenih mreža i omogucava istraživačima da kvantifikuju različite aspekte mreže, kao što su njena veličina, gustina i povezanost.

Još jedno važno sredstvo u analizi društvenih mreža je linearna algebra, grana matematike koja proučava vektorske prostore i linearne transformacije. Linearna algebra se koristi za analizu sopstvenih vektora i sopstvenih vrednosti matrica, što može da pruži uvid u strukturu grafa i odnose između njegovih čvorova. Spektralno grupisanje, tema našeg rada, je jedna od tehnika koja se u velikoj meri oslanja na linearnu algebru.

Drugi matematički koncepti koji se često koriste u analizi društvenih mreža uključuju teoriju verovatnoće, statistiku i optimizaciju. Ovi alati omogućavaju istraživačima da modeliraju i analiziraju tok informacija ili uticaja unutar mreže i da naprave predviđanja o ponašanju mreže kao celine.

U analizi društvenih mreža mreže su predstavljene kao grafovi, pri čemu su pojedinci ili organizacije predstavljeni kao čvorovi, a njihovi odnosi su predstavljeni kao veze (linije). Ovo omogućava istraživačima da proučavaju strukturu i dinamiku društvenih mreža i kako one utiču na ponašanje i ishode.

Postoji mnogo različitih mera i matematičkih tehnika koje se koriste za analizu mreža, uključujući:

- Mere centralnosti - koriste se za identifikaciju najuticajnijih ili centralnih čvorova u mreži. Ovi čvorovi mogu igrati centralnu ulogu u strukturi zajednice mreže.

- Analiza klastera - tehnika koja koristi algoritme, kao što su k-srednja vrednost ili hijerarhijsko grupisanje, da grupiše čvorove u klastere na osnovu obrazaca veza unutar mreže. Klasteri se mogu tumačiti kao zajednice ili podgrupe unutar mreže. Ovi algoritmi obično uključuju minimiziranje funkcije cilja, kao što je zbir kvadrata unutar klastera ili zbir kvadrata između klastera, korišcenjem tehnika optimizacije kao što su gradijentni spuštanje ili Njutnov metod.

- Mrežna udaljenost - mera odražava broj koraka potrebnih da se dođe od jednog čvora do drugog u mreži. Mala mrežna udaljenost može ukazivati na visok stepen povezanosti unutar mreže.

- Dinamika mreže - uključuje proučavanje kako se mreže menjaju tokom vremena i kako reaguju na različite stimuluse ili intervencije.

Social network analysis se može koristiti za proučavanje širokog spektra pitanja i problema u vezi sa društvenim mrežama, uključujuci razumevanje načina na koji se informacije šire unutar mreže, identifikaciju ključnih igrača i uticajnih osoba i poboljšanje komunikacije i saradnje unutar organizacije ili zajednice.

2 Teorija grafova - Uvod i osnovni pojmovi

Definicija 2.1. Graf G je uređen par $(V(G), E(G))$, gde je $V(G)$ konačan neprazan skup elemenata koji se zovu čvorovi, a $E(G)$ je konačan skup parova skupa $V(G)$ koji se zovu grane.

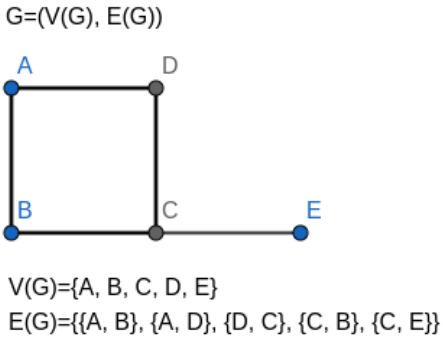
Grafovi se mogu podeliti na *usmerene* i *neusmerene* grafove:

- *Usmereni grafovi* - Skup čvorova povezanih granama, gde grane imaju svoj pravac. Redosled kojim se navode čvorovi je važan, u skupu $V(G)$ nalaze se uređeni parovi.
- *Neusmereni grafovi* - Skup čvorova povezanih granama, gde pravac grana nije definisan. Redosled kojim se navode čvorovi nije važan, u skupu $V(G)$ nalaze se neuređeni parovi.

.

2.0.1 Geometrijska reprezentacija grafa

Graf G se može geometrijski predstaviti crtežom u ravni. Čvorovi grafa se predstavljaju tačkama ravni, a grane grafa linijama koje povezuju odgovarajuće čvorove.



Слика 1: Predstavljanje neusmerenog grafa u ravni

Geometrijska reprezentacija usmerenog i neusmernog grafa se razlikuju, kod usmerenog grafa grane se crtaju kao strelice koje se usmeravaju od jednog čvora ka drugom, dok su kod neusmernog grafa grane crtaju kao obične duži.

2.0.2 Stepen čvora

Definicija 2.2. Stepen čvora, u oznaci d, jednak je broju grana koje su tom čvoru incidentne.

Primer. U slučaju grafa sa prethodne slike izračunaćemo stepen čvorova. To ćemo uraditi tako što ćemo izbrojati broj grana koje mu pripadaju:

$$\begin{aligned} d(E) &= 1 \\ d(A) &= d(B) = d(D) = 2 \\ d(C) &= 3 \end{aligned}$$

Definicija 2.3. Neka postoji graf G sa n čvorova i m grana. Tada važi da je suma stepena svi čvorova jednaka dvostrukom broju grana grafa G, odnosno:

$$\sum_{i=1}^{n} d_i = 2m \tag{1}$$

Ako se radi o grafu većeg obima ova formula je teže primenjiva, pa se koristi malo drugačiji pristup.

Definicija 2.4. Neka imamo skup $R(d) = \{1, 2.....m\}$ svih mogućih stepena čvorova u grafu. Možemo da definišemo udeo stepena nekog čvora u grafu:

$$p_d = \frac{n_d}{n} \tag{2}$$

gde n_d predstavlja broj čvorova čiji je stepen d, n je ukupan broj čvorova.

Primer. Uzmimo već pomenuti graf kako bismo pokazali primenu ove formule.

$$R = \{1, 2, 3\}$$

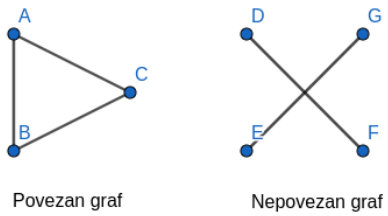
$$p_1 = \frac{n_1}{n} = \frac{1}{5} = 0.2 \qquad p_2 = \frac{n_2}{n} = \frac{3}{5} = 0.6 \qquad p_3 = \frac{n_3}{n} = \frac{1}{5} = 0.2$$

2.0.3 Povezanost grafova

Definicija 2.5. Šetnja u grafu G je konačan neprazan niz $W = v_0e_1v_2e_2.....e_kv_k$ u kome se smenjuju čvorovi i grane, pri čemu su v_i i v_{i-1} krajnji čvorovi grane e_i. Šetnja koja počinje u čvoru v_0, a završava se u čvoru v_k je $(v_0 - v_k)$ šetnja.

- *Dužina šetnje* je broj njenih grana.
- *Staza* je šetnja u kojoj se grane ne ponavljaju.
- *Put* je staza u kojoj su čvorovi različiti.

Definicija 2.6. Čvorovi u i v su povezani u grafu G, ako u G postoji (u, v)-put. Graf G je povezan ako su svaka dva njegova čvora povezana.



Слика 2: Predstavljanje grafa u ravni

Povezanost grafa je bitna osoba za analizu mreža. Naime, grupe ili pojedince ćemo mnogo lakše analizirati u povezanom grafu jer između njih postoje jasno definisane veze.

2.0.4 Matrično prikazivanje grafova

Pored klasičnog prikazivanja grafova na koje smo navikli unutar matematike, grafove možemo predstaviti i pomoću matrice. Pretpostavimo da imamo matricu A. Tada će njeni elementi biti:

- $a_{i,j} = 1$ - Ukoliko izmedju čvorova V_i i V_j postoji grana (veza).
- $A_{i,j} = 0$ - Ukoliko izmedju čvorova V_i i V_j ne postoji grana (veza).

Primer. Uzmimo već pomenuti graf sa *slike*1 i predstavljamo ga putem matrice.

	A	B	C	D	E
A	0	1	0	1	0
B	1	0	1	0	0
C	0	1	0	1	1
D	1	0	1	0	0
E	0	0	1	0	0

Na *slici*1 možemo videti sledeće:

- Između čvorova B i D nema grane, te se u matrici na toj poziciji nalazi 0
- Između čvorova C i D postoji grana, te se u matrici na toj poziciji nalazi 1

Ovaj način zapisivanja je takođe pogodan za izračunavanje stepena čvorova. Vidimo da ako saberemo vrstu ili kolonu dobićemo stepen čvora kojoj ta vrsta ili kolona pripada. Ovakav prikaz grafa je pogodniji za matematičke manipulacije. Postoje dva uobičajena načina da se graf predstavi pomocu matrice: matrica povezanosti i matrica incidencije.

2.1 Prikaz grafova u R-u

Programski jezik R pruža neke vrlo jednostvane i brze alate da prikazivanje grafova, kao i za konvertovanje naših podataka u graf. Neophodan paket za baratanje njima je *igraph*. Nakon zadavanja svih informacija o grafu, njega možemo vizualizovati pomoću funkcije *plot*().

Postoji veliki broj funkcija koje se koriste za prikazivanje grafova u analizi društvenih mreža.

2.1.1 Full Graph

Full Graph je funkcija koja se koristi za crtanje potpunog grafa.

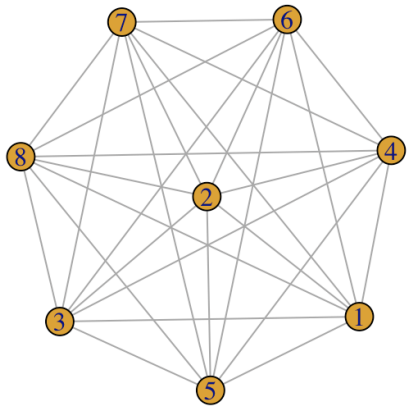
```
make_full_graph(n, directed = FALSE, loops = FALSE)
```

- n - Broj čvorova u grafu
- *directed* - $TRUE \backslash FALSE$, da li se kreira usmereni graf ili ne
- *loops* - $TRUE \backslash FALSE$, da li se dodaju petlje, tj. da li čvorovi treba da budu povezani sami sa sobom ili ne

Primer.

```
install.packages("igraph")
library(igraph)

Full_Graph <- make_full_graph(8, directed = FALSE)
plot(Full_Graph)
```



Слика 3: Primer crtanja potpunog grafa

2.1.2 Ring Graph

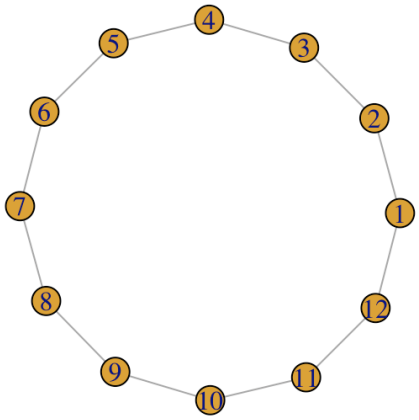
Prstenasti graf je jednodimenziona rešetka i funkcija koja se koristi za njegovo crtanje je poseban slučaj funkcije *make_lattice*(). Sintaksa ove funkcije je:

```
make_ring(n, directed = FALSE, circular = TRUE, mutual = FALSE)
```

- *n* - Broj čvorova u grafu
- *directed* - $TRUE \backslash FALSE$, da li se kreira usmereni graf ili ne
- *circular* - $TRUE \backslash FALSE$, da li kreirati cirkularan prsten ili ne. Necirkularni prsten je linija
- *mutual* - $TRUE \backslash FALSE$, da li su usmerene grane zajedničke ili ne, informacija koja se kod neusmerenih grafova ignoriše

Primer.

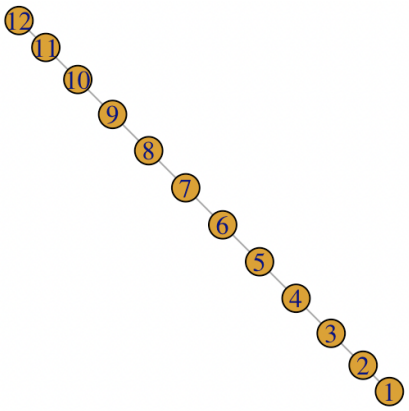
```
library(igraph)
Ring_Graph <- make_ring(12, directed = FALSE, mutual = FALSE, circular = TRUE)
plot(Ring_Graph)
```



Слика 4: Primer crtanja cirkularnog prstenastog grafa

Primer.

```
Ring_Graph <- make_ring(12, directed = FALSE, mutual = FALSE, circular = FALSE)
plot(Ring_Graph)
```



Слика 5: Primer crtanja necirkularnog prstenastog grafa

2.1.3 Star Graph

Zvezdani graf je graf gde je svaki čvor povezan samo sa centralnim čvorom i nijednim drugim. Funkcija koja se koristi za njegovo crtanje je:

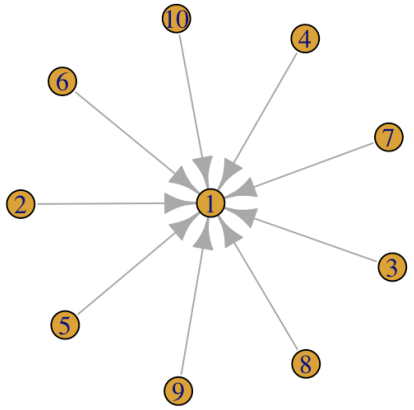
```
make_star(n,center = 1, mode = c( "in", "out","mutual", "undirected")
```

- n - Broj čvorova u grafu
- *center* - ID centralnog čvora

- *mode* - Definiše usmerenost ivica i može biti: *in* - grane pokazuju ka centru, *out* - grane pokazuju suprotno od centra, *mutual* - kreira se usmeren zvezdani graf sa zajedničkim granama i *undirecteed* - ivice nisu usmerene

Primer.

```
library(igraph)
Star_Graph <- make_star(10, center = 1)
plot(Star_Graph)
```



Слика 6: Primer crtanja zvezdanog grafa

2.1.4 Sample Graphs

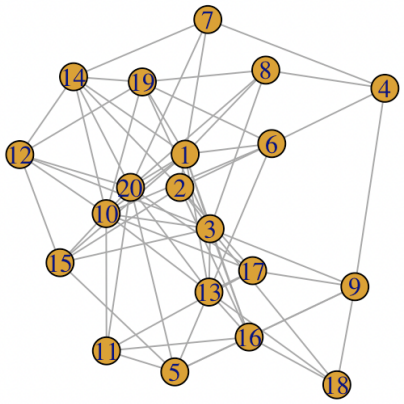
Grafovi se takođe mogu nasumično generisati sa zadatom konstantnom verovatnoćom kreiranja ivice.

```
sample_gnp(n, p, loops = FALSE, directed = FALSE)
```

- *n* - Broj čvorova u grafu
- *p* - Verovatnoća crtanja grane između dva nasumična čvora
- *directed* - $TRUE \backslash FALSE$, da li se kreira usmereni graf ili ne
- *loops* - $TRUE \backslash FALSE$, da li se dodaju petlje, tj. da li čvorovi treba da budu povezani sami sa sobom ili ne

Primer.

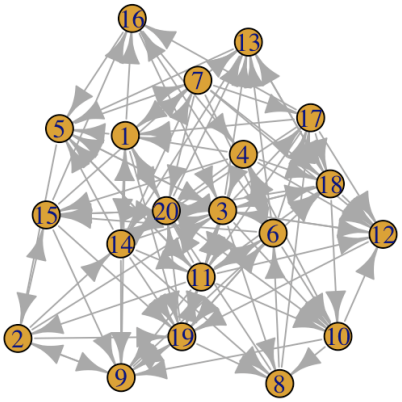
```
library(igraph)
gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
plot(gnp_Graph)
```



Слика 7: Primer generisanja nasumičnog grafa sa 20 čvorova i verovatnoćom formiranja ivica 0.3

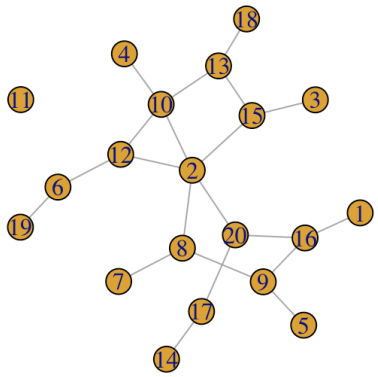
Primer.

```
library(igraph)
gnp_Graph <- sample_gnp(20, 0.3, directed = TRUE, loops = FALSE)
plot(gnp_Graph)
```



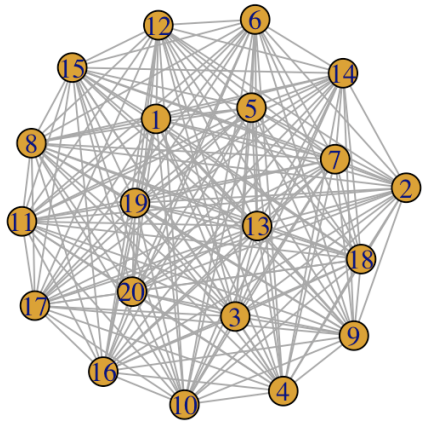
Слика 8: Primer generisanja nasumičnog usmerenog grafa sa 20 čvorova i verovatnoćom formiranja ivica 0.3

```
library(igraph)
gnp_Graph <- sample_gnp(20, 0.1, directed = FALSE, loops = FALSE)
plot(gnp_Graph)
```



Слика 9: Primer generisanja nasumičnog grafa sa 20 čvorova i verovatnoćom formiranja ivica 0.1

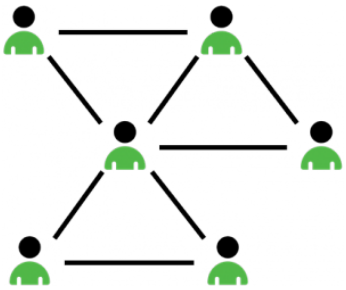
```
library(igraph)
gnp_Graph <- sample_gnp(20, 1, directed = FALSE, loops = FALSE)
plot(gnp_Graph)
```



Слика 10: Primer generisanja nasumičnog grafa sa 20 čvorova i verovatnoćom formiranja ivica 1

2.2 Mreža

Mreža se definiše kao graf, čvorovi kod tog grafa se nazivaju vrhovi ili akteri, dok se grane nazivaju vezama ili ivicama. Veze u socijalnim mrežama (Social Network) su prijateljstvo, ljubavni odnosi, poslovna saradnja i slično. Zamislimo da imamo grupu studenata na fakultetu, svaki čvor predstavlja jednog studenta dok poznanstvo predstavlja vezu između njih - to bi bio primer socijalne mreže.



Слика 11: Mreža studenata

Socijalne mreže najčešće nisu jednostavne kao na slici iznad. Obično su to puno veći grafovi koje treba ispitati,te se zato se za njihovu analizu koriste napredne statističke metode, kao i mašinsko učenje.

2.2.1 Primer mreže

Za primer analize društvenih mreža može se uzeti popularna igra *Six degress of Kevin Bacon*. Ova igra zasnovana je na konceptu šest stepeni separacije koji tvrdi da se svake dve osobe na planeti mogu povezati kroz šest poznanika ili manje. Ljubitelji filmova zadaju jedni drugima izazov da pronađu najkraći put između određenog glumca i Kevina Bacon-a, glumca sa veoma bogatom karijerom. Ova igra počiva na pretpostavci da se bilo ko iz holivudske filmske industrije može povezati kroz svoje uloge sa Bacon-om u okviru šest koraka. Igračima se zadaje izazov da povežu datog glumca sa Kevinom Bacon-om najbrže moguće i u što manje koraka. Na primer, Tom Hanks ima *Bacon rezultat* 1 jer je zajedno sa Bacon-om igrao u fimu Apollo 13.

U ovom primeru, koristićemo tri filma da napravimo mrežu između glumaca : *Apollo 13*, *Forest Gump* i *The rock*.

Da bismo učitali baze *Actors.csv* i *Movies.csv* koje se nalaze na GitHub-u, potreban nam je paket *readr*.

```
install.packages("readr")
library(readr)

actors <- read_csv("https://raw.githubusercontent.com/OPER682-Tucker/Social-Network-Analysis/master/Actors.csv")
movies <- read_csv("https://raw.githubusercontent.com/OPER682-Tucker/Social-Network-Analysis/master/Movies.csv")

actors
movies
```

```
> actors
# A tibble: 8 × 3
  Actor         Gender BestActorActress
  <chr>         <chr>  <chr>
1 Tom Hanks     Male   Winner
2 Gary Sinise   Male   None
3 Robin Wright  Female None
4 Bill Paxton   Male   None
5 Kevin Bacon   Male   None
6 Ed Harris     Male   Nominated
7 Sean Connery  Male   None
8 Nicolas Cage  Male   Winner
>
```

Слика 12: Baza Actors

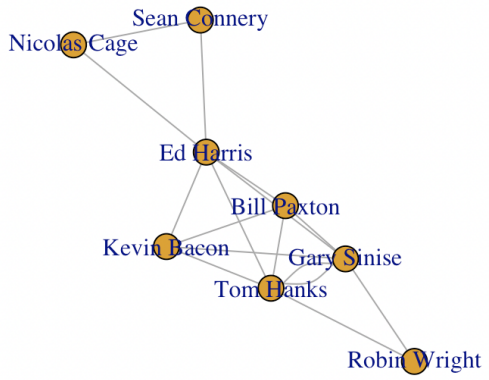
```
> movies
# A tibble: 16 × 3
   `Actor 1`     `Actor 2`     Movie
   <chr>         <chr>         <chr>
 1 Tom Hanks     Gary Sinise   Forest Gump
 2 Tom Hanks     Robin Wright  Forest Gump
 3 Gary Sinise   Robin Wright  Forest Gump
 4 Tom Hanks     Gary Sinise   Apollo 13
 5 Tom Hanks     Bill Paxton   Apollo 13
 6 Tom Hanks     Kevin Bacon   Apollo 13
 7 Tom Hanks     Ed Harris     Apollo 13
 8 Gary Sinise   Bill Paxton   Apollo 13
 9 Gary Sinise   Kevin Bacon   Apollo 13
10 Gary Sinise   Ed Harris     Apollo 13
11 Bill Paxton   Kevin Bacon   Apollo 13
12 Bill Paxton   Ed Harris     Apollo 13
13 Kevin Bacon   Ed Harris     Apollo 13
14 Ed Harris     Sean Connery  The Rock
15 Ed Harris     Nicolas Cage  The Rock
16 Sean Connery  Nicolas Cage  The Rock
>
```

Слика 13: Baza Movies

Vidimo da se u bazi *Actors* nalaze imena glumaca, njihov pol i podatak da li su osvojili ili bili nominovani za nagradu za najboljeg glmuca, dok se baza *Movies* sastoji iz veza između glumaca.

Prvi korak pravljenja mreže je kreiranje *igraph* objekta. Funkcijom $graph\_grom\_data\_frame()$ kreiramo ovaj objekat od naših postojećih baza podataka. Parametar d predstavlja bazu podataka koja će sadržati podatke o ivicama koje treba kreirati, u našem slučaju to ce biti movies, a parametar *vertices* prihvata bazu podataka koja sadrži čvorove, u našem slučaju actors. Pošto je ovo mreža koja se sastoji od glumaca koji su glumili u raznim filmovima zajedno, graf će biti neusmeren, te će parametar *directed* biti FALSE. Potom dobijenu mrežu treba plotovati za šta je najlakši način korišćenje funkcije $plot()$.

```
actorNetwork <- graph_from_data_frame(d=movies, vertices = actors, directed = FALSE)
plot(actorNetwork)
```

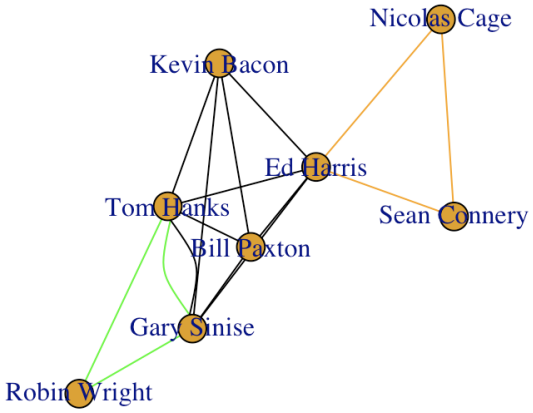


Слика 14: Мреža glumaca

Možemo prikazati i neke dodatne informacije, na primer ako želimo da znamo u kom filmu su glumci zajedno glumili, možemo ivice obojiti različitom bojom.

```
E(actorNetwork)$color <- ifelse(E(actorNetwork)$Movie == "Forest Gump", "green",
                                ifelse(E(actorNetwork)$Movie == "Apollo 13", "black",
                                       "orange"))

plot(actorNetwork)
```



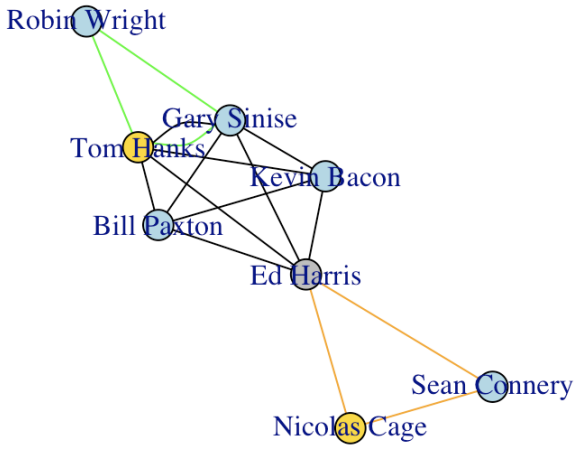
Слика 15: Мреža glumaca sa ivicama označenim različitom bojom u zavisnosti od filma u kom su glumci zajedno glumili

Sada su glumci koji su zajedno igrali u filmu Forrest Gump povezani zelenom bojom, Apollo 13 crnom i u filmu The rcok narandžastom. Takođe možemo predstaviti čvorove glumaca različitom bojom u zavisnosti od toga da li su osvojili nagradu za najboljeg glumca ili makar bili nominovani.

Takođe, može biti korisno koristite legende da prikažu šta koja boja na grafu znači.

```
V(actorNetwork)$color <- ifelse(V(actorNetwork)$BestActorActress == "Winner", "gold",
                                ifelse(V(actorNetwork)$BestActorActress == "Nominated","grey",
                                       "lightblue"))

plot(actorNetwork)
```

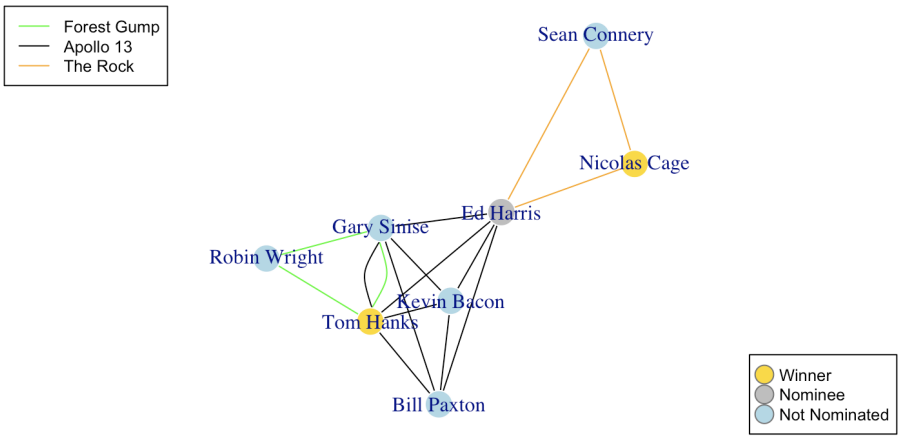


Слика 16: Мреža glumaca sa čvorovima označenim različitom bojom u zavisnosti od toga da li je dati glumac osnojio nagradu za najboljeg glumca ili bio nominovan

```
plot(actorNetwork, vertex.frame.color="white")

legend("bottomright", c("Winner","Nominee", "Not Nominated"), pch=21,
       col="#777777", pt.bg=c("gold","grey","lightblue"), pt.cex=2, cex=.8)


legend("topleft", c("Forest Gump","Apollo 13", "The Rock"),
       col=c("green","black","orange"), lty=1, cex=.8)
```



Слика 17: Мреža glumaca sa legendom koja objašnjava značenje svih boja

3 Centralnost čvora u mreži

U teoriji grafova, centralnost čvora u mreži je mera njegove važnosti ili uticaja unutar mreže. Postoji nekoliko načina da se kvantifikuje centralnost čvora, od kojih svaki obuhvata drugačiji aspekt njegove važnosti. Neke uobičajene mere centralnosti uključuju:

Centralnost stepena: Ova mera kvantifikuje broj veza koje čvor ima sa drugim čvorovima u mreži. Čvorovi sa visokim stepenom centralnosti su oni koji imaju mnogo veza i stoga su dobro povezani unutar mreže.

Centralnost između: Ova mera obuhvata obim do kojeg čvor leži na najkracem putu između drugih čvorova u mreži. Čvorovi sa visokom centralnošcu između su oni koji povezuju mnoge parove čvorova i stoga su važni za komunikaciju unutar mreže.

Centralnost bliskosti: Ova mera kvantifikuje koliko je čvor blizu svim drugim čvorovima u mreži, na osnovu najkracih putanja između njih. Čvorovi sa visokom centralnošcu bliskosti su oni koji su dobro povezani sa ostatkom mreže i mogu brzo da dođu do drugih čvorova.

Centralnost sopstvenog vektora: Ova mera kvantifikuje uticaj čvora na osnovu uticaja njegovih suseda. Čvorovi sa visokom centralnošcu sopstvenog vektora su oni koji su povezani sa drugim uticajnim čvorovima u mreži.

Ovo je samo nekoliko primera mera centralnosti koje se obično koriste u analizi mreže. Postoji mnogo drugih, od kojih svaki može biti manje ili više pogodan za određenu situaciju u zavisnosti od ciljeva analize i karakteristika mreže.

3.1 Centralnost stepena

Centralnost stepena je mera važnosti ili uticaja čvora u mreži na osnovu broja veza koje ima sa drugim čvorovima. Čvorovi sa visokim stepenom centralnosti su oni koji imaju mnogo veza i stoga su dobro povezani unutar mreže.

Da bismo izračunali stepen centralnosti čvora, jednostavno brojimo broj veza koje su incidentne tom čvoru. Na primer, u neusmerenom grafu, stepen centralnosti čvora je jednostavno broj suseda koji ima. U usmerenom grafu, centralnost stepena se može izračunati odvojeno za dolazne veze (broj čvorova koji imaju vezu koja pokazuje na čvor) i odlazne veze (broj čvorova na koje čvor ima vezu koja pokazuje).

Centralnost stepena se često koristi za identifikaciju najuticajnijih ili centralnih čvorova u mreži, kao što su ključni igrači u društvenoj mreži ili najvažnije stranice na World Wide Web-u. Takođe je korisno za identifikaciju strukture mreže, pošto su čvorovi sa visokim stepenom centralnosti često povezani sa mnogim drugim čvorovima i mogu igrati centralnu ulogu u mreži.

Centralnost stepena u *R*-u računa se pomoću funkcije *degree()* u okviru paketa igraph. primenimo je na već započeti primer sa bazama *Actors* i *Movies*:

```
degree(actorNetwork, mode="all")
```

```
> degree(actorNetwork, mode="all")
   Tom Hanks  Gary Sinise Robin Wright  Bill Paxton  Kevin Bacon
           6           6            2           4            4
   Ed Harris Sean Connery Nicolas Cage
           6           2            2
>
```

Слика 18: Centralnost stepena čvorova koji predstavljaju glumce

3.1.1 Normalizacija vrednosti centralnosti stepena

Neobrađene vrednosti centralnosti stepena za čvor možda nisu uvek direktno uporedive, jer zavise od veličine i strukture mreže. Na primer, čvor sa 10 veza u maloj mreži može biti uticajniji od čvora sa 100 veza u velikoj mreži. Da bi vrednosti centralnosti stepena bile uporedive u različitim mrežama, one se mogu normalizovati.

Postoji nekoliko načina da se normalizuju vrednosti centralnosti stepena. Jedan uobičajeni pristup je da se sirova vrednost centralnosti stepena za čvor podeli sa maksimalnom mogucom vrednošcu centralnosti stepena za taj čvor, s obzirom na veličinu i strukturu mreže. Na primer, u neusmerenom grafu, maksimalna vrednost centralnosti stepena za čvor je jednostavno ukupan broj čvorova u grafu minus 1, jer svaki čvor može imati najviše jednu granu do svakog drugog čvora osim samog sebe.

Drugi pristup je da se sirova vrednost centralnosti stepena za čvor podeli sa prosečnom vrednošcu centralnosti stepena za sve čvorove u mreži. Ovo može dati preciznije poređenje relativnog uticaja čvorova, jer uzima u obzir ukupnu gustinu mreže.

Važno je izabrati odgovarajuci metod normalizacije u zavisnosti od ciljeva analize i karakteristika mreže. Normalizovane vrednosti centralnosti stepena mogu se lakše uporediti u različitim mrežama i mogu pružiti precizniju sliku o uticaju čvorova unutar mreže.

3.2 Centralnost između

Centralnost između je mera do koje se čvor nalazi na najkracem putu između ostalih čvorova u mreži. To je način kvantifikacije važnosti ili uticaja čvora na osnovu njegove sposobnosti da poveže druge čvorove u mreži.

Da bismo izračunali centralnost između čvora, prvo identifikujemo sve parove čvorova u mreži i izbrojimo broj najkracih putanja između svakog para koji prolaze kroz čvor. Centralnost između čvora se tada izračunava kao zbir dela svih najkracih puteva koji prolaze kroz čvor za svaki par čvorova.

Na primer, razmotrite sledeći graf sa četiri čvora i pet veza:

	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	0	1	0	1
4	1	1	1	0

Na ovom grafu postoji šest parova čvorova: $(1, 2)$, $(1, 3)$, $(1, 4)$, $(2, 3)$, $(2, 4)$ i $(3, 4)$. Najkraci putevi između ovih parova su:

$$
\begin{array}{lcl}
(1,2) & : & 1-2 \\
(1,3) & : & 1-4-3 \\
(1,4) & : & 1-4 \\
(2,3) & : & 2-3 \\
(2,4) & : & 2-4 \\
(3,4) & : & 3-4
\end{array}
$$

Centralnost između čvora 1 se izračunava na sledeći način:

- Za par $(1,2)$, $1/1 = 1$, dakle jedan od najkraćih puteva prolazi kroz čvor 1.
- Za par$(1,3)$, $1/1 = 1$, dakle jedan od najkraćih puteva prolazi kroz čvor 1.
- Za par $(1,4)$, $1/1 = 1$, dakle jedan od najkraćih puteva prolazi kroz čvor 1.
- Za par $(2,4)$,$0/1 = 0$, dakle nijedan od najkraćih puteva ne prolazi kroz čvor 1.
- Za par $(2,4)$,$0/1 = 0$, dakle nijedan od najkraćih puteva ne prolazi kroz čvor 1.
- Za par $(3,4)$, $0/1 = 0$, dakle nijedan od najkraćih puteva ne prolazi kroz čvor 1.

Dakle, centralnost između čvora 1 je:

$$\frac{(1+1+1+0+0+0)}{6} = \frac{1}{2} \tag{3}$$

Sa matematičke tačke gledišta, centralnost između čvora V u grafu G može se predstaviti kao:

$$C_B(V) = \sum_{s \neq t \neq V} \frac{\sigma(s,t/V)}{\sigma(s,t)} \tag{4}$$

gde je $C_B(V)$ centralnost između čvora V, $\sigma(s,t)$ je broj najkracih puteva između čvorova s i t, a $\sigma(s,t/V)$ je broj najkracih puteva između s i t koji prolaze kroz V. Kombinacije se prave po svim parovima temena s i t u grafu G.

Čvorovi sa visokom centralnošću između su oni koji povezuju mnoge parove čvorova i stoga su važni za komunikaciju unutar mreže. Oni takođe mogu igrati centralnu ulogu u mreži i imati vecći uticaj na njenu ukupnu strukturu.

Ispitajmo sada centralnost između na primeru sa bazama *Actors* i *Movies*, koristeći funkciju *betweenness()* u okviru paketa igraph:

```
betweenness(actorNetwork, directed=F, weights=NA, normalized = T)
```

```
> betweenness(actorNetwork, directed=F, weights=NA, normalized = T)
   Tom Hanks  Gary Sinise Robin Wright  Bill Paxton  Kevin Bacon
   0.1190476    0.1190476    0.0000000    0.0000000    0.0000000
   Ed Harris Sean Connery Nicolas Cage
   0.4761905    0.0000000    0.0000000
>
```

Слика 19: Centralnost između čvorova koji predstavljaju glumce

3.3 Centralnost bliskosti

U teoriji grafova, centralnost bliskosti je mera relativne važnosti temena (ili čvora) u grafu. Izračunava se kao recipročna vrednost zbira rastojanja najkraćih putanja od nekog temena do svih ostalih temena u grafu. U suštini, meri koliko je neko teme "blizu"svim drugim temenima u grafu, pri čemu temena koja su centralnija imaju niži zbir rastojanja do drugih temena.

Centralnost bliskosti se može izračunati na sledeći način:

1. Za dati čvor V u grafu G, potrebno je pronaći najkraću udaljenost između V i svih ostalih čvorova u grafu.
2. Sabrati rastojanja od V do svih ostalih čvorova u grafu.
3. Uzeti recipročnu vrednost ove sume da bi se dobila centralnost bliskosti čvora V.

Matematički, centralnost bliskosti čvora V u grafu G može se predstaviti kao:

$$C_c(V) = \frac{1}{\sum d(V,W)} \tag{5}$$

gde je $C(V)$ centralnost blizine čvora V, $d(V,W)$ najkraća udaljenost između V i W, gde W predstavlja sve ostale čvorove u grafu G.

Centralnost bliskosti se često koristi za identifikaciju ključnih čvorova u mreži koji mogu efikasno da dosegnu i komuniciraju sa mnogim drugim čvorovima. Posebno je koristan za identifikaciju čvorova koji igraju centralnu ulogu u širenju informacija ili resursa kroz mrežu.

Pored upotrebe u identifikaciji centralnih čvorova, centralnost bliskosti se takođe može koristiti za upoređivanje relativne centralnosti različitih čvorova u grafu. Na primer, ako želite da uporedite centralnost dva čvora u društvenoj mreži, možete izračunati centralnost njihove bliskosti i uporediti rezultate. Čvor sa većom centralnošću bliskosti bi se smatrao centralnijim i potencijalno uticajnijim u mreži.

Valja imati na umu da definicija centralnosti bliskosti zavisi od pretpostavke da se koriste najkraće udaljenosti između čvorova. U nekim slučajevima može biti prikladnije koristiti druge mere udaljenosti, kao što su euklidsko rastojanje ili Taksi geometrija, umesto udaljenosti najkraće putanje.

Sada možemo ispitati centralnost bliskosti na našem primeru korišćenjem funkcije *closeness()*:

```
closeness(actorNetwork, mode="all", weights=NA, normalized=T)
```

```
> closeness(actorNetwork, mode="all", weights=NA, normalized=T)
   Tom Hanks  Gary Sinise Robin Wright  Bill Paxton  Kevin Bacon
   0.7777778    0.7777778    0.5000000    0.7000000    0.7000000
   Ed Harris Sean Connery Nicolas Cage
   0.8750000    0.5384615    0.5384615
>
```

Слика 20: Centralnost bliskosti čvorova koji predstavljaju glumce

3.4 Centralnost sopstvenog vektora

Centralnost sopstvenog vektora je mera uticaja čvora u mreži, zasnovana na uticaju njegovih suseda. Izračunava se kao sopstveni vektor matrice povezanosti grafa G, koja predstavlja odnose između čvorova.

3.4.1 Matrica povezanosti

Matrica povezanosti je matrična reprezentacija konačnog grafa. Ima redove i kolone koji odgovaraju čvorovima na grafu, a unos u i-tom redu i j-toj koloni je 1 ako postoji veza između čvora i i čvora j, i 0 u suprotnom.

Na primer, razmotrimo sledeci jednostavan graf sa četiri čvora i pet veza:

	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	0	1	0	1
4	1	1	1	0

Na ovom grafu postoji veza između čvorova 1 i 2, čvorova 1 i 4, čvorova 2 i 3, čvorova 2 i 4 i čvorova 3 i 4. Matrica koja se nalazi iznad je matrični prikaz grafa i naziva se matrica povezanosti.

Matrice povezanosti su korisne za predstavljanje i analizu grafova, jer obezbeđuju jednostavan i kompaktan način za predstavljanje veza između čvorova. Takođe su korisni za skladištenje grafova u računaru, jer se njima može lako manipulisati i analizirati pomocu matričnih operacija.

Postoji nekoliko varijacija matrica povezanosti, u zavisnosti od tipa grafa koji se predstavlja. Na primer, neusmereni graf se može predstaviti korišcenjem matrice simetrične poveznosti, gde je unos u i-tom redu i j-toj koloni isti kao unos u j-tom redu i i-toj koloni. Usmereni graf se može predstaviti korišcenjem asimetrične matrice povezanosti, gde unosi u matricu zavise od smera veze.

3.4.2 Sopstveni vektori

Centralnost sopstvenog vektora čvora je unos u sopstvenom vektoru koji odgovara tom čvoru. Sopstveni vektor je poseban tip vektora koji zadovoljava određenu jednačinu koja uključuje matricu povezanosti mreže. Može se izračunati korišcenjem tehnika linearne algebre, kao što je iteracija ili Jacobi metoda.

Da bismo matematički razumeli centralnost sopstvenih vektora, korisno je razmotriti jednačinu sopstvenih vrednosti za matricu povezanosti A grafa:

$$A \cdot v = \lambda \cdot v \tag{6}$$

Ovde je v sopstveni vektor matrice A, a λ je odgovarajuća sopstvena vrednost. Jednačina kaže da kada se matrica A pomnoži sa vektorom v, rezultat je skalarni višekratnik (broj koji je djeljiv s tim brojem) vektora v. Skalarni višekratnik je sopstvena vrednost λ.

Sopstveni vektori matrice su vektori različiti od nule koji zadovoljavaju ovu jednačinu. Oni se mogu smatrati „pravcima“ u kojima matrica rasteže ili skuplja vektor kada se pomnoži sa matricom. Sopstvene vrednosti su skalari koji određuju količinu rastezanja ili skupljanja koja se dešava u svakom pravcu.

Sopstveni vektor daje svakom čvoru rezultat proporcionalan zbiru rezultata svih njegovih suseda. Ukoliko posmatramo matricu povezanosti A i v kao njen sopstveni vektor, važno je napomenuti da množenje sa A ne menja pravac u kome v pokazuje.

U kontekstu centralnosti sopstvenog vektora, sopstveni vektor v predstavlja uticaj čvorova u mreži, a unosi u sopstveni vektor su proporcionalni uticaju odgovarajucih čvorova. Sopstvena vrednost λ određuje ukupnu snagu uticaja.

3.4.3 Važnost i upotreba metode

Da bismo razumeli centralnost sopstvenog vektora, korisno je razmišljati o uticaju čvora kao o toku „uticaja“ ili „važnosti“ koji se širi od tog čvora do njegovih suseda, a odatle do njihovih suseda, i tako dalje. Čvorovi koji su povezani sa drugim uticajnim čvorovima ce dobiti vecći protok uticaja, i stoga ce imati vecću centralnost sopstvenog vektora.

Centralnost sopstvenog vektora se često koristi za identifikaciju najuticajnijih čvorova u mreži, kao što su ključni igrači u društvenoj mreži ili najvažnije stranice na World Wide Web-u. Takođe je korisno za identifikaciju strukture mreže, pošto su čvorovi sa visokom centralnošću sopstvenog vektora često povezani sa mnogim drugim uticajnim čvorovima i mogu igrati centralnu ulogu u mreži.

Postoji nekoliko razloga zašto se centralnost sopstvenih vektora često smatra boljom merom centralnosti od nekih drugih mera.

Jedan od razloga je taj što centralnost sopstvenog vektora uzima u obzir uticaj suseda čvora, a ne samo broj veza koje čvor ima. Ovo može biti preciznije u situacijama kada uticaj čvora nije direktno proporcionalan njegovom broju veza. Na primer, čvor može imati mnogo veza, ali i dalje ima nisku centralnost sopstvenog vektora ako njegovi susedi nisu mnogo uticajni.

Drugi razlog je taj što je centralnost sopstvenih vektora osetljivija na ukupnu strukturu mreže. Uzima u obzir odnose između čvorova i tok uticaja unutar mreže, a ne samo direktne veze između čvorova. Ovo može biti korisno za identifikaciju ključnih igrača ili važnih čvorova u mreži, kao i za razumevanje ukupne strukture mreže.

Važno je napomenuti da je centralnost sopstvenih vektora samo jedan od načina merenja centralnosti čvora u mreži, i možda nije uvek najpogodnija mera u zavisnosti od ciljeva analize i karakteristika mreže.

Izračunajmo i centralnost sopstvenih vektora na našem primeru. Izračunavanje centralnosti sopstvenih vektora je iterativni proces i može proizvesti jako veliku količinu informacija. Većinu korisnika će zanimati samo rezultati centralnosti, te prilikom ovog računanja treba reći igraph-u da nas zanima samo vektor rezultata centralnosti koje računamo.

```
Eig <- evcent(actorNetwork)$vector
Eig
```

```
> Eig <- evcent(actorNetwork)$vector
> Eig
   Tom Hanks  Gary Sinise Robin Wright  Bill Paxton  Kevin Bacon    Ed Harris
   1.0000000    1.0000000    0.4204341    0.7532679    0.7532679    0.8300186
Sean Connery Nicolas Cage
   0.2209266    0.2209266
>
```

Слика 21: Centralnost sopstvenih vektora čvorova koji predstavljaju glumce

Sada ćemo ilustrovati tabelom kada je najbolje koristiti koji metod centralnosti:

Centrality	Use case
Degree Centrality	Allows identification of individuals who are well connected, popular, or allows connection to a wider network
Betweenness Centrality	Allows the identification of individuals that controls the flow in a network. E.g., when accessing emails, it's easier to identify if some one conducts the communication for a client
Closeness Centrality	Helps identify individuals best placed to influence the network. E.g., If we have to communicate a social event, individual best placed in the network can be targeted for faster communication or spread of information
Eigenvector Centrality	Understanding human social network

Слика 22: Tabelarni prikaz koji ilustruje kada je najbolje koristiti koji metod centralnosti, zasnovan na primerima iz stvarnog života

3.5 Vizualizacija važnih čvorova u grafu

Postoji nekoliko načina da se vizualizuju važni čvorovi na grafu, u zavisnosti od tipa grafa i mere centralnosti koja se koristi.

Jedan uobičajeni pristup je korišćenje dijagrama veze čvorova, gde su čvorovi predstavljeni krugovima, a veze su predstavljene linijama koje povezuju čvorove. Veličina čvorova može biti proporcionalna njihovoj centralnosti, pri čemu veći čvorovi ukazuju na uticajnije čvorove. Boja čvorova se takođe može koristiti za kodiranje centralnosti, sa skalom boja koja preslikava niske vrednosti centralnosti na jedan kraj spektra i visoke vrednosti centralnosti na drugi kraj.

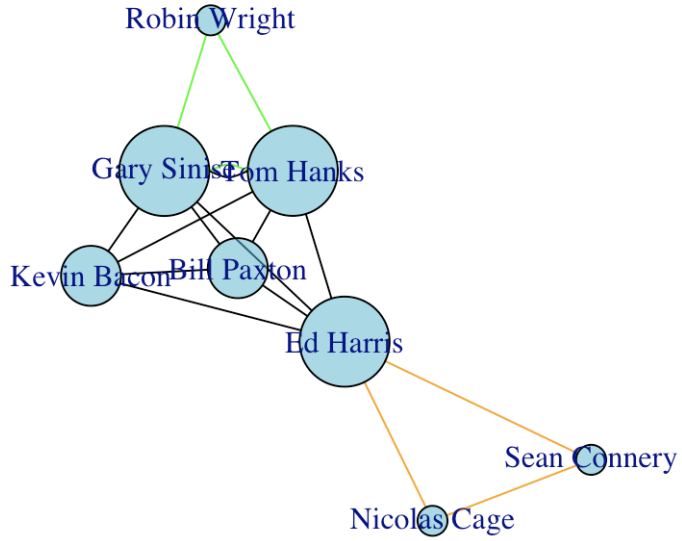
Drugi pristup je korišćenje matričnog dijagrama, gde su čvorovi predstavljeni redovima i kolonama u matrici, a unosi u matrici ukazuju na prisustvo ili odsustvo veze između čvorova. Boja unosa se može koristiti za kodiranje centralnosti, sa skalom boja koja preslikava niske vrednosti centralnosti na jedan kraj spektra i visoke vrednosti centralnosti na drugi kraj.

Druga opcija je korišćenje rasporeda usmerenog na silu, gde su čvorovi pozicionirani na osnovu privlačnih i odbojnih sila između njih. Čvorovi mogu biti veličine i obojeni na osnovu njihove centralnosti, a linije mogu biti deblje ili vidljivije kako bi se ukazale na jače veze.

Postoji mnogo drugih načina da se vizualizuju važni čvorovi na grafu, a najbolji pristup ce zavisiti od specifičnih ciljeva analize i karakteristika grafa. Važno je odabrati metod vizuelizacije koji jasno komunicira centralnost čvorova i strukturu grafa.

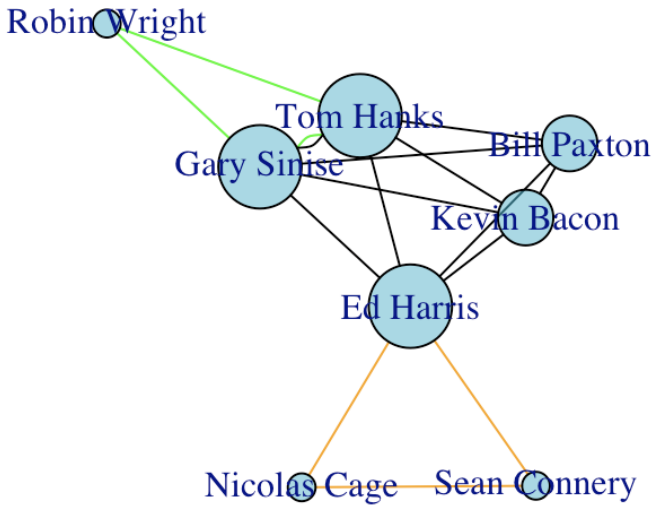
Pogledajmo sada kako bi izgledao dijagram veze čvorova gde je veličina čvora proporcionalna njegovoj centralnosti stepena i centralnosti između:

```
deg <- degree(actorNetwork, mode="all")
plot(actorNetwork, vertex.size=deg*6, vertex.color=rgb(0.1,0.7,0.8,0.5) )
```



Слика 23: Dijagram veze čvorova gde je veliličina čvora proporcionalna broju čvorova sa kojim je povezan

```
bet <- betweenness(actorNetwork, directed=F, weights=NA, normalized = T)
plot(actorNetwork, vertex.size=deg*6, vertex.color=rgb(0.1,0.7,0.8,0.5) )
```



Слика 24: Dijagram veze čvorova gde je veliličina čvora proporcionalna njegovoj centralnosti između

4 Analiza klastera

4.1 Definicija i osnovni pojmovi

Clustering, tj. analiza klastera, na našem jeziku grupisanje, je metoda nenadgledanog učenja u mašinskom učenju i *data mining-u*, tj. rudarenju podataka koja ima za cilj da podeli skup tačaka podataka u klastere ili grupe. U kontekstu *clustering-a* , klaster je grupa tačaka podataka koje su sličnije jedna drugoj nego što su tačke podataka u drugim klasterima. Algoritmi za grupisanje grupišu tačke podataka u klastere na osnovu obrazaca i odnosa unutar podataka.

Na primer, ako imate skup podataka o klijentima, klaster bi mogao da grupiše kupce koji imaju slične karakteristike, kao što su starost, lokacija i kupovne navike. Grupisanje se može koristiti za identifikaciju obrazaca ili trendova u podacima, ili za segmentiranje podataka u grupe radi dalje analize.

U analizi društvenih mreža, grupisanje se odnosi na sklonost pojedinaca da formiraju grupe ili klastere unutar veće mreže. Ovi klasteri mogu biti zasnovani na različitim faktorima, kao što su zajednički interesi, geografska lokacija ili društvene veze.

Grupisanje je važan aspekt društvenih mreža i može imati značajne implikacije na širenje informacija i formiranje društvenih normi. Na primer, ako pojedinci unutar društvene mreže imaju tendenciju da formiraju tesno povezane klastere, veća je verovatnoća da ce se informacije širiti unutar tih grupa, a ne po celoj mreži. Slično tome, ako pojedinci unutar klastera imaju slične stavove ili uverenja, veća je verovatnoća da ce ti stavovi uticati na ponašanje drugih u klasteru.

Koncept grupisanja je neizostavan deo analize društvenih mreža i može pružiti vredan uvid u strukturu i dinamiku društvenih mreža.

Klasteri se mogu vizualizovati kao grupe tačaka (podataka) u *scatterplot-u*, ili drugom vizuelnom prikazu podataka. Klasteri su obično razdvojeni regionima niže gustine, a granice između klastera često nisu dobro definisane. To je zato što je cilj grupisanja (*clustering-a*) da se pronađu prirodni obrasci u podacima, a ne da se prave precizna predviđanja ili klasifikacije.

4.1.1 Merenje grupisanja u klaster analizi

Postoji nekoliko razloga zašto je važno meriti grupisanje u klaster analizi:

- Za procenu kvaliteta klastera - Merenje grupisanja može pomoci da se utvrdi koliko su dobro tačke podataka grupisane u klastere i da li se klasteri razlikuju jedan od drugog. Ovo je važno jer je cilj klaster analize da pronađe smislene obrasce u podacima.
- Upoređivanje različitih algoritama ili konfiguracije za grupisanje - Merenjem grupisanja, moguce je uporediti performanse različitih algoritama ili konfiguracija klasterovanja i izabrati onaj koji najbolje funkcioniše za dati skup podataka.
- Upoređivanje rezultata klaster analize sa drugim tehnikama - Merenje grupisanja može pomoci da se uporede rezultati klaster analize sa drugim tehnikama, kao što su klasifikacija ili regresija, i odredi koji pristup je najprikladniji za dati problem.

Postoji nekoliko različitih načina za merenje grupisanja u grafu ili društvenoj mreži. Neke uobičajene metrike za merenje grupisanja uključuju:

- Koeficijent grupisanja - metrika koja meri stepen do kojeg su susedni čvorovi takođe povezani jedni sa drugima. Može da se kreće od 0 do 1, sa višim vrednostima koje ukazuju na viši nivo grupisanja unutar mreže.
- Prosečna dužina najkraće putanje unutar klastera - metrika koja meri prosečnu udaljenost između parova čvorova unutar klastera. Manja prosečna dužina najkraće putanje ukazuje na viši nivo grupisanja.
- Odnos veza unutar klastera prema ukupnim vezama - metrika koja meri proporciju veza unutar klastera prema ukupnom broju ivica u klasteru. Veći odnos ukazuje na viši nivo grupisanja.
- Modularnost - metrika koja meri stepen do kojeg graf može da se podeli na različite klastere ili zajednice. Veća vrednost modularnosti ukazuje na viši nivo grupisanja.

4.2 Tipovi algoritama u analizi klastera

Postoji mnogo različitih algoritama koji se mogu koristiti za klaster analizu, a izbor algoritma zavisi od specifičnih karakteristika podataka i ciljeva analize. Evo nekoliko primera najčešće korišcenih algoritama za analizu klastera:

- Grupisanje K-srednjih vrednosti - Ovo je popularan algoritam koji deli podatke na određeni broj klastera (k) iterativno dodeljivanjem svake tačke podataka grupi sa najbližom srednjom vrednošcu.
- Hijerarhijsko grupisanje - algoritam koji gradi hijerarhiju klastera, pri čemu se svaki klaster deli na manje klastere dok svaka tačka podataka ne bude u svom klasteru. Postoje dva glavna tipa hijerarhijskog grupisanja: aglomerativno (odozdo-nagore) i podela (od vrha prema dole).
- Grupisanje zasnovano na gustini - algoritam koji definiše klastere kao oblasti visoke gustine okružene oblastima niske gustine. Jedan popularan algoritam za grupisanje zasnovan na gustini je DBSCAN (Prostorno grupisanje aplikacija sa bukom zasnovano na gustini).
- Grupisanje sa maksimizacijom očekivanja (EM) - algoritam koji se koristi za podatke koji imaju mešavinu različitih distribucija. Počinje pogađanjem parametara distribucija, a zatim ih iterativno precizira sve dok podaci nisu dobro modelirani.
- Proširivanje afiniteta - algoritam koji je zasnovan na konceptu „prenošenja poruke“ između tačaka podataka. Počinje dodeljivanjem svakoj tački podataka kao sopstvenom klasteru, a zatim iterativno prilagođava preferencije tačaka podataka sve dok se ne postigne konvergencija.
- Spektralno grupisanje - algoritam koji koristi tehnike iz linearne algebre da transformiše podatke u prostor niže dimenzije, a zatim primenjuje k-means klasterisanje na transformisane podatke.

4.3 Proces primene analize klastera na podacima

Klaster analiza je mocan alat za istraživanje i razumevanje obrazaca u podacima i može se primeniti na širok spektar tipova podataka i problematičnih domena.

Koraci za korišcenje analize klastera:

1. Predobrada podataka - Ovo može uključivati čišcenje podataka, uklanjanje odstupanja i normalizaciju varijabli.

2. Biranje algoritam za grupisanje - Postoji mnogo različitih algoritama koji se mogu koristiti za analizu klastera, a izbor algoritma zavisi od specifičnih karakteristika podataka i ciljeva analize.

3. Određivanje broja klastera - Neki algoritmi za grupisanje, kao što su k-means, zahtevaju od korisnika da unapred odredi broj klastera. Drugi algoritmi, kao što je hijerarhijsko grupisanje, ne zahtevaju ovu specifikaciju.

4. Pokretanje algoritam grupisanja - Izabrani algoritam se primenjuje na podatke, a tačke podataka se grupišu u klastere.

5. Procena grupisanja - Važno je izmeriti kvalitet grupisanja da biste utvrdili koliko su dobro tačke podataka grupisane u klastere i da li se klasteri razlikuju jedan od drugog.

6. Vizuelizacija rezultata - Vizuelizacija klastera može pomoci u razumevanju obrazaca u podacima i sticanju uvida u odnose između tačaka podataka.

7. Interpretacija rezultata - Nakon što su klasteri identifikovani, važno je interpretirati rezultate i razumeti šta oni znače u kontekstu problema koji se proučava.

5 Spektralno grupisanje

Jedan od algoritama za grupisanje podataka je *Spectral clustering*, tj. spektralno grupisanje, tehnika koja se koristi za particionisanje grafa na klastere ili grupe čvorova na osnovu karakteristika sopstvenih vektora grafa. Uključuje konstruisanje matrice sličnosti za graf, koji kodira jačinu veza između čvorova, a zatim korišćenje tehnika iz linearne algebre za pronalaženje sopstvenih vektora ove matrice. Sopstveni vektori se zatim mogu koristiti za particionisanje grafa na klastere, pri čemu se čvorovi koji su bliže povezani stavljaju u isti klaster.

Spektralno grupisanje se često koristi u analizi društvenih mreža da bi se identifikovale grupe pojedinaca ili organizacija koje su usko povezane jedna sa drugom. Može biti korisno u identifikaciji zajednica unutar veće mreže i može pomoći u otkrivanju obrazaca odnosa koji možda nisu odmah očigledni iz sirovih podataka.

Postoje mnogi drugi alati i tehnike koje se koriste u analizi društvenih mreža, uključujući mere centralnosti, vizuelizaciju mreže i teoriju grafova. Proučavajući obrasce odnosa unutar društvenih mreža, istraživači mogu steći uvid u dinamiku društvenih sistema i načine na koje se oni menjaju tokom vremena.

5.1 Matematički prikaz spektralnog grupisanja

Spektralno grupisanje je tehnika za grupisanje tačaka podataka koja koristi tehnike iz linearne algebre za transformaciju podataka u prostor niže dimenzije. Često se koristi za podatke koji nisu linearno odvojivi ili imaju neujednačenu distribuciju.

Matematički, spektralno klasterisanje se može opisati kao problem optimizacije. Cilj je pronaci klastere koji minimiziraju zbir kvadrata rastojanja između tačaka podataka i njihovog najbližeg težišta. Dakle, potreban nam je minimum funkcije:

$$\sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{7}$$

gde su C_1, C_2, ..., C_k klasteri,(μ_1,μ_2, ..., μ_k) težišta, a k je tačka podataka.

U analizi društvenih mreža, spektralno grupisanje se može koristiti za identifikaciju zajednica ili grupa pojedinaca unutar mreže koje su bliže međusobno povezane nego sa ostatkom mreže. Ove grupe mogu predstavljati različite društvene krugove, interesne grupe ili zajednice prakse.

Pre nego što detaljnije objasnimo spektralno grupisanje i njegov algoritam, potrebno je da se upoznamo sa načinom implementacije ovog algoritma i pojmovima čiju primenu ovaj algoritam koristi.

5.2 Implementacija spektralnog gupisanja

Algoritam spektralnog grupisanja je iterativni proces koji se sastoji od sledećih koraka:

1. Računanje matrica afiniteta: Matrica afiniteta je mera sličnosti parova tačaka podataka. Može se izračunati korišćenjem raznih metoda, kao što su metod Gausovog jezgra ili algoritam k-najbližih suseda.

2. Računanje Laplasove matrice: Laplasova matrica je transformacija matrice afiniteta koja predstavlja kodiranje povezivosti među tačkama podataka. Može se izračunati na različite načine, poput normalizovanog i nenormalizovanog Laplasovog metoda.

3. Računanje sopstvenih vektora Laplasove matrice: Sopstveni vektori Laplasove matrice prikazuju podatke u prostoru niže dimenzije. Pomoću njih se formira skup podataka koji se dalje može grupisati korišćenjem k-means klasteringa ili nekog drugog algoritma grupisanja.

4. Grupisanje dobijenih podatka: Dobijeni podaci se grupišu koristeći k-means ili nekog drugi algrotiam grupisanja.

5.3 Matrica afiniteta

Matrica afiniteta je kvadratna matrica koja se koristi za predstavljanje jačine odnosa ili veza između različitih entiteta. Matrica afiniteta ima dimenzije jednake broju entiteta koji se porede. Entiteti mogu biti bilo šta što se može uporediti ili povezati jedni sa drugima, kao što su ljudi, proizvodi ili koncepti. Svaki element u matrici predstavlja snagu odnosa između dva entiteta povezana sa redom i kolonom tog elementa. Vrednosti u matrici mogu biti ili kvantitativne ili kvalitativne, u zavisnosti od prirode odnosa koji se predstavljaju.

Na primer, ako su entiteti koji se porede ljudi, matrica afiniteta može predstavljati snagu odnosa između svakog para ljudi. Vrednosti u matrici mogu biti zasnovane na faktorima kao što su učestalost komunikacije, količina vremena provedenog zajedno ili stepen emocionalne povezanosti.

U nekim slučajevima, matrica afiniteta se koristi za predstavljanje sličnosti između različitih entiteta. Na primer, u sistemima za pronalaženje informacija, matrica afiniteta može da se koristi za predstavljanje sličnosti između različitih dokumenata ili ključnih reči. U ovom slučaju, vrednosti u matrici bi predstavljale stepen preklapanja ili sličnosti između entiteta koji se porede.

Matrica afiniteta se može koristiti za obavljanje različitih matematičkih operacija, kao što su množenje matrice, transpozicija i inverzija. Ove operacije se mogu koristiti za analizu i manipulisanje odnosima predstavljenim u matrici, kao i za izdvajanje uvida i obrazaca iz podataka.

Matrica afiniteta se može koristiti za identifikaciju klastera ili grupa entiteta koji su jako povezani ili povezani jedan sa drugim. Takođe se može koristiti za izračunavanje udaljenosti ili sličnosti između različitih entiteta, na osnovu vrednosti u matrici.

Najčešće se matrica afiniteta konstruiše korišćenjem neke mere sličnosti, poput Euklidskog rastojanja ili kosinusne sličnosti, koje upoređuju određene karakteristike tačaka podataka i dodeljuju nivo sličnosti na osnovu stepena preklapanja.

U suštini, matrica afiniteta je od krucijalne važnosti za algoritam spektralnog grupisanja jer nam pruža meru sličnosti tačaka podataka koja se dalje koristi za identifikovanje klastera. Bez matrice afiniteta, primena spektralnog grupisanja na podatke uopšte ne bi bila moguća.

5.3.1 Izračunavanje matrice afiniteta korišćenjem *The k-nearest neighbor (KNN)* algoritma

Algoritam k-najbližeg suseda (KNN) je algoritam mašinskog učenja koji se može koristiti za klasifikaciju podataka na osnovu njihove sličnosti sa drugim tačkama podataka. Funkcioniše tako što identifikuje k tačaka podataka koje su najsličnije datoj tački podataka i koristi te tačke podataka za predviđanje klase ili oznake date tačke podataka.

Jedan od načina da se KNN algoritam koristi sa matricom afiniteta je da se redovi matrice tretiraju kao tačke podataka, a vrednosti u matrici kao karakteristike ili atributi. KNN algoritam se zatim može koristiti za klasifikaciju svake tačke podataka (reda) na osnovu njene sličnosti sa drugim tačkama podataka, kao što je određeno vrednostima u matrici afiniteta.

Na primer, pretpostavimo da matrica afiniteta predstavlja sličnost između različitih dokumenata na osnovu reči koje sadrže. KNN algoritam bi se mogao koristiti za klasifikaciju svakog dokumenta kao pripadnika određene teme ili kategorije na osnovu sličnosti reči koje sadrži sa rečima u drugim dokumentima.

Korišćenje KNN algoritam sa matricom afiniteta, obuhvata zadavanje vrednosti k (broj najbližih suseda koje treba uzeti u obzir) i mere udaljenosti koju cete koristiti za poređenje tačaka podataka. Nakon zadavanja početnih uslova, sledi izračuvanje rastojanja između svakog para tačaka podataka korišćenjem matrice afiniteta i izabrane meru udaljenosti. Konačno, KNN algoritam se koristi da klasifikuje svaku tačku podataka na osnovu udaljenosti do njenih k najbližih suseda.

Da bi se izračunala matrica afiniteta korišćenjem KNN algoritma, koristiće se sledeći koraci:

1. Definisati tačke podataka koje ce biti uključene u matricu afiniteta. To može biti bilo koji tip entiteta koji se može porediti ili povezati jedan sa drugim, kao što su ljudi, proizvodi

ili koncepti. Skup podataka treba da bude predstavljen kao skup od n tačaka podataka, gde je svaka tačka podataka d-dimenzionalni vektor vrednosti ili atributa obeležja.

2. Prikupiti podatke ili informacije o karakteristikama ili atributima tačaka podataka. Ovo može uključivati prikupljanje podataka iz anketa, intervjua ili drugih izvora.

3. Izabrati meru udaljenosti koja ce se koristiti za izračunavanje sličnosti između tačaka podataka. Neke uobičajene mere udaljenosti koje se koriste sa KNN algoritmom uključuju Euklidsko rastojanje, kosinusnu sličnost i rastojanje na Menhetnu.

4. Napraviti praznu matricu afiniteta sa dimenzijama jednakim broju tačaka podataka.

5. Koristiti zadatu meru udaljenosti da bi se izračunala sličnost između svakog para tačaka podataka i popuniti matricu afiniteta sa rezultujucim vrednostima sličnosti.

6. Izabrati vrednost za k, što je broj najbližih suseda koje treba uzeti u obzir prilikom klasifikacije tačke podataka.

7. Koristiti matricu afiniteta i izabranu vrednost k za klasifikaciju svake tačke podataka na osnovu sličnosti njenih karakteristika sa karakteristikama njenih k najbližih suseda.

8. Analizirati rezultate klasifikacije radi procene tačnosti KNN algoritma i izvlačenja uvida i obrazaca iz podataka.

5.3.2 Izračunavanje matrice afiniteta korišćenjem *Gausovog metoda jezdra*

Metod Gausovog jezgra je tehnika koja se može koristiti za izračunavanje matrice afiniteta na osnovu sličnosti između tačaka podataka. Radi korišcenjem Gausove funkcije kernela, koja je tip funkcije kernela koja se široko koristi u mašinskom učenju i analizi podataka.

Ovaj metod funkcioniše tako što koristi Gausovu funkciju jezgra za izračunavanje sličnosti između svakog para tačaka podataka i popunjavanje matrice afiniteta sa rezultujucim vrednostima sličnosti. Matrica afiniteta se zatim može analizirati da bi se iz podataka izdvojili uvidi i obrasci.

Da bi se izračunala matrica afiniteta korišćenjem metoda Gausovog kernela, koristiće se sledeći koraci:

1. Definišu se tačke podataka koje ce biti uključene u matricu afiniteta. To može biti bilo koji tip entiteta koji se može porediti ili povezati jedan sa drugim, kao što su ljudi, proizvodi ili koncepti. Skup podataka treba da bude predstavljen kao skup od n tačaka podataka, gde je svaka tačka podataka d-dimenzionalni vektor vrednosti ili atributa obeležja.

2. Prikupe se podaci ili informacije o karakteristikama ili atributima tačaka podataka. Ovo može uključivati prikupljanje podataka iz anketa, intervjua ili drugih izvora.

3. Definiše se Gausova funkcija kernela, koja je funkcija koja izračunava sličnost između dve tačke podataka na osnovu vrednosti njihovih karakteristika. Gausova funkcija kernela ima parametar koji se zove propusni opseg, koji kontroliše širinu jezgra i određuje stepen glatkoce mere sličnosti.

4. Napravi se prazna matrica afiniteta sa dimenzijama jednakim broju tačaka podataka.

5. Koristi se funkcija Gausovog jezgra za izračunavanje sličnosti između svakog para tačaka podataka i popunjava se matrica afiniteta sa rezultujucim vrednostima sličnosti.

6. Matrica afiniteta se analizira kako bi se izdvojili uvidi i obrasci iz podataka. Ovo može uključivati izvođenje matematičkih operacija na matrici, kao što je množenje matrice ili inverzija.

5.4 Laplasova matrica

Laplasova matrica je matrica koja se koristi za predstavljanje strukture grafa ili mreže. Usko je povezana sa matricom povezanosti, koju smo ranije pominjali kao matricu koja se koristi za predstavljanje strukture grafa.

Laplasova matrica se definiše kao razlika između matrice stepena i matrice susednosti grafa. Matrica stepena je dijagonalna matrica koja sadrži stepen (broj veza) svakog čvora u grafu, a matrica susednosti je matrica koja sadrži veze unutar grafa.

Laplasova matrica se često koristi u teoriji grafova i analizi mreža za predstavljanje strukture i svojstava grafa. Ima niz interesantnih svojstava, kao što je simetričnost i pozitivno poluodređeno, što ga čini korisnim za analizu i manipulisanje grafovima.Laplasova matrica može se koristiti u algoritmima za grupisanje za identifikaciju grupa ili klastera tačaka podataka u skupu podataka.

5.4.1 Nenormalizovana Laplasova matrica

Nenormalizovana Laplasova metoda je tehnika za izračunavanje Laplasove matrice grafa, koja je matrica koja se koristi za predstavljanje strukture grafa. Nenormalizovana Laplasova matrica se dobija oduzimanjem matrice susednosti od matrice stepena grafa.

Izračunavanje Laplasove matrice korišcenjem nenormalizovane Laplasove metode opisano je kroz sledeće korake:

1. Definisati graf koji je potrebno predstaviti Laplasovom matricom. Graf treba predstaviti kao skup temena (čvorova) i ivica (veza) koje povezuju vrhove.

2. Napraviti matricu povezanosti za gorepomenuti graf. Kao što je već poznato, matrica povezanosti je matrica koja sadrži veze grafa. Svaki red i kolona matrice odgovaraju nekom čvoru u grafu, a vrednost na svakoj poziciji (i, j) u matrici je 1 ako postoji veza između čvorova i i j, i 0 ako nema veze.

3. Napraviti matricu stepena za graf. Matrica stepena je dijagonalna matrica koja sadrži stepen (broj veza) svakog čvora u grafu. Svaka pozicija (i, i) u matrici odgovara stepenu (broju veza) čvora i.

4. Izračunati nenormalizovanu Laplasovu matricu oduzimanjem matrice povezanosti od matrice stepena. Nenormalizovana Laplasova matrica je definisana kao:

$$L = D - A \tag{8}$$

5.4.2 Normalizovana Laplasova matrica

Normalizovana Laplasova metoda je tehnika za izračunavanje Laplasove matrice grafa, koja je matrica koja se koristi za predstavljanje strukture grafa. Normalizovana Laplasova matrica se dobija normalizacijom elemenata nenormalizovane Laplasove matrice, koja se izračunava oduzimanjem matrice susednosti od matrice stepena grafa.

Izračunavanje Laplasove matrice korišcenjem normalizovane Laplasove metode opisano je kroz sledeće korake:

1. Izračunati nenormalizovanu Laplasovu matricu korišćenjem goreopisanog metoda.

2. Normalizovati nenormalizovanu Laplasovu matricu tako što ce se svaki element u matrici podeliti kvadratnim korenom stepena odgovarajuceg čvora. Normalizovana Laplasova matrica je definisana kao
$$L' = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \tag{9}$$
gde L' predstavlja normalizovanu Laplasovu matricu, L nenomalizovanu, a $D^{-1/2}$ inverzni koren matrice stepena D.

Normalizovana Laplasova matrica je važan alat za spektralno grupisanje i nju ćemo koristiti jer ima niz svojstava koja je čine dobrom za identifikaciju klastera u skupu podataka. Ona je simetrična i pozitivno poludefinirana i može se izračunati iz matrice afiniteta.

5.5 Sopstveni vektori Laplasove matrice

U spektralnom klasterovanju, sopstveni vektori Laplasove matrice se koriste za identifikaciju klastera u skupu podataka analizom strukture i osobina Laplasove matrice.

Sopstvene vrednosti i sopstveni vektori Laplasove matrice predstavljaju strukturu grafa ili mreže u smislu njegovih čvorova i ivica (veza). Sopstveni vektori su kolone matrice, a vrednosti koje odgovaraju vektorima su odgovarajuce sopstvene vrednosti matrice.

Sopstvene vrednosti i sopstvene vektori koriste se za identifikaciju klastera u podacima. Sopstveni vektori koji odgovaraju najvecim sopstvenim vrednostima su obično najinformativniji i često se koriste za identifikaciju klastera u podacima. Da biste identifikovali klastere, možete da izaberete podskup sopstvenih vektora, obično onih koji odgovaraju k najvecim sopstvenim vrednostima, i da ih koristite da transformišete tačke podataka u prostor niže dimenzije. Transformisane tačke podataka se zatim mogu grupirati korišcenjem bilo koje odgovarajuce metode grupisanja, kao što je klasterisanje k-srednjih vrednosti ili hijerarhijsko grupisanje.

5.6 Grupisanje dobijenih podataka - *K-means clustering*

Jedan od najjednostavnijih i najpopularnijih tipova klasterizacije u nenadgledanom mašinskom učenju je upravo *k-means clustering* clustering ili metod k-sredina. Ovo je iterativni algoritam koji deli neoznačeni skup podatka u k grupa iliti klastera tako da svaka grupa predstavlja skup podataka sa određenim sličnostima.

Ovaj algoritam može se prikazati u nekoliko narednih koraka:

1. Odabir broja k - Ovaj broj predstavlja broj klastera na koji će skup podataka biti podeljen i obično se određuje na osnovu prethodnog znanja o podacima.

2. Fiksiranje k centroida - Centroidi predstavlja imaginarne ili stvarne lokacije koje će biti centri klastera i biraju se na slučajan način.

3. Računanje najbližeg centroida za svaku tačku podataka: Svaka tačka podataka dodeljuje se klasteru čijem je centru najbliža, a za merenje tog rastojanja najčešće se koristi euklidska metrika ili neka od l^p metrika.

4. Računanje centra novodobijenog klastera i premeštanje centroida u njega - nakon što se svi podaci dodele nekom klasteru, nova vrednost centroida svakog od tih klastera treba da bude medijana svih tačaka podataka u njegovom okviru.

Koraci 3 i 4 ponavljaju se dok proces ne iskonvergira, nakon čega se dobijaju finalni klasteri.

5.6.1 Metode odabira broja k - Pravilo lakta

Metoda lakta je zasnovana na ideji da je optimalni broj klastera vrednost k koja minimizira zbir kvadrata rastojanja unutar klastera (VCSS). VCSS je mera kompaktnosti klastera, pri čemu niža vrednost ukazuje na čvršce klastere.

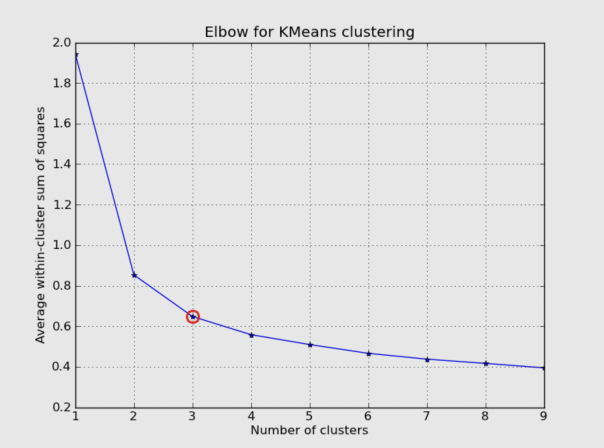
Matematički, VCSS je definisan na sledeci način:

$$WCSS = \sum_{k=1}^{K} \sum_{x \in C_k} (x - \mu_k)^2 \tag{10}$$

gde je x tačka u skupu podataka, μ_k je srednja vrednost tačaka u klasteru k, a zbir se uzima za sve tačke u skupu podataka.

VCSS se može izračunati za svaku vrednost k i nacrtati kao funkcija k, tj. funkcija koja oslikava zavisnost zbira suma u odnosu na k. "Lakat"na dijagramu se tumači kao optimalna vrednost k, jer odgovara tački gde VCSS počinje da se izravnava ili smanjuje sporije.

Vredi napomenuti da je metoda lakta misaona metoda i da izbor k možda nije uvek jasan. U takvim slučajevima,potrebno je koristiti dodatne metode ili razmotriti druge faktore kao što su priroda podataka i svrha grupisanja.



Слика 25: Pravilo lakta

5.6.2 Metode odabira broja k - Analiza silueta

Metoda siluete je zasnovana na ideji da je optimalan broj klastera vrednost k koja daje najveci prosečni koeficijent siluete. Koeficijent siluete je mera koliko dobro je uzorak dodeljen sopstvenom klasteru u poređenju sa drugim klasterima. Formula za računanje koeficijenta siluete je:

$$S_i = \frac{b_i - a_i}{max(a_i, b_i)}$$

.

Gde S_i predstavlja koeficient siluete za datu tačku podataka, a_i prosečno rastojanje između date tačke podataka i ostalih u istom klasteru i b_i prosečno rastojanje između date tačke podataka i svih tačka iz najbližeg klastera.

Koeficijent siluete se krece od -1 do 1, a visok koeficijent siluete ukazuje da je uzorak dobro usklađen sa sopstvenim klasterom.

1. Ako važi $S_i = 1$, to ukazuje na veoma dobru dodelu ispravnom klasteru, tj. da je data tačka podataka jako blizu tačaka iz istog klastera, a daleko od onih iz susednog klastera.

2. Ako važi $S_i = 0$, data tačka podataka se nalazi blizu granice svog klastera.

3. Ako važi $S_i = -1$, data tačka podataka je dodeljena pogrešnom klasteru.

Konačni koeficijent silueta računa se kao prosečni koeficijent silueta svih tačaka podataka. Potom računamo koeficijent silueta za vrednosti k od 2 do N. Što je koeficijent silueta viši, to je podela na klastere bolja.

5.7 Spektralno grupisanje - prednosti i mane

Jedna od prednosti spektralnog grupisanja je što tačke podataka prilikom njegovog implementiranja treba da budu povezane, međutim, ne moraju nužno imati konveksne granice, za razliku od standardnih tehnika grupisanja gde je klasterizacija zasnovana na kompaktnosti podatka.

Neke tehnike grupisanja, poput *k-means clustering*-a, služe se pretpostavkom da su podaci koji pripadaju jednom klasteru sferno raspoređeni oko njegovog centra. Ovo je jaka pretpostavka koja ne mora uvek važiti i u takvim slučajevima metoda spektralnog grupisanje nam može pomoći da formiramo prikladnije klastere. Metoda *k-means clustering*-a će se javiti kao poslednji korak implementacije spektralnog grupisanja, međutim, ne na originalnim podacima, već na prikazu koji čini grubu predstavu povezivosti. Umesto da minimizira kvadratne greške u ulaznom domenu, minimiziraće kvadratne greške sposobnosti rekonstrukcije suseda, što se vrlo često pokazuje boljim.

Jedan od glavnih razloga zašto spektralno grupisanje ne spada u popularne metode i jedna od njegovih glavnih mana je to što je jako sporo (obično uključuje konstrukciju $O(n^2)$ spostvene matrice i pronalaženje sopstvenih vektora koje može imati čak i kubnu vremensku složenost $O(n^3)$), a pored toga i dalje se moramo osloniti na početna rastojanja i sličnosti prilikom formiranja ulaznog grafa pre *embedding*-a. Većinu problema grupisanja čini upravo rukovanje podacima da bismo dobili pouzdane udaljenosti i sličnosti.

Još neke od mana su to što se sa obimom uzorka složenost povećava, a preciznost smanjuje i to što spektralno grupisanje predstavlja računski skupu metodu za velike baze podataka s obzirom na to da se sopstvene vrednosti i vektori moraju računati i da potom treba izvršiti klasterovanje nad njima.

5.8 Primer spektralnog klasterovanja

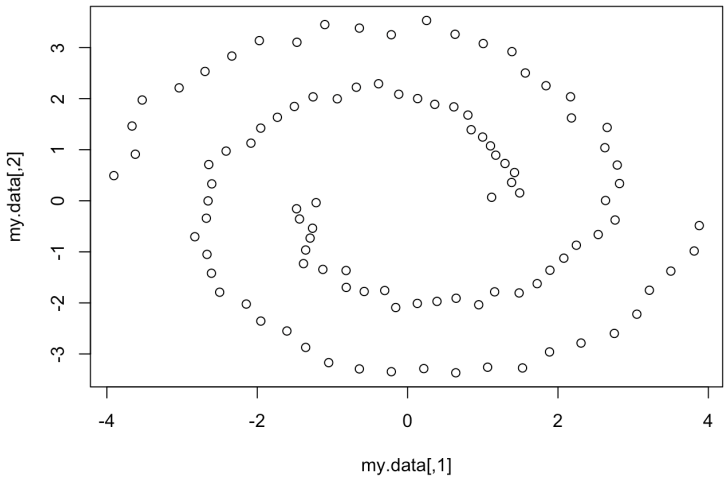
Ilustrujmo sada metod spektralnog klasterovanja na primeru. Baza sa glumcima koju smo ranije koristili je previše mala, te ćemo primer ilustrovati na drugoj bazi.

Koristimo paket *mlbench* koji predstavlja kolekciju problema iz mašinskog učenja veštaki generisanih ili iz stvarnog života.

Učitajmo prvo željenu bazu podataka i predstavimo ih na spiralnom grafiku.

```
install.packages("mlbench")
library(mlbench)

set.seed(111)
obj <- mlbench.spirals(100,1,0.025)
my.data <-  4 * obj$x
plot(my.data)
```



Слика 26: Podaci prikazani na spiralnom grafiku

Za spektralno klasterovanje potrebna nam je matrica sličnosti S. Izračunajmo S za datu bazu podataka koristeći Gausov metod jezgara.

```
s <- function(x1, x2, alpha=1) {
  exp(- alpha * norm(as.matrix(x1-x2), type="F"))
}
```

```r
make.similarity <- function(my.data, similarity) {
  N <- nrow(my.data)
  S <- matrix(rep(NA,N^2), ncol=N)
  for(i in 1:N) {
    for(j in 1:N) {
      S[i,j] <- similarity(my.data[i,], my.data[j,])
    }
  }
  S
}

S <- make.similarity(my.data, s)
S[1:8,1:8]
```

```
           [,1]       [,2]       [,3]       [,4]        [,5]        [,6]
[1,] 1.000000000 0.23071765 0.28038182 0.22306530 0.011854717 0.005865093
[2,] 0.230717649 1.00000000 0.68212910 0.67490015 0.039588037 0.023491599
[3,] 0.280381816 0.68212910 1.00000000 0.79379092 0.039804649 0.020892533
[4,] 0.223065296 0.67490015 0.79379092 1.00000000 0.049864554 0.026286997
[5,] 0.011854717 0.03958804 0.03980465 0.04986455 1.000000000 0.278492316
[6,] 0.005865093 0.02349160 0.02089253 0.02628700 0.278492316 1.000000000
[7,] 0.400350756 0.16584963 0.16663077 0.13394952 0.006707548 0.003945659
[8,] 0.090494161 0.10008191 0.07757167 0.06855753 0.005016359 0.004310321
           [,7]        [,8]
[1,] 0.400350756 0.090494161
[2,] 0.165849632 0.100081907
[3,] 0.166630771 0.077571672
[4,] 0.133949518 0.068557530
[5,] 0.006707548 0.005016359
[6,] 0.003945659 0.004310321
[7,] 1.000000000 0.193956073
[8,] 0.193956073 1.000000000
>
```

Слика 27: Matrica S

Sledeći korak je računanje matrice afiniteta A na osnovu S. Matrica A mora biti pozitivna i simetrična.

Matricom A klasterovanje je zamenjeno problemom particionisanja grafa, gde su povezane komponente grafa interpretirane kao klasteri. Graf se particioniše tako da su ivice koje povezuju različite klastere male težine, a ivice unutar istog klastera velike.

Potom se javlja nam se potreba za dijagonalnom matricom D, gde je svaka vrednost na dijagonali stepen odgovarajućeg čvora, a sve ostale vrednosti 0.

Nakon što smo izračunali D, potrebna nam je Laplasova matrica, koju ćemo naći nenormalizovanom metodom $U = D - A$.

Definišemo pomoćnu funkciju stepenovanja matrice koja računa vrednost matrice M na zadati stepen, gde matrica M mora biti dijagonalna.

S obzirom na to da želimo k klastera, sledeći korak je pronalaženje k minimalnih sopstvenih vektora, ne uključujući trivijalni sopstveni vektor.

Transformaciju opservacije X_i definisana je i-tim redoom u Z. Treba proveiriti da li su tačke lepo razdvojene.

Sada je metodi k sredina lako da pronađe adekvatne klastere.

```
make.affinity <- function(S, n.neighboors=2) {
  N <- length(S[,1])

  if (n.neighboors >= N) {
    A <- S
  } else {
    A <- matrix(rep(0,N^2), ncol=N)
    for(i in 1:N) {
      best.similarities <- sort(S[i,], decreasing=TRUE)[1:n.neighboors]
      for (s in best.similarities) {
        j <- which(S[i,] == s)
        A[i,j] <- S[i,j]
        A[j,i] <- S[i,j]
      }
    }
  }
  A
}

A <- make.affinity(S, 3)
A[1:8,1:8]
```

```
     [,1]      [,2]      [,3]      [,4] [,5] [,6] [,7] [,8]
[1,]    1 0.0000000 0.0000000 0.0000000    0    0    0    0
[2,]    0 1.0000000 0.6821291 0.6749001    0    0    0    0
[3,]    0 0.6821291 1.0000000 0.7937909    0    0    0    0
[4,]    0 0.6749001 0.7937909 1.0000000    0    0    0    0
[5,]    0 0.0000000 0.0000000 0.0000000    1    0    0    0
[6,]    0 0.0000000 0.0000000 0.0000000    0    1    0    0
[7,]    0 0.0000000 0.0000000 0.0000000    0    0    1    0
[8,]    0 0.0000000 0.0000000 0.0000000    0    0    0    1
> 
```

Слика 28: Matrica A

```
D <- diag(apply(A, 1, sum))
D[1:8,1:8]
```

```
         [,1]     [,2]    [,3]     [,4]    [,5]     [,6]    [,7]     [,8]
[1,] 2.437019 0.000000 0.00000 0.000000 0.00000 0.000000 0.00000 0.000000
[2,] 0.000000 2.357029 0.00000 0.000000 0.00000 0.000000 0.00000 0.000000
[3,] 0.000000 0.000000 2.47592 0.000000 0.00000 0.000000 0.00000 0.000000
[4,] 0.000000 0.000000 0.00000 3.288785 0.00000 0.000000 0.00000 0.000000
[5,] 0.000000 0.000000 0.00000 0.000000 2.28715 0.000000 0.00000 0.000000
[6,] 0.000000 0.000000 0.00000 0.000000 0.00000 2.269999 0.00000 0.000000
[7,] 0.000000 0.000000 0.00000 0.000000 0.00000 0.000000 2.47123 0.000000
[8,] 0.000000 0.000000 0.00000 0.000000 0.00000 0.000000 0.00000 2.509627
> 
```

Слика 29: Matrica D

```
U <- D - A
round(U[1:12,1:12],1)
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
 [1,]  1.4  0.0  0.0  0.0  0.0  0.0  0.0  0.0 -0.7   0.0   0.0   0.0
 [2,]  0.0  1.4 -0.7 -0.7  0.0  0.0  0.0  0.0  0.0   0.0   0.0   0.0
 [3,]  0.0 -0.7  1.5 -0.8  0.0  0.0  0.0  0.0  0.0   0.0   0.0   0.0
 [4,]  0.0 -0.7 -0.8  2.3  0.0  0.0  0.0  0.0  0.0  -0.8   0.0   0.0
 [5,]  0.0  0.0  0.0  0.0  1.3  0.0  0.0  0.0  0.0   0.0   0.0   0.0
 [6,]  0.0  0.0  0.0  0.0  0.0  1.3  0.0  0.0  0.0   0.0   0.0   0.0
 [7,]  0.0  0.0  0.0  0.0  0.0  0.0  1.5  0.0  0.0   0.0   0.0   0.0
 [8,]  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.5  0.0   0.0   0.0   0.0
 [9,] -0.7  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.4   0.0   0.0   0.0
[10,]  0.0  0.0  0.0 -0.8  0.0  0.0  0.0  0.0  0.0   1.6  -0.8   0.0
[11,]  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  -0.8   1.6  -0.8
[12,]  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0   0.0  -0.8   1.6
> 
```

Слика 30: Nenormalizovana Laplasova matrica U

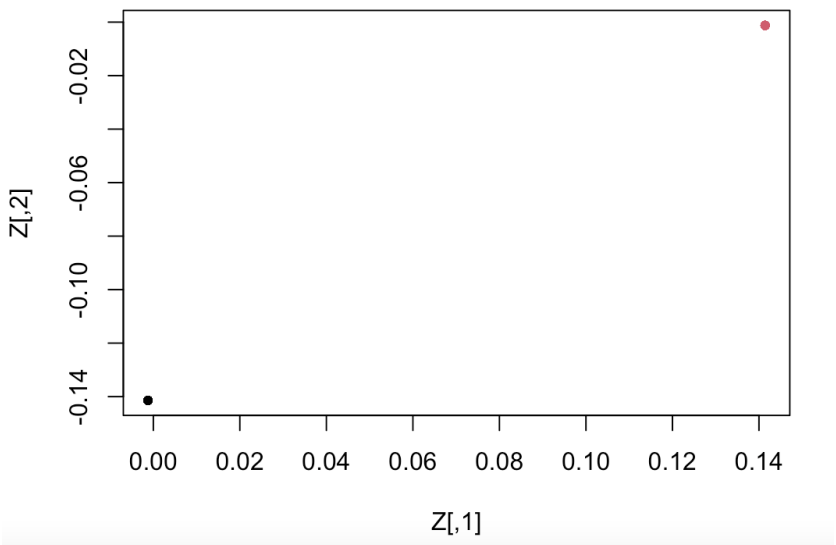
```
"%^%" <- function(M, power)
  with(eigen(M), vectors %*% (values^power * solve(vectors)))
```

Слика 31: Funkcija stepenovanja matrice

```
k   <- 2
evL <- eigen(U, symmetric=TRUE)
Z   <- evL$vectors[,(ncol(evL$vectors)-k+1):ncol(evL$vectors)]
```

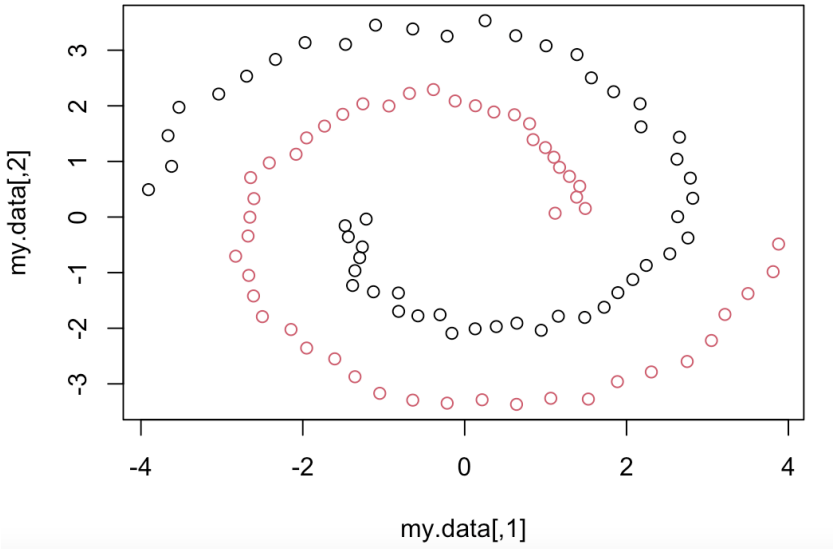
Слика 32: Računanje k minimalnih sopstvenih vektora

```
plot(Z, col=obj$classes, pch=20)
```



Слика 33: Pokazujemo dobru razdvojenost tačaka

```
library(stats)
km <- kmeans(Z, centers=k, nstart=5)
plot(my.data, col=km$cluster)
```



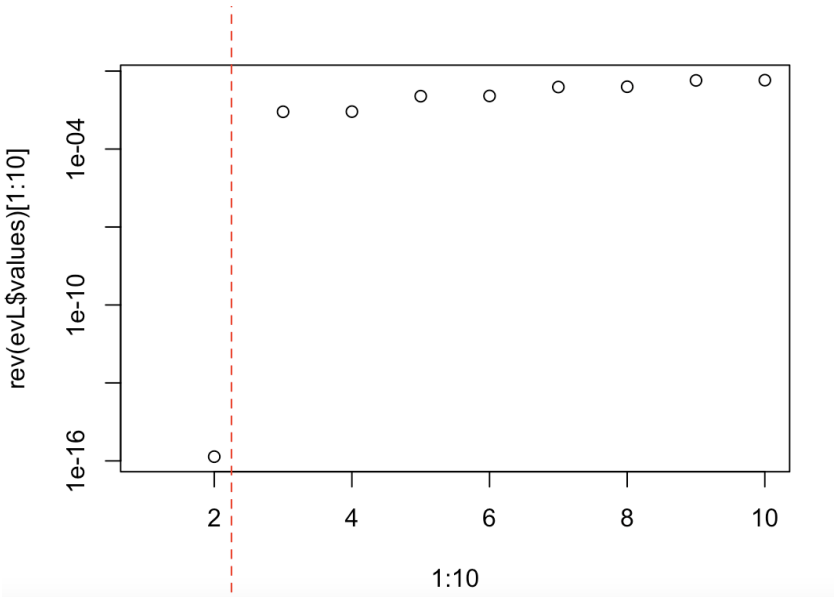
Слика 34: Klasteri izdvojeni metodom k sredina

Ako ne znamo koliko klastera postoji, spektar sopstvenih vrednosti će nam pružiti *pukotinu* koja nam daje broj k.

```
signif(evL$values,2)
```

```
 [1] 3.3e+00 3.3e+00 3.1e+00 3.0e+00 2.9e+00 2.9e+00 2.9e+00 2.8e+00
 [9] 2.8e+00 2.7e+00 2.7e+00 2.7e+00 2.6e+00 2.6e+00 2.6e+00 2.6e+00
[17] 2.5e+00 2.5e+00 2.5e+00 2.5e+00 2.4e+00 2.4e+00 2.4e+00 2.4e+00
[25] 2.3e+00 2.3e+00 2.3e+00 2.2e+00 2.2e+00 2.2e+00 2.2e+00 2.1e+00
[33] 2.1e+00 2.1e+00 2.0e+00 2.0e+00 2.0e+00 1.9e+00 1.9e+00 1.9e+00
[41] 1.8e+00 1.8e+00 1.7e+00 1.7e+00 1.7e+00 1.6e+00 1.6e+00 1.6e+00
[49] 1.5e+00 1.5e+00 1.4e+00 1.4e+00 1.3e+00 1.3e+00 1.2e+00 1.2e+00
[57] 1.1e+00 1.1e+00 1.0e+00 1.0e+00 9.5e-01 9.3e-01 8.6e-01 8.5e-01
[65] 7.8e-01 7.6e-01 6.9e-01 6.8e-01 6.2e-01 6.0e-01 5.4e-01 5.3e-01
[73] 4.7e-01 4.6e-01 4.0e-01 4.0e-01 3.4e-01 3.3e-01 2.8e-01 2.7e-01
[81] 2.3e-01 2.2e-01 1.8e-01 1.8e-01 1.4e-01 1.3e-01 1.0e-01 9.9e-02
[89] 7.0e-02 6.8e-02 4.5e-02 4.4e-02 2.5e-02 2.4e-02 1.1e-02 1.1e-02
[97] 2.8e-03 2.7e-03 1.5e-16 -6.6e-16
> 
```

```
plot(1:10, rev(evL$values)[1:10], log="y")
abline(v=2.25, col="red", lty=2)
```

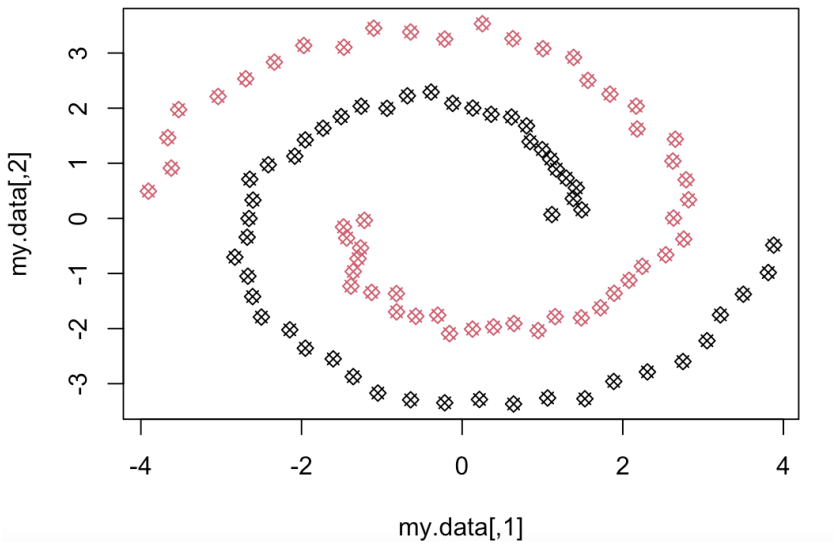


Слика 35: Prikaz pukotine koja nam govori vrednost k

Sve ovo je implementirano u *R*-u i može se dobiti korišćenjem paketa *kernlab*.

```
install.packages("kernlab")
library(kernlab)

sc <- specc(my.data, centers=2)
plot(my.data, col=sc, pch=4)
points(my.data, col=obj$classes, pch=5)
```



Слика 36: izdvojeni klasteri korišćenjem kernlab-a

6 Zaključak

Unutar projekta istaknuti su načini korišcenja teorije grafova, mera centralnosti i spektralnog grupisanja za analizu složenih mreža. Ovi alati i tehnike su primenjeni na specifičnu mrežu, tj. skup podataka, dajuci uvid u strukturu i funkciju sistema. Rezultati ove studije sugerišu da se ove metode mogu efikasno koristiti za proučavanje drugih sličnih mreža.

Jedan od ključnih doprinosa rada je inkorporacija igre „Šest stepeni Kevina Bejkona" kao metode za ispitivanje društvenih veza. Ovaj pristup nam je omogucio da ilustrujemo koncept mreža „malog sveta", u kojima su pojedinci povezani malim brojem posrednika, i da demonstriramo potencijal takvih mreža da olakšaju širenje informacija i uticaja.

Zaključili smo da je spektralno grupisanje mocno sredstvo za otkrivanje osnovne strukture mreže i identifikaciju zajednica unutar nje. Korišcenjem sopstvenih vektora Laplasove matrice, spektralno klastovanje je u stanju da efikasno podeli mrežu u koherentne grupe, čak i kada struktura nije jasno definisana.

Međutim, vredi napomenuti da spektralno grupisanje nije bez ograničenja. U nekim slučajevima, kvalitet klastera može zavisiti od izbora broja klastera i tipa korišcene Laplasove matrice. Pored toga, spektralno grupisanje može biti osetljivo na šum i možda nece raditi dobro na veoma nepravilnim ili nepovezanim grafovima.

Ova studija naglašava važnost korišcenja mera centralnosti, teorije grafova i spektralnog grupisanja za razumevanje dinamike društvenih mreža. Ove metode nam omogucavaju da identifikujemo obrasce i trendove koji možda nisu očigledni iz pojedinačnih tačaka podataka i da steknemo uvid u procese koji oblikuju ove sisteme i ponašanje njihovih članova.

Литература

[1] Wasserman, S., Faust, K. (1994).*Social Network Analysis: Methods and Applications*. Cambridge: Cambridge University Press, 2nd ed.

[2] West,Douglas B. (2001) *Introduction to Graph Theory*. Prentice Hall, 2nd ed.

[3] Everitt B., Landau S., Leese M. and Stahl D. (2011) *Cluster Analysis*, John Wiley Sons, Ltd ., 1st ed.

[4] Aggarwal, Charu C. and Reddy, Chandan K. (2013) *Data Clustering: Algorithms and Applications*. Chapman Hall/CRC, 1st ed.