Stručni članak / Professional paper

UDK/UDC 004.62

Vladimir Špišić, dipl. ing.[1]
Ivan Štedul, prof.[2]
mr. sc. Vedran Vyroubal[3]

KORIŠTENJE *OPEN SOURCE* ALATA ZA ANALIZU PODATAKA GRUPIRANJEM

Clustering with Open Source Tools

SAŽETAK: U većini istraživanja obrada podataka predstavlja jedan od ključnih koraka. U većem broju slučajeva struktura tih podataka nije unaprijed poznata, te je potrebno tijekom analize grupirati podatke u klastere te iz takve strukture donijeti zaključke istraživanja.
Danas se koristit vrlo veliki broj metoda za navedeni način obrade podataka i različitih alata pomoću kojih se te obrade izvršavaju. Razvijen je i veći broj alata koji su slobodni za korištenje (*engl*. open source) čija kvaliteta u sve više slučajeva prerasta kvalitetu komercijalnih rješenja.
U radu će biti dan pregled jedne od najčešće korištenih metoda obrade podataka hijerarhijskim grupiranjem.
Zatim će biti predstavljeni slobodni alati koji se mogu koristiti za primjenu te metode (npr. CLUTO, R,...). Neki od alata su samostalne aplikacije, dok su ostali alati biblioteke klasa koje se mogu jednostavno primijeniti iz nekog programskog jezika pomoću programskih sučelja.

Ključne riječi: grupiranje podataka, open source

ABSTRACT: Data processing represents one of the key steps during most research projects. In most cases structure of data to process is not known in advance, so it is necessary during the data analysis to group the research data into data clusters, from which research conclusions can be derived.
Today large numbers of methods, as well as diverse set of software tools are used for data clustering. Many of such software tools are open source software, which in quality in many cases surpass the quality of many commercial software solutions.
This paper will provide an overview of one of the most used methods for hierarchical data clustering, as well as overview of open source software tools for using the afore mentioned method (e.g. CLUTO, R). Some of the software tools are imple-

[1] Veleučilište u Karlovcu, vladimir.spisic@vuka.hr
[2] Veleučilište u Karlovcu, ivan.stedul@vuka.hr
[3] Veleučilište u Karlovcu, vedran.vyroubal@vuka.hr

mented as standalone applications, while others are implemented as libraries which can be easily invoked from within some other programming language development environment.

Keywords: clustering, open source

1. UVOD

Podaci su rezultat gotovo svih istraživačkih aktivnosti. Opisuju karakteristike živih bića, prirodnih pojava, dinamičkih sustava, karakteristika materijala, trenutnih stanja sustava, promjene u vremenu nekog sustava itd. U istraživanjima ti podaci su osnova za analizu, zaključivanje i razumijevanje predmeta istraživanja. Često je potrebno podatke klasificirati ili grupirati u neke kategorije ili klastere. Te kategorije mogu biti unaprijed poznate ili se tek analizom pronalazi broj kategorija koje ćemo u budućnosti koristiti. Cilj nam je da podaci unutar iste kategorije prikazuju slične karakteristike prema nekom kriteriju.

U radu će biti dan pregled jedne od najčešće korištenih metoda obrade podataka hijerarhijskim grupiranjem.

Zatim će biti predstavljeni slobodni alati koji se mogu koristiti za primjenu te metode (npr. CLUTO, R,...).

2. GRUPIRANJE PODATAKA ANALIZOM

Klasifikacija u osnovi može biti nadzirana ili nenadzirana. Razliku predstavlja znanje o grupama, pa u slučaju nadziranog učenja novi objekt (predmet ili pojava koju klasificiramo) dodjeljuje u jednu od konačnog broja diskretnih grupa koje su unaprijed poznate, dok kod nenadzirane klasifikacije te grupe unaprijed nisu poznate.

Objekt (predmet ili pojavu) predstavljamo sa nizom vrijednosti koje predstavljaju pojedine karakteristike izmjerene ili određene prema nekoj mjeri ili pravilu. Npr. risa možemo predstaviti sa nizom – dužina tijela, visina u ramenima, težina, boja krzna, pjege ili pruge, pa dvije različite životinje mogu imati sljedeće karakteristike:

1. 130 cm, 64 cm, 22 kg, žuta, pruge i

2. 125 cm, 66 cm, 21 kg, sivo smeđa, pjege.

Ako opisnim vrijednostima dodijelimo kodove (npr. žuta - 1, sivo smeđa - 2 itd.) tada navedene karakteristike pišemo u obliku vektora realnih brojeva (130, 64, 22, 1, 1) i (125, 66, 21, 2, 2), te ih označavamo kao $x \in \mathbb{R}^d$. Sa d je označena dimenzionalnost ulaznog prostora, tj. broj karakteristika u vektoru.

U nadziranoj klasifikaciji imamo dostupan poznati set podataka, tj. ulaznih vektora x_i, $i \in N$ (gdje N predstavlja broj dostupnih uzoraka) i pripadne grupe y_i, $i \in C$(gdje je C ukupan broj grupa), pa je i prvi korak učenje sustava pomoću poznatih podataka. Dakle u fazi učenja cilj je odrediti vektor v slobodnih parametara pomoću kojega funkcijom $y = y(x,v)$ možemo klasificirati novi objekt u ispravnu grupu. Ova vrsta učenja

se naziva nadzirano učenje. Postoji veliki broj metoda nadziranog učenja i uspjeh pojedine metode je ovisan o količini i distribuciji ulaznih podatka. Za detaljni pregled metoda mogu se konzultirati različite knjige[4].

Nenadzirana klasifikacija se koristi kada ne postoji unaprijed određena pripadnost grupa uz poznate vektore karakteristika, tj. objekata x_i, te u velikom broju slučajeva ne postoji niti znanje o broju grupa. U nenadziranu klasifikaciju ubrajamo slijedeće procese:

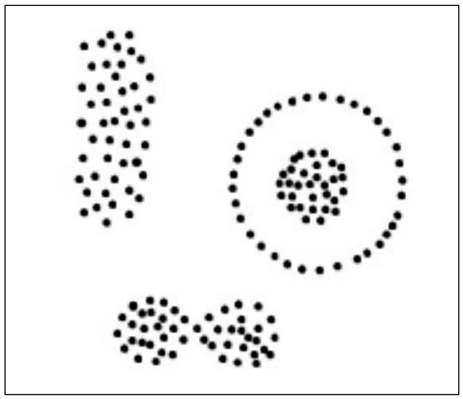
- **grupiranje podataka analizom** (*engl*. clustering) – pronalaženje grupa koje se sastoje od objekata sa sličnim karakteristikama, tj. pronalaženje skrivenih (ili slabo uočljivih) struktura u podacima koje imamo;
- **određivanje gustoće podataka** (*engl*. density estimation) – određivanje gustoće podataka u ulaznom prostoru, tj. pronalaženje najbliže poznate distribucije koja bi mogla opisati postojeće podatke;
- **vizualizacija** – projiciranje podataka iz visoko dimenzionalnog prostora u dvije ili tri dimenzije gdje se mogu vizualno utvrditi odnosi između podataka.

2.1. Definicija klastera

Grupiranje analizom podataka svrstava objekte (uzorke, entitete, instance, opažanja, jedinice) u određeni broj klastera (grupa, podskupova, kategorija). Ipak, u literaturi nema jedinstvene definicije termina klaster. Definicija koja najbolje odgovara rasuđivanju u ovom radu glasi:

„*Klaster je skup točaka u testnom prostoru takav da udaljenost između bilo koje dvije točke u klasteru je manja od odaljenosti bilo koje točke u klasteru i točke izvan klastera*“[5]

Slika 1. Klasteri kao točke u prostoru



Na Slici 1. su navedeni neki primjeri grupa točaka koji prikazuju klastere na više razina. Gruba podjela nam daje 3 grupe, dok detaljnija (finija) podjela daje 5 različitih grupa – najniži klaster se može podijeliti u dvije grupe (lijevu i desnu), te krug u dvije grupe (kružnicu i jezgru). Ovo je ujedno primjer subjektivnosti grupiranja o čemu će biti riječ u opisu procesa grupiranja, točnije određivanja kvalitete klastera.

[4] Bishop, C.M.: **Pattern Recognition and Machine Learning**, Springer, 2006. Everitt, B.: **Cluster analysis**, London: Social Science Research Council, 1980.

[5] Everitt, B.: **Cluster analysis**, London: Social Science Research Council, 1980.

2.2. Proces grupiranja podataka analizom

Proces grupiranja podataka analizom možemo podijeliti u četiri osnovna koraka:

1. **Odabir ili ekstrakcija karakteristika**. Odabir karakteristika označava odabir onih karakteristika koje će nam pomoći razlikovati objekte upravo na način koji je cilj grupiranja, dok ekstrakcija karakteristika uključuje neku transformaciju originalnih karakteristika na načina da se ta razlika u korisnom smislu dodatno poveća[6]. Naravno, ekstrakcija karakteristika se češće koristi u nadziranim metodama (jer nam je struktura podataka jasna i imamo primjere ispravnog grupiranja).
2. **Odabir algoritma grupiranja**. Uobičajeno je prvo odrediti prikladnu mjeru udaljenosti između pojedinih karakteristika. Taj korak je vrlo važan jer su svi algoritmi grupiranja ovisni o toj mjeri, što je i intuitivno jasno. Odabir mjera koje ne ističu različitost između objekata koji pripadaju različitim grupama, povećava pogrešku grupiranja u svim algoritmima grupiranja[7]. Odabir algoritma grupiranja je također ovisan o distribuciji podataka koje posjedujemo i često u samim algoritmima već postoje neke pretpostavke (npr. algoritam k-srednjih vrijednosti pretpostavlja euklidsku udaljenost te kreira hipersferične klastere).

   Vrste algoritama možemo grubo podijeliti prema vrsti grupiranja:

   - hijerarhijsko grupiranje;
   - particijsko grupiranje;
   - grupiranje korištenjem neuronskih mreža;
   - grupiranje jezgrenim metodama;
   - grupiranje sekvencijalnih podataka.

   Hijerarhijsko grupiranje je odabrano zbog vrlo rasprostranjenog korištenja ali i implementacije u najraširenijim alatima otvorenog koda. Za ostale metode predlaže se pregled u referentnoj literaturi[8].
3. **Validacija klastera**. Općenito za validaciju primjenjujemo vanjske, unutarnje ili relativne kriterije. Vanjski kriteriji pretpostavljaju neke informacije o strukturi podataka koju znamo unaprijed kao što je npr. konačni broj klastera, te se najčešće koriste u particijskom grupiranju. Unutarnji kriteriji testiraju samo poznate informacije iz samih podataka i trenutnog koraka grupiranja (npr. veličina klastera u odnosu na druge klastere, entropija pojedinog klastera prema drugima itd.) i koriste se češće u metodama hijerarhijskog grupiranja. Relativni kriteriji jednostavno uspoređuju različite strukture klastera te se odabire kriterij koji u tom trenutku daje najveću različitost i sl.

[6] Jain A., Murty M., Flynn P.: **Data clustering: A review**. ACM Computing Surveys, 1999.

[7] Cha, S.: **Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions**. International Journal of Mathematical Models and Methods in Applied Sciences, 2007.

[8] Theodoridis S., Koutroumbas K.: **Pattern Recognition**, Fourth Edition. Academic Press, 2008.

4. **Interpretacija rezultata**. Konačni cilj grupiranja je pronalaženje skrivene strukture u ulaznim podacima, odnosno otkrivanje grupa kojoj podaci pripadaju. Uobičajeno da eksperti iz različitih područja u interpretaciji rezultata uključuju dokaze iz ostalih eksperimenata, postojećih saznanja i pretpostavki, tj. da se ne oslanjaju isključivo na rezultate grupiranja podataka.

3. HIJERARHIJSKO GRUPIRANJE

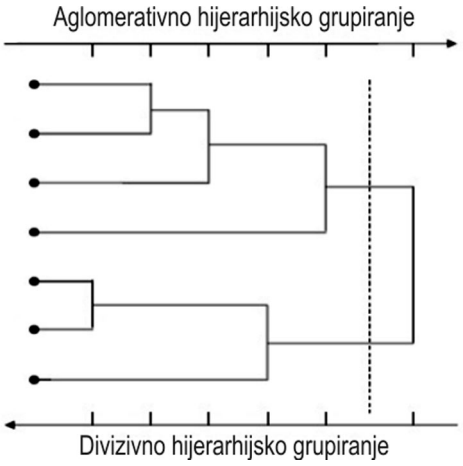
Metode hijerarhijskog grupiranja grupiraju podatke u slijed ugniježđenih grupa kao što je prikazano na Slici 2. Aglomerativno hijerarhijsko grupiranje predstavlja grupiranje od pojedinačnih objekata prema jednom zajedničkom klasteru. Divizivno grupiranje s druge strane u svakom koraku dijeli postojeće klastere prema nekom kriteriju. Rezultat hijerarhijskog grupiranja se uobičajeno prikazuje u obliku dendograma tj. binarnog stabla gdje svaka grana može imati točno dvije grane ili lista. U nastavku će se termin hijerarhijski algoritam ili samo algoritam koristiti isključivo za aglomerativni algoritam. Sva pravila i objašnjenja vrijede i za divizivni algoritam uzimajući u obzir različiti smjer grupiranja. Divizivni algoritmi u svakom koraku moraju ispitati udaljenosti mogućih podjela za točaka, što je računski vrlo zahtjevno, te se aglomerativni algoritmi puno češće koriste.

Hijerarhijski algoritmi organiziraju podatke na osnovu matrice udaljenosti. Matrica udaljenosti je simetrična matrica u koju se unose vrijednosti udaljenosti (na osnovu neke odabrane mjere udaljenosti) između svaka dva objekta. Algoritam u svakom koraku traži najmanju udaljenost između dva objekta (ili objekta i klastera, ili dva klastera). Dendogram u stvari prikazuje udaljenost između dva objekta (uobičajeno se udaljenosti ispisuju na osi pa ih je moguće jednostavno očitati), tj. svaki među-čvor prikazuje koliko su blizu dva objekta ili dvije grupe objekata. Krajnji rezultat se dobiva rezom grafa u nekom koraku. Taj rez može biti proizvoljan od strane korisnika sustava ili dobiven korištenjem nekog kriterija validacije klastera.

Ovaj način prikaza je vrlo informativan posebno u slučaju kada u podacima postoje hijerarhijski odnosi kao što su na primjer u podacima o sličnosti jezika, istraživanju evolucijskih procesa, medicini, biologiji i arheologiji.

Na Slici 2 prikazan je dendogram koji dobivamo kao rezultat aglomerativnog ili divizivnog hijerarhijskog grupiranja. Točke predstavljaju pojedine objekte. Crtkana crta označava proizvoljan rez grafa koji određuje korisnik ili se koristi neka od metoda validacije klastera. Crtice predstavljaju korake algoritma.

Slika 2. Dendogram



Algoritam aglomerativnog hijerarhijskog grupiranja se može sažeti u slijedeće korake:

1. Početi sa N jednočlanih klastera. Izračunati matricu udaljenosti za svih N klastera.
2. Pronaći minimalnu udaljenost između dva klastera $d(C_i, C_j) = \min_{\substack{1 \le m,l \le N, \\ m \ne l}} d(C_m, C_l)$ te spojiti klastere C_i i C_j u novi klaster C_{ij}.
3. Ažurirati matricu udaljenosti na način da se iz nje uklone klasteri C_i i C_j te se doda novi klaster C_{ij} i izračunaju se udaljenosti između C_{ij} i svih ostalih klastera.
4. Ponavljati korake 2. i 3. dok ne ostane samo jedan klaster, tj. svi klasteri su spojeni u jedinstveni klaster.

Uvjet zaustavljanja algoritma može biti promijenjen na način da se u 4. koraku izračunava neki kriterij validnosti klastera i donosi odluka o zaustavljanju ili nastavku daljnjeg grupiranja.

3.1. Određivanje udaljenosti između klastera

Udaljenost između pojedinih objekata je jednoznačno definirana sa nekom odabranom mjerom udaljenosti. Kada objekte spojimo u novi klaster, udaljenost tog klastera od ostalih objekata ili klastera je potrebno definirati na drugačiji način. Naime, potrebno je odrediti kako pojedini objekti u klasteru utječu na funkciju udaljenosti. U praksi imamo veći broj definicija tog utjecaja koji se općenito mogu pisati u obliku formule predložene od Lance i Williamsa 1967.:

$$D\big(C_l, (C_i, C_j)\big) = \alpha_i D(C_l, C_i) + \alpha_j D(C_l, C_j) + \beta D(C_i, C_j) + \gamma \left| D(C_l, C_i) - D(C_l, C_j) \right|$$

U navedenoj formuli C_l predstavlja klaster koji je formiran u nekom prethodnom klasteru, dok C_i i C_j predstavljaju klastere koji su spojeni u trenutnom koraku u novi klaster C_{ij}. Udaljenost između klastera se tada može definirati za najznačajnije funkcije kroz definiciju parametara $\alpha_i, \alpha_j, \beta, \gamma$ kako je navedeno u Tablici 1. (Everitt et. al. 2001). Vrijednosti n_i, n_j i n_l predstavljaju broj objekata u klasterima sa istom oznakom tj. C_i, C_j *i* C_l.

Tablica 1. Parametri $\alpha_i, \alpha_j, \beta, \gamma$ prema najznačajnijim funkcijama udaljenosti klastera

Funkcija udaljenosti	α_i	α_j	β	γ
pojedinačna veza (*engl.* single linkage – nearest neighbour)	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$
potpuna veza (*engl.* complete linkage – farthest neighbour)	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
prosječna veza (*engl.* group average linkage)	$\frac{n_i}{n_i + n_j}$	$\frac{n_i}{n_i + n_j}$	0	0

Odabir funkcije je ovisno o podacima i kao posljedicu imaju određene tendencije prilikom kreiranja klastera. Na primjer:

- u pojedinačnoj vezi udaljenost između klastera je određena udaljenošću najbližih objekata iz različitih klastera. Ovaj način stvara izdužene klastere i u slučaju smetnji (pojedinačnih objekata u prostoru između klastera) jednostavno spoji klastere.
- potpuna veza koristi udaljenost između dva najudaljenija objekta pojedinog klastera i ima tendenciju odvajanja malih ali kompaktnih klastera.

3.2 Programski jezik R

R je programski jezik otvorenog koda (*engl*. open-source), prvenstveno namijenjen za statističke proračune i vizualizaciju podataka. R se razvija u sklopu GNU projekta, te je dostupan pod licencom GNU General Public License, dok su izvršne datoteke dostupne za sve relevantne operacijske sustave. Dok postoje razna grafička razvojna okruženja za manipuliranje podacima koja za statističke izračune koriste R, primarni način korištenja R jezika ostaje naredbeni redak.

R je skriptni jezik koji omogućava većinu programskih konstrukta koje očekujemo od modernog programskog jezika (npr. kreiranje petlji, uvjetno grananje,...). Skriptna priroda jezika može postati usko grlo u brzini izvršavanja programa, tada je moguće proširiti R modulima pisanima u nižim jezicima koji se prevode u strojni kod. Također moguće je iz drugih programskih jezika direktno manipulirati objektima kreiranima u R-u. Za razliku od ostalih programskih jezika poput SPSS-a i SAS-a kojima je statistička obrada podataka osnovna funkcionalnost R omogućava i objektno-orijentiranu paradigmu pri pisanju programa.

Distribucijski paket jezika R dolazi s mnogim statističkim procedurama i algoritmima, među koji su linearni i generalizirani linearni modeli, nelinearni regresijski modeli, procedure za analizu podataka grupiranjem, te s mnogim drugim procedurama. Uz osnovnu namjenu R-a (statistički proračuni) moguće ga je koristiti i za matrične proračune, s performansama koje se približavaju mnogim skupim komercijalnim alatima (npr. MATLAB). Potpuna lista osnovnih i dodatnih modula za jezik R je dostupna na web stranicama projekta samog jezika R [8], te na stranicama CRAN (*engl*. Comprehensive R Archive Network) projekta [9]. Također je bitno napomenuti sposobnost R-a u dohvaćanju podataka s različitih izvora. R nije ograničen na učitavanje podataka iz standardnih tekstualnih datoteka (*engl*. CSV, comma separated values), već je u stanju učitavati podatke iz datoteka mnogih komercijalnih rješenja (npr.: Excel, SAS, SPSS), te dohvaćati podatke iz objektno-relacijskih baza podataka.

Sama sintaksa jezika je relativno jednostavna i intuitivna, dok naprednije korištenje zahtjeva upoznavanje s odgovarajućim bibliotekama i njihovim procedurama. Jednostavan način za prikaz dostupnih biblioteka je pozivom naredbe:

```
> library()
```

Za analizu podataka grupiranjem potrebno je učitati biblioteku “cluster”:

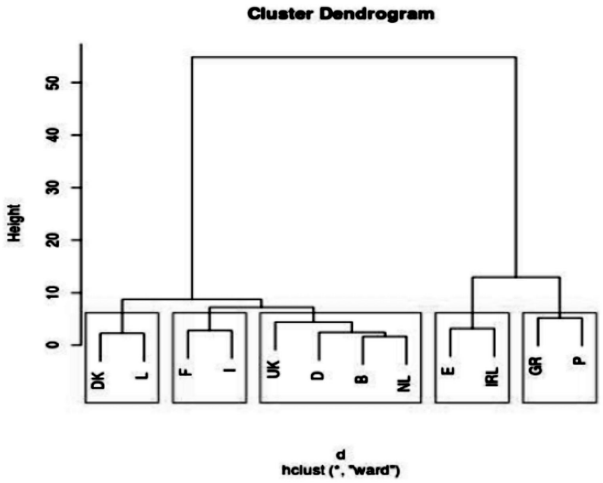
```
> library(cluster)
```

U nastavku je prikazana jednostavna analiza podataka grupiranjem, korištenjem Ward metode i euklidske udaljenosti:

```
> library(cluster)
> podaci <- agriculture # primjer seta podataka o radnoj snazi
u poljoprivredi u EU
> d <- dist(podaci, method = "euclidian") # izračun distanci
> fit <- hclust(d, method="ward")
> plot(fit) # crtavanje dobivenog grupiranja
> groups <- cutree(fit, k=5) # rezanje grafa
> rect.hclust(fit, k=5, border="red") # uokvirivanje grupa
```

Dobiveni rezultati prikazani su grafom na Slici 3.

Slika 3. Rezultati analiza podataka grupiranjem u programskom jeziku R



3.3. Programski paket CLUTO

CLUTO je softverski paket namijenjen klasteriranju jednodimenzionalnih i višedimenzionalnih skupova podataka i za analizu obilježja različitih klastera. Program je kreirao tim profesora Georga Karypisa na Sveučilištu Minesota [10]. CLUTO nudi tri različite vrste algoritama namijenjenih klasteriranju. Ti algoritmi su algoritam particijskog grupiranja, algoritam grupiranja baziranog na teoriji grafova i algoritam aglomerativnog grupiranja.

Sama distribucija CLUTO sastoji se od samostalnih programa scluster i vcluster, algoritama namijenjenih klaster analizi i dvije vrste funkcija, kriterijska funkcija i funkcija sličnosti.

Glavna značajka većine algoritama namijenjenih klaster analizi u CLUTO-u je da problemu klasteriranja pristupaju kao problemu optimizacije procesa kojim se nastoji minimizirati ili maksimizirati određena funkcija sličnosti. U CLUTO je implementirano ukupno sedam funkcija sličnosti koje koristimo u aglomerativnim klaster algoritmima.

U sklopu CLUTO projekta razvijena je grafička aplikacija gCLUTO namijenjena vizualizaciji rezultata klaster analize provedene CLUTO-m.

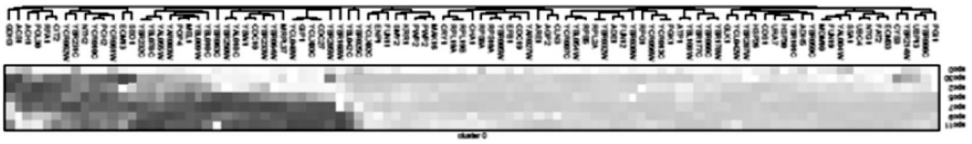
Poput R-a CLUTO-ve izvršne datoteke dostupne za sve relevantnije operacijske sustave. Primaran način korištenja CLUT-a je poput R-a putem naredbenog retka. U ovom radu biti će prikazana jednostavna analiza podataka grupiranjem.

CLUTO-vi programi koji su predviđeni za klaster analizu, su vcluster i scluster. Pomoću tih programa možemo kreirati hijerarhijsko aglomerativno stablo (Dendogram) u kojem su pronađeni klasteri. Slijedi jednostavan primjer takve klaster analize korištenjem programa vcluster. Naredbom –rclassfile u programu vclaster izračunavamo informacije o pripadnosti različitih objekata klasterima i statistički izračunavamo kvalitetu te pripadnosti. Hijerarhijsko aglomerativno stablo u programu vcluster kreiramo pomoću parametra –showtree. Parametrom –plottree dobivamo i grafički prikaz dendograma. Rezultati jedne klaster analize i kreiranja hijerarhijskog aglomerativnog stabla programom vcluster u windows okruženju prikazano je na Slici 4.

Ulazni podaci u program vcluster nalaze se u datoteci genes2.mat i organizirani su u obliku matrice. U toj matrici svaki redak predstavlja jedan objekt gdje stupci matrice korespondiraju s dimenzijom objekta. Podacima iz matrice koja je spremljena u datoteci upravljamo pomoću parametara iz naredbenog retka: Primjer je preuzet iz[9]

naredba: `vcluster -clmethod=agglo -plotmatrix=fig7.ps genes2.mat 1`

Slika 4. Prikazuje vizualizaciju generiranu parametrom – plotmatrix klaster analize dobivene aglomerativnom metodom vcluster programa



[9] **CLUTO - Software for Clustering High-Dimensional Datasets**. http://glaros.dtc.umn.edu/gkhome/views/cluto (21.03.2012.)

4. ZAKLJUČAK

Grupiranje podataka (*engl*. clustering) omogućuje analizu nepoznate ili skrivene strukture podataka. Cilj ovog rada bio je dati pregled jedne od najčešće korištenih metoda obrade podataka hijerarhijskim grupiranjem i predložiti neke alate otvorenog koda kojima se ta obrada može provesti.

Predloženi su programski paketi R i CLUTO. Oba predložena programa su alati otvorenog koda kojima se mogu provesti različita grupiranja (*engl*. cluster analysis) i jednostavno učitavati podatke u različitim oblicima i iz različitih sustava. Ti alati svojim mogućnostima nimalo ne zaostaju za komercijalnim alatima poput MATLABA, SPSS-a ili SAS-a, već u potpunosti omogućuju realizaciju kvalitetne analize, a na dva jednostavna pokazna primjera je pokazano da je korištenje istih vrlo jednostavno.

Literatura

1. Bishop, C.M.: **Pattern Recognition and Machine Learning**,. Springer, 2006.
2. Duda, R.O., Hart, P.E., Stork, D.G.: **Pattern Classification**, Wiley, 2001.
3. Xu R., Wunsch D.: **Clustering**, Wiley – IEEE Press, 2008.
4. Everitt, B.: **Cluster analysis**, London: Social Science Research Council, 1980.
5. Jain A., Murty M., Flynn P.: **Data clustering: A review.** ACM Computing Surveys, 1999.
6. Cha, S.: **Cmoprehensive Survey on Distance/Similarity Measures between Probability Density Functions**. International Journal of Mathematical Models and Methods in Applied Sciences, 2007.
7. Theodoridis S., Koutroumbas K.: **Pattern Recognition**, Fourth Edition. Academic Press, 2008.
8. **The R Project for Statistical Computing**. http://www.r-project.org/ (15.03.2012.)
9. **Comprehensive R Archive Network**. http://cran.r-project.org/ (15.03.2012.)
10. **CLUTO - Software for Clustering High-Dimensional Datasets**. http://glaros.dtc.umn.edu/gkhome/views/cluto (21.03.2012.)