Omejeno gručenje

Ana Golob, Primož Durcik

1. Uvod

Gručenje že dolgo predstavlja enega temeljnih delov kombinatorične optimizacije in analize podatkov. V pričujoči projektni nalogi se bomo najprej seznanili z osnovnim teoretičnim ozadjem omejenega gručenja. Le-to bo v drugem delu predstavljalo osnovo za pisanje algoritma za optimalno gručenje, ki bo ravno reševanje linearnega programa.

V grobem si pri omejenem gručenju želimo dano množico obteženih točk množice X v določenem prostoru $\mathcal{X}$ razdeliti na dano število k gruč z vnaprej določeno težo.

Algoritem bova uporabila na dveh eksperimentih:

V prvem eksperimentu rešujemo problem oblikovanja volilnih enote. Zanima nas, kako razčleniti notranje območje države na posamezne volilne okraje. Pri tem zahtevamo, da okraji pokrivajo skoraj enake populacije volivcev in imajo »razumno« obliko.

V drugem eksperimentu imamo podatke o stanovanjih, ki se prodajajo v okolici Ljubljane. Glede na njihove lastnosti jih želimo razdeliti v gruče, tako da bo vsaka gruča vsebovala stanovanja s čim bolj podobnimi lastnostmi.

2. Teoretični opis omejenega gručenja

Naj bosta $k, n \in \mathbb{N}$ in $\mathcal{X}$ poljuben prostor. Z $X = \{x_1, \dots x_m\} \subset \mathcal{X}$ označimo množico točk s pripadajočimi utežmi $\Omega = (\omega_1, \dots \omega_m) \in \mathbb{R}^m_{>0}$. Nadalje, naj bo $\mathcal{K} = (\kappa_1, \dots, \kappa_k) \in \mathbb{R}^k_{>0}$, tako da $\sum_{i=1}^k \kappa_i = \sum_{j=1}^m \omega_j$. $\mathcal{K}$ je vektor željenih "velikosti" gruč. Naš cilj je poiskati take particije množice X, da bo skupna teža gruče C_i, v prej določeni normi "čim bližje" prepisanemu κ_i.

Uporabljali bomo spodnjo predpostavko, ki problem nekoliko poenostavi:

Naj bo

$$C = (\xi_{i,j})_{\substack{i=1,\dots,k \\ j=1,\dots,m}} \in [0,1]^{k \times m},$$

tako da je vsota $\sum_{i=1}^k \xi_{i,j} = 1$ za vsak j. C imenujemo delno gručenje množice X in $\xi_{i,j}$ je del enote j predpisane gruči i. Tako definirane $C_i = (\xi_{i,1}, \dots, \xi_{i,m})$ imenujemo gruča i. Če je $C \in \{0,1\}^{k \times m}$, imenujemo gručenje celoštevilsko.

Teža gruče je podana z $\omega(C_i) = \sum_{j=1}^m \xi_{i,j}\omega_j$. Gručenje C je močno uravnoteženo, če je $\omega(C_i) = \kappa_i$ za vsak i. Če so za teže grozdov dane zgornje in spodnje meje κ_i^-, κ_i^+ in velja: $\kappa_i^- < \sum_{j=1}^m \xi_{i,j}\omega_j < \kappa_i^+$ za vsak i, potem pravimo, da je gručenje C šibko uravnoteženo. V posebnem primeru vzamemo $\kappa_i^- = (1-\epsilon)\kappa_i$ in $\kappa_i^+ = (1+\epsilon)\kappa_i$ za vsak i in dan $\epsilon > 0$ in tako gručenje imenujemo ϵ-gručenje. Z BC in BC^ϵ označujemo množici vseh močno uravnoteženih in ϵ-uravnoteženih delnih gručenj.

Za program je potrebno definirati še funkcije $f_i : \mathcal{X} \to \mathbb{R}$, $i=1,\dots ,k, ki določajo C_i, podmnožico $\mathcal{X}$ tako, da za vsak $x \in \mathcal{X}$ velja: če $f_i(x)$ je minimalen, potem $x \in C_i$.

Če želimo najti ustrezne rešitve je ključnega pomena, da dobro definiramo funkcije f_i. Za nabor parametrov $(\mathcal{D}, h, \mathcal{S}, \mathcal{M})$ tako definiramo k-terico funkcij $\mathcal{F}(\mathcal{D}, h, \mathcal{S}, \mathcal{M}) = (f_i, \dots, f_k)$ s predpisom:

$$f_i(x) = h(d_i(s_i, x)) + \mu_i,$$

kjer je:

- $\mathcal{D} = (d_1, \ldots, d_k)$ k-terica metrik, oziroma predpisov za merjenje razdalj na prostoru $\mathcal{X}$,
- $h : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ monotono naraščajoča funkcija,
- $\mathcal{S} = (s_1, \ldots, s_k)$ k-terica točk iz $\mathcal{X}$,
- $\mathcal{M} = (\mu_1, \ldots, \mu_k) \in \mathbb{R}^k$.

Če metrike d_i niso identične, dobimo anizotropen primer.

Primeri funkcij, ki jih lahko izberemo:

1. $f_i = \|x - s_i\|_2 + \mu_i$ v Evklidskem prostoru.
2. V diskretnem primeru, kjer velja $\mathcal{X} = X$, je dan povezan graf $G = (X, E, \delta)$; kjer $\delta : E \to \mathbb{R}_{>0}$ priredi vsaki povezavi neko pozitivno vrednost. Če je $d_G(x, y)$ definitana kot najkrajša pot od x do y, to inducira metriko na $\mathcal{X}$. Torej dobimo funkcije f_i oblike: $f_i = d_G(s_i, x) + \mu_i$.

Videli bomo, da parametra $\mathcal{D}$ in h v glavnem določata karakteristike končnih diagramov. Točke si služijo kot referenčne točke za določanje gruč.

Da se pokazati, da za vsako izbiro $\mathcal{D}$, h in $\mathcal{S}$, obstaja izbira dodatnih parametrov $\mathcal{M}$ takšna, da so porojene gruče tako predpisanih tež kot tudi optimalno porazdeljene.

Parametre $\mathcal{D}$, h in $\mathcal{S}$ imenujemo strukturni parametri, $\mathcal{M}$ pa izbirni parameter. Za vsako izbiro strukturnih parametrov izbirni parameter $\mathcal{M}$ dobimo z rešitvijo linearnega programa.

3. Algoritem

Dokazati se da, da je rešitev problema optimalnega gručenja ravno rešitev linearnega programa, oziroma njegovega duala.

Algoritem linearnega programa je naslednje oblike:

vhodni podatki:

- množica točk $X = \{x_1, \ldots x_m\} \subset \mathcal{X}$, ki jo bomo razdelili na gruče,
- uteži $\Omega = (\omega_1, \ldots \omega_m)$ na točkah
- središča gruč $\mathcal{S} = (s_1, \ldots, s_k)$
- razdalje $\mathcal{D} = (d_1, \ldots, d_k)$,

izhodni podatki: matrika

$$C = (\xi_{i,j})_{\substack{i=1,\ldots,k \\ j=1,\ldots,m}} \in [0,1]^{k \times m},$$

ki določa, v katerih gručah se posamezne točke nahajajo.

Minimiziramo:

$$\sum_{i=1}^{k} \sum_{j=1}^{m} \xi_{i,j} \omega_j d_i(s_i, x_j)$$

pri pogojih

$$\sum_{i=1}^{k} \xi_{i,j} = 1, \qquad j = 1, \ldots, m$$

$$\sum_{j=1}^{m} \xi_{i,j} \omega_j = \kappa_i \qquad i = 1, \ldots, k$$

$$\xi_{i,j} > 0 \qquad j = 1, \ldots, m; i = 1, \ldots, k.$$

Z vpeljavo pomožnih spremenljivk $E = (\eta_1, \ldots, \eta_m)$ lahko zgornjemu programu priredimo dual oblike:\ maksimiziramo

$$\sum_{j=1}^{m} \omega_j \eta_j - \sum_{i=1}^{k} \kappa_i \mu_i$$

pri pogojih

$$\eta_j \leq d_i(s_i, x_j) + \mu_i \qquad j = 1, \ldots, m; i = 1, \ldots, k.$$

Očitno velja, da parametri $\eta_j, j = 1, \ldots, m$ zadoščajo enakosti

$$\eta_j = \min_{i=1,\ldots,k} d_i(s_i, x_j) + \mu_i.$$

Psevdo kodo algoritmov si lahko ogledate v datotekah "LinearniProgram.R", "DualniLinearniProgram.R".

4. Eksperiment: gručenja volilnih enot

Cilj tega eksperimenta je bil določiti volilne okraje Slovenije, ki imajo približno enako število prebivalcev. Podatke smo pridobili na sledeč način: najprej smo pridobili število prebivalcev za slovenske kraje (datoteka "eksperiment_volitve/uvoz/populacije_krajev.csv"). Nato smo s funkcijo "geocode" pridobili koordinate krajev. Zaradi omejitev, ki jih ima funkcija "geocode", je ta postopek trajal dolgo časa. Nato smo pridobljene podatke shranili v novo datoteko (eksperiment_volitve/tabela_krajev_populacij_koordinat.csv), ki smo jo v nadaljevanju uporabljali (koda pridobivanja koordinat je v datoteki "eksperiment_volitve/uvoz/koordinate.R").

4.1 Podatki

S pridobljenimi podatki smo začela konstruirat ustrezen linearni program. Na spletu ("https://sl.wikipedia.org/wiki/Volilne_enote_v_Sloveniji") smo dobila željene informacije o volilnih enotah. Slovenija je razdeljena na 8 enot: Kranj, Postojna, Ljubljana Center, Ljubljana Bežigrad, Celje, Novo mesto, Maribor in Ptuj. Vsaka enota ima približno 200000 volilnih upravičencev. Ker pa smo v nasi nalogi upostevali vse prebivalce in ne le volilnih upravicencev, smo dobili nekoliko večje število na enoto. Prav tako nismo imeli podatkov za posamezne dele Ljubljane, zato smo se odločili, da bomo ločili primere:

- enoti Ljubljana Center in Ljubljana Bežigrad združimo v mesto Ljubljana in obravnavamo kot eno enoto,
- Ljubljano postavimo samo kot eno enoto (Ljubljana ima prebivalcev približno ravno za eno enoto) in nato še kot središče ene enote, tako da dobimo dve enoti s središčem v Ljubljani.

4.2 Vrste uporabljenih norm

Za razdalje med točkami (mesti) smo uporabili Evklidsko normo

$$\|x\| = \sqrt{x_1^2 + x_2^2},$$

kjer je x vektor $x = (x_1, x_2)$, ter elipsoidno normo. Elipsoidna norma je definirana na sledeč način: na podlagi celoštevilske rešitve iz Evklidskega primera smo najprej določili ustrezna središča po formuli

$$s_i = c(C_i) = \frac{1}{\kappa_i} \sum_{j=1}^{m} \xi_{i,j} \omega_j x_j$$

za $i = 1, \ldots, k$. Pomožne uteži, ki jih potrebujemo za nadalno računanje, smo določili po formuli $\omega(C_i) = \sum_{j=1}^{m} \xi_{i,j}\omega_j$. Potem smo vsakemu središču določili matriko

$$V_i = \sum_{j=1}^{m} \frac{\xi_{i,j}^o \omega_j}{\omega(C_i^o)}(x_j - c(C_i^o))(x_j - c(C_i^o))^T.$$

S pomočjo singularnega razcepa (funkcija svd) dobimo ortogonalno matriko Q in $\sigma_1^{(i)}, \sigma_2^{(i)} > 0$, tako da je $V_i = Q * diag(\sigma_1^{(i)}, \sigma_2^{(i)}) * Q^T$. Sedaj definiramo $M_i = Q * diag((\sigma_1^{(i)})^{-1}, (\sigma_2^{(i)})^{-1}) * Q^T$ in sledi norma podana s predpisom

$$\|x\|_{M_i} = \sqrt{x^T M_i x}.$$

4.3 Linearni program

V datoteki "eksperiment_volitve/LinearniProgram_volitve.R" je spremenjen linearni program, ki smo ga uporabljali za ta eksperiment. V linearnem programu, ki je opisan v uvodu, imamo stroge enakosti za pogoje, ki določajo velikosti gruč. Zaradi fleksibilnosti pa bodo željene velikosti enot lahko večje, zato pri teh pogojih spremenimo enačaje v "<=" (vrstica 46). V Evklidskem primeru so norme za vsa središča iste, pri eliptičnem primeru pa iz definicije vidimo, da je za vsako središče definirana svoja norma. Zato za ta linearni program določimo vektor, ki izračuna vse norme (dolžine je kolikor je središč), nato pa uporabi za vsako središče ustrezno normo (vrstica 21).

4.4 Izvajanje eksperimenta in rezultati

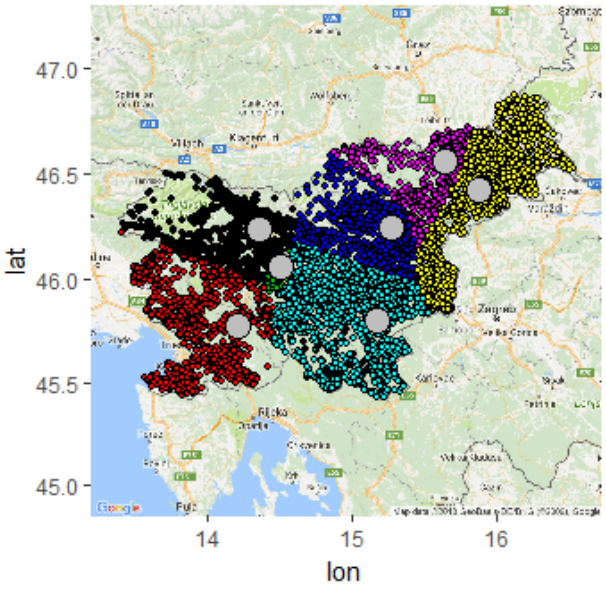
Koda eksperimenta je v datoteki "eksperiment_volitve/volitve.R". Ponovili smo ga za tri primere:

- Ljubljana kot eno središče in uporabljena Evklidska norma,
- Ljubljana kot dve središči in uporabljena Evklidska norma,
- Ljubljana kot eno središče in uporabljena elipsoidna norma.

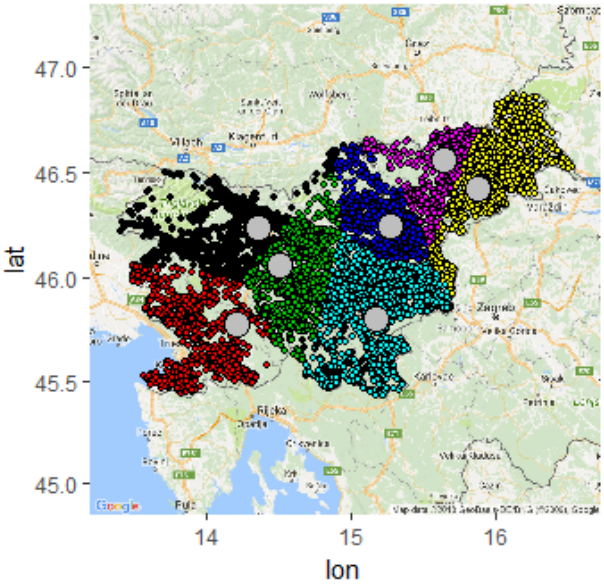
V kodi lahko s spreminjanjem p-ja spremenimo Evklidsko normo v p-to normo (vrstici 76 in 228). Tako dobimo drugačne oblike enot. V naših primerih smo uporabljali $p = 2$.

Spodaj so grafični prikazi rezultatov: prva slika je iz spleta in predstavlja volilne enote kot so v resnici, pri drugi sliki je v Ljubljani ena enota in uporabljena je Evklidska norma, tretja slika predstavlja v Ljubljani dve enoti in Evklidsko normo, četrta slika pa prikazuje spet le eno enoto v Ljubljani in uporablja zgoraj definirano elipsoidno normo.

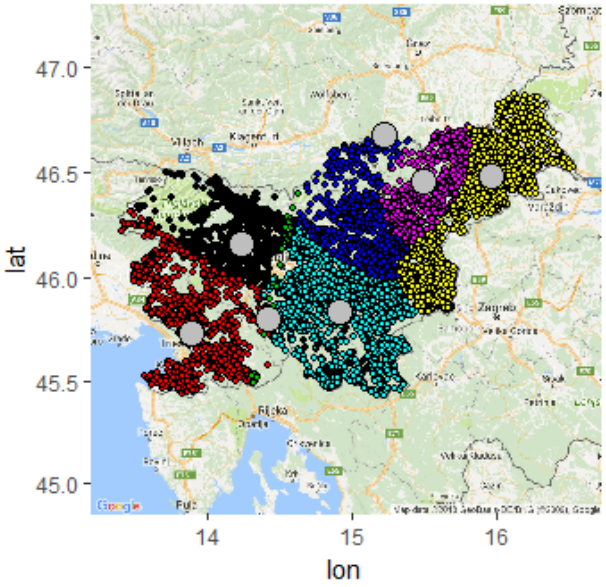
1.



2.



3.



4.

Rezultati so zelo podobni realnim podatkom. Največ odstopanj je v osredni Sloveniji in iz prehoda iz osrednje v vzhodno. Pomembno se je zavedati, da smo pri tem eksperimentu uporabljali zračno razdaljo med kraji. Ta ne upošteva geografskih dejavnikov, ki imajo vlogo predvsem v severozahodni Sloveniji. Pri določanju volilnih enot pa je vlada najbrž upoštevala tudi take dejavnike in je enote določila tudi na podlagi dostopnosti kraja do središča.

5. Eksperiment: Grupiranje stanovanj razpoložljivih za prodajo

5.1 Opis problema

V tem eksperimentu je naš cilj stanovanja, ki se prodajajo v okolici ljubljane, razdeliti v posamezne gruče glede na njihove karakteristike. To smo želeli storiti tako, da bodo imela stanovanja v posameznih gručah podobne karakteristike. Ker nismo imeli implicitnih realnih razdalj, niti prostora v katerem se podatki nahajajo, smo norme želeli "smiselno" definirati tako, da bodo postale kriterij podobnosti med stanovanji. A začetku se torej sprašujemo, ali pri reševanju takšnega problema z zgornjim algoritmom dobimo "smiselne" rezultate? Drugo vprašanje, ki se je med konstrukcijo problema pokazalo kot izredno smiselno pa je, kako izbira izbirnih parametrov (norm ter centrov) vplina na rezultate gnezdenja?

VHODNI PODATKI:

Uvozili smo podatke o stanovanjih, ki se prodajajo v Ljubljani ter njeni okolici. Za vsako stanovanje poznamo naslednje lastnosti:

- id (indeks),
- ime,
- tip (npr. soba, garsonijera),
- področje (v katerem delu Ljubljane se nagaja. Možnosti so: Center, Vič-Rudnik, Šiška,Bežigrad, Moste-Polje),
- nadstropje (v katerem se stanovanje nahaja),
- leto (izgradnje),
- velikost,
- ceno.

Tabeli s temi podatki se nahajata v datotekah z imeni "eksperimentStanovanja/stanovanja.csv", "eksperimentStanovanja/tipi.csv"

MOTIVACIJA: Ker je vseh oglasov za prodajo stanovanja, ki smo jih uvozili 1310 ter ima vsako stanovanje šest različnih karakteristik (,če ne štejemo imena ter indeksa) se zdi, za boljšo preglednost, smiselno njihovo grupiranje v gruče s "podobnimi" lastnostmi. Tako npr. kot iskalec stanovanja lahko pregledamo le stanovanja iz gruč, ki imajo nam sprejemljive karakteristike.

5.2 Konstruiranje problema

Točke, ki jih grupiramo v tem eksperimentu predstavljajo stanovanja. Za lažje modeliranje problema, smo se najprej osredotočili le na podatke o ceni ter velikosti stanovanja, nato smo dodali še podatek o področju Ljubljane v katerem se ta nahaja.

Na koncu se torej točke nahajajo v tridimenzionalnem prostoru. Cena ter velikost stanovanja napenjata dve osi tega prostora. Tretja os je diskretna, predstavlja področje Ljubljane na katerem se stanovanje nahaja in lahko zavzema 5 različnih možnosti: Center, Vič-Rudnik, Šiška,Bežigrad, Moste-Polje.

Posamezno stanovanje torej predstavlja točka oblike (cena stanovanja, velikost stanovanja, področje), konkreten primer bi bil npr. (120 000, 61, Šiška).

Vsi do sedaj definirani parametri so bili v eksperimentu na nek način določeni že z izbiro podatkov. Ostala nam je še konstrukcija izbirnih parametrov, to so centri gruč, ter razdalje. Izbire teh parametrov podatki direktno ne določajo, temveč smo jih poizkušali čim bolj "smiselno"" izbrati sami. Ker je težko določiti, kaj pomeni njihova smiselna izbira smo program večkrat zahnali za različne izbire centov ter norm in spremljali, kako se glede na izbiro spreminjajo izhodni rezultati.

NORME: Na oseh, ki predstavljata ceno ter velikost stanovanja, smo izmenično preizkusili drugo, neskončno in p-to normo. Vrednosti koordinat smo pred tem normirali s povprečno ceno, ter velikostjo (, ker je najdražje stanovanje precej dražje od povprečnih cen, se normiranje z maksimalni vrednostjo ni izkazalo za smiselno). Medtem ko smo na osi, ki predstavlja področje normo definirali tako kot prikazuje spodnje psevdo koda:

```
norma.podrocje <- konstanta
if(identical(podrocje.tocke1, podrocje.cocke2)){
  norma.podrocje <- 0
}
```

Torej, če imata dve točki enako področje je njuna razdalja v tej koordinati enaka 0, sicer je enaka neki konstanti, karete vrednost smo tekom eksperimenta poljubno spreminjali.

Na koncu smo konstruirali dva primera, z različnimi izbirnimi parametri, ki sta shranjena v datotekah "eksperimentStanovanja/primer1.R", "eksperimentStanovanja/primer2.R".

5.3 Analiza rezultatov in zaključek

GLAVNO ANALIZO REZULTATOV TEGA EKSPEREMENTA Z GRAFI SI LAHKO POGLEDATE V AMPLIKACIJI IZ DATOTEKE: "stanovanjaShiny.R".

Izkazalo se je, da algoritem dobro deluje tudi za takšne vrste problemov. Najtežji del je predstavljala smiselna izbira izbirnih paramerov. Končni rezultati so bili najbolj odvisni od izbire centrov gnezd. Pri reševanju dvodimenzionalnega problema ta odvisnost ni bila tako zelo velika. Večinoma so točke kljub spreminjanju centrov ostale podobno razdeljene v gnezda. Pri reševanju tridimenzionalnega problema, pa so se se že z manjšim spreminjanjem centrov, pri fiksnih ostalih parametrih gnezda začela precej hitro spreminjati. Vpliv izbire norm na končno razvrstitev nima tako zelo velikega učinka. Največje spremembe smo lahko opazili, ko smo povečevali vpliv razlike v zgolj eni koordinati na končno razdaljo. To smo storili tako, da smo razdaljo po eni koordinati pomnožili ali potencirali, ostali dve pa pustili pri miru. V tem primeru je ta lastnost postala dominantna in gnezda so se nekoliko spremenila.

6. Vir

1. Breiden A., Gritzmann P., Klemm F. 2017. Constrained clustering via diagrams: A unified theory and its applications to electoral district design. *European Journal of Operational Research.* 263, 18-34