Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Peter Us

Uporaba predznanja pri vizualizaciji visokodimenzionalnih podatkov

diplomsko delo

univerzitetni študijski program prve stopnje
računalništvo in informatika

Mentor: izr. prof. dr. Janez Demšar

Ljubljana 2014

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Pri vizualizaciji podatkov z velikim številom dimenzij naletimo, tako kot pri sestavljanju napovednih modelov in iskanju vzorcev, na problem prekletstva dimenzionalnosti: zaradi velikega števila možnih kombinacij spremenljivk (atributov), lahko v podatkih odkrijemo veliko število izrazitih, vendar v resnici naključnih vzorcev. Tudi rešitev problema je morda podobna: uporaba predznanja.

Raziščite, kako z uporabo predznanja uspešneje iskati uporabne vizualizacije visokodimenzionalnih podatkov. Pri tem se osredotočite predvsem na podatke s področja genetike, za katere je na voljo veliko primernih podatkovnih zbirk in pripadajočega predznanja.

Izjava o avtorstvu diplomskega dela

Spodaj podpisani Peter Us, z vpisno številko **63110345**, sem avtor diplomskega dela z naslovom:

Uporaba predznanja pri vizualizaciji visokodimenzionalnih podatkov

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom izr. prof. dr. Janeza Demšarja,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 2. septembra 2014 Podpis avtorja:

Kazalo

Povzetek

Vizualizacije podatkov so lahko odličen pristop k analizi in odkrivanju novega znanja o podatkih. O podatkih imamo velikokrat na voljo tudi določeno predznanje, ki se ga izplača uporabiti pri iskanju koristnih vizualizacij. V diplomski nalogi raziskujemo vpliv genskih skupin pridobljenih iz baze MSigDB na kakovost vizualizacij mikromrež DNA. Ukvarjamo se z vprašanjem, ali lahko skupine genov uporabimo za pridobitev boljših vizualizacij. V prvem poglavju predstavimo vizualizacijo podatkov na splošno in metodi razsevni diagram ter Radviz, ki se uporabljata tekom dela. Na koncu poglavja sledi predstavitev metode VizRank, ki nam omogoča avtomatsko ocenjevanje in iskanje kakovostnih vizualizacij. Drugo poglavje opiše podatke mikromrež in podatke o genskih skupinah. Sledi predstavitev našega eksperimentalnega dela, ki je razdeljeno v tri sklope. V vsakem sklopu predstavimo idejo, postopek, rešitve in analizo rezultatov eksperimentiranja. Zaključimo s predstavitvijo glavnih ugotovitev.

Ključne besede: vizualizacije, VizRank, Radviz, razsevni diagram, mikromreže DNA.

Abstract

Data visualisation can be an extremely efficient way of analyzing and discovering new knowledge from data. More often than not, we have background knowledge about the data, which we can then use to find meaningful and useful visualizations. This thesis examines the influence of gene sets gained from MSigDB upon the quality of visualizations of DNA microarrays. We hypothesize, that we can use gene sets to gain better and clearer visualizations. In the first chapter we explain what data visualization is, and introduce our working methods. At the end of the first chapter we present a method called VizRank, which allows us to automatically find and rate quality visualizations. This is followed by the second chapter, in which we describe the DNA microarray data and the data of gene sets. In the last part we present our experiential work, which is split into three sections. In each individual section we present the idea, procedure, solution and analysis of the results of experimentation.

Keywords: visualisations, VizRank, Radviz, scatter plot, DNA microarrays.

Poglavje 1

Uvod

V našem delu raziskujemo uporabo predznanja pri vizualizaciji visokodimenzionalnih podatkov. Vizualizacije podatkov so lahko odličen pristop pri odkrivanju novega znanja o podatkih in njihovi analizi. S pomočjo vizualizacij lahko dobimo v podatke drugačen vpogled in lažje opazimo določene lastnosti, ki bi jih v podatkih v tekstovni obliki opazili težje.

Kot opisano kasneje (poglavje 2), vse vizualizacije niso uporabne. Pregledovanje vizualizacij je zamudno delo, zato lahko uporabimo metodo VizRank. Metoda nam avtomatsko poišče in oceni zanimive vizualizacije. Tekom celotnega eksperminetiranja v delu uporabljamo metodo VizRank, zato je podrobno opisana v podpoglavju 2.5.

O podatkih imamo velikokrat na voljo tudi predznanje. Predznanje predstavljajo dodatne informacije, ki jih imamo o podatkih, ki jih želimo vizualizirati. V našem delu se ukvarjamo z vizualizacijami mikromrež DNA, kjer imamo izmerjene izraženosti več tisoč genov (podrobneje opisani v 3.2). Naše predznanje o genih so genske skupine, ki smo jih opisali v podpoglavju 3.3. Za vizualizacijske podatke in predznanje bi lakho izbrali tudi druge vrste podatkov, vendar so genski podatki najprimernejši, saj so podatki visokodimenzionalni, prostodostopni, primernih velikosti in se pogosto uporabljajo za poskuse različnih metod podatkovnega rudarjenja.

V drugem delu diplomskega dela želimo s pomočjo različnih poskusov ugotoviti, kakšen vpliv imajo genske skupine na kakovost dobljenih vizualizacij mikromrež DNA. Glavni vprašanje dela je, ali bi se genske skupine dalo uporabiti za hitrejše

in boljše pridobivanje dobrih vizualizacij mikromrež.

Eksperimentiranje smo razdelili v tri večje dele. V vsakem delu na drugačen način iščemo vpliv genskih skupin na dobljene vizualizacije. Ideja, postopek in analiza rezultatov vsakega poskusa so predstavljeni vsak v svojem poglavju.

Čeprav poskuse delamo na genskih podatkih, je vredno omeniti, da so ideje eksperimentov splošne in bi jih bilo možno uporabiti tudi za druge vrste podatkov, kjer imamo na voljo podobno predznanje, kot ga imamo v našem delu.

Poglavje 2

Vizualizacije podatkov

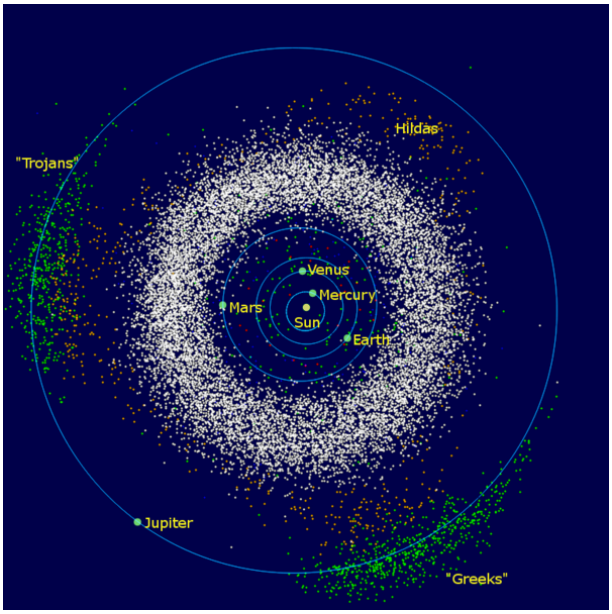
2.1 Splošno

Pojem vizualizacija podatkov zajema izdelovanje in preučevanje podatkov predstavljenih v slikovni obliki. Ljudje pogosto podatke bolje razumemo, če so ti predstavljeni s sliko. Med podatki hitro opazimo povezave, zanimivosti, vzorce in podobno. Take lastnosti pri podatkih predstavljenih v numerični ali tekstovni obliki opazimo težje.

Vizualizacije podatkov lahko delimo na dve večji področji: znanstvene vizualizacije (scientific visualizations) [14] in vizualizacije informacij (information visualization) [5].

2.1.1 Znanstvene vizualizacije

Pod znanstvene vizualizacije spada izgradnja grafičnih modelov, ki jih ponavadi zgradimo iz že poznanih (izmerjenih/pridobljenih) podatkov. Te modele največkrat uporabljamo za simulacijo. Za boljše razumevanje naštejmo nekaj primerov: simulacija valovanja, vizualizacija osončja ter gibanja planetov (prikazana na sliki 2.1), vizualizacija geografskih zemljevidov [10] in podobni primeri grafičnih modelov.

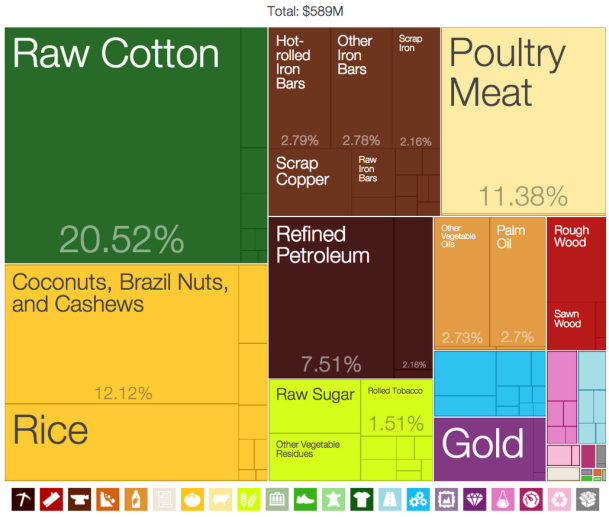


Slika 2.1: Primer znanstvene vizualizacije. Vir: wikipedia

2.1.2 Vizualizacija informacij

Druga večja skupina vizualizacij je vizualizacija informacij. Vizualizacija informacij je pomembno področje podatkovnega rudarjenja. Ukvarja se s prikazom abstraktnih podatkov, pri katerih relacij med njimi ponavadi ne poznamo. Pogosto se uporablja z namenom, da med podatki poiščemo še prej nepoznane relacije in tako pridobimo novo znanje o podatkih. Primeri vizualizacij informacij so: slikovni prikaz prodanih izdelkov v nekem časovnem obdobju, vizualizacije izraženosti genov, prikaz izvoza izdelkov (slika 2.2) in podobni. V diplomskem delu se ukvarjamo izključno samo z vizualizacijami informacij.

Pri vizualizaciji informacij je pomembno, da podatke predstavimo tako, da opazovalec v sliki hitro opazi zanimivosti in vzorce. Za prikazovanje podatkov obstaja veliko število različnih metod [19]. Izbira metode je odvisna od lastnosti podatkov, ki jih predstavljamo (količina podatkov, diskretnost atributov, količina atributov ipd.) in lastnosti, ki jih z vizualizacijo želimo predstaviti. V diplomskem delu uporabljamo dve metodi: razsevni diagram in metodo Radviz. Obe metodi bosta sedaj podrobneje predstavljeni, saj se uporabljata tekom celotnega



Slika 2.2: Vizualizacija izvoza izdelkov zahodno afriške države Benin v letu 2009. Vir: wikipedia

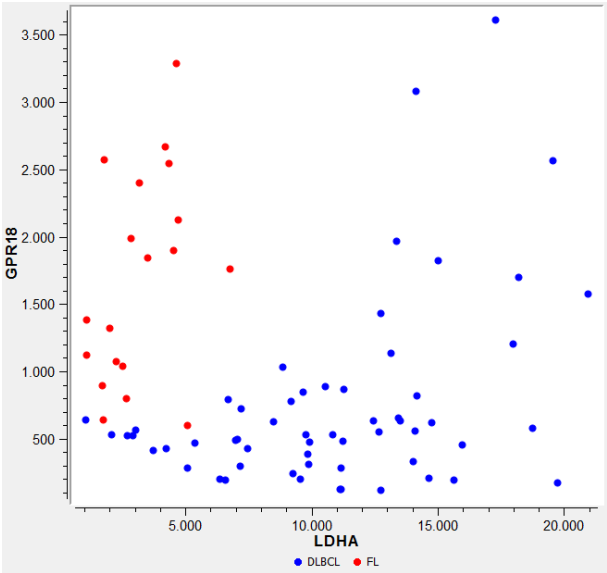
diplomskega dela.

2.2 Razsevni diagram

Razsevni diagram [23] (graf raztrosa, scatter plot) je ena izmed najbolj poznanih in uporabljenih metod za vizualizacijo podatkov. Metoda uporablja kartezijske koordinate za prikaz vrednosti dveh atributov v dvo-dimenzionalnem prostoru. Podatki so prikazani kot zbirka točk, katere lega je odvisna od vrednosti njenih dveh atributov. Tretjo spremenljivko lahko predstavimo z barvo ali obliko točke, ki predstavlja podatek.

V diplomskem delu uporabljamo razsevne diagrame za prikaz podatkov z dvema atributoma (slika 2.3). Barvo točk uporabimo za predstavitev razreda, kateremu primer pripada. Vredno je omeniti, da lahko razsevne diagrame razširimo na različne načine in tako predstavimo še več atributov. Nekaj primerov je opisanih v članku [7].

Kot omenjeno je najpogostejša uporaba razsevnih diagramov za primere z dvema atributoma. Velikokrat (tudi v diplomskem delu) želimo v vizualizaciji uporabiti več atributov. Za vizualizacije z več atributi je bolj primerna metoda



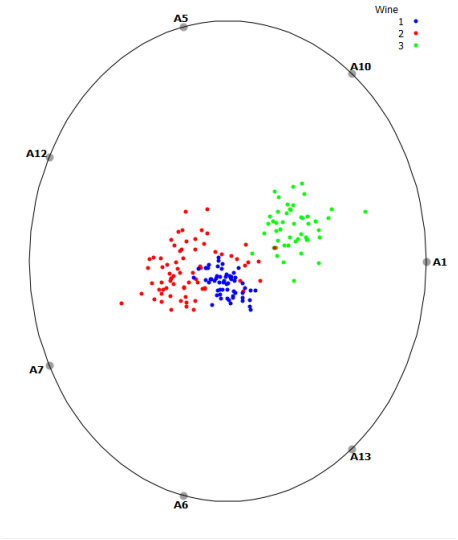
Slika 2.3: Primer vizualizacije izraženosti genov z razsevnim diagramom.

Radviz, ki je predstavljena v naslednjem poglavju.

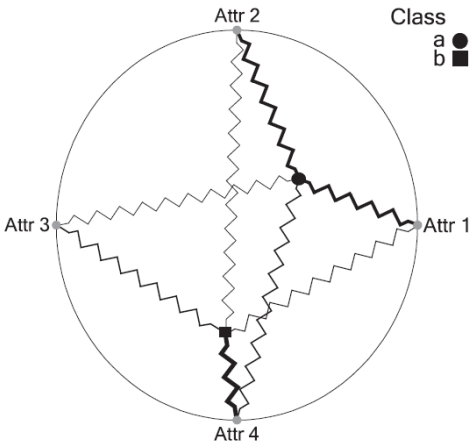
2.3 Metoda Radviz

Radviz [11][4][9] (radial visualisation) je nelinearna vizualizacijska metoda. Omogoča nam izdelavo preglednih vizualizacij podatkov z več kot dvema atributoma. Primer Radviz vizualizacije je prikazan na sliki 2.4. Kot je razvidno iz slike, je vsak podatek predstavljen kot točka znotraj krožnice. Vizualizacijski atributi so enakomerno razporejeni na krožnici in prav tako predstavljeni kot točke.

Izračun položaja točke znotraj krožnice je najlažje razložiti z analogijo z vzmetmi. Predstavljajte si vzmeti, ki so na eni strani priključene na točko, ki predstavlja podatek, na drugi pa na atribut, ki leži na krožnici. V pomoč služi slika 2.5. Položaj točke je sedaj odvisen od sil, s katerimi priključene vzmeti delujejo na točko. Za vzmeti velja Hookov zakon, zato je sila, s katero deluje vzmet na točko, odvisna od koeficienta vzmeti. Koeficient naših navideznih vzmeti je premo sorazmeren z vrednostjo atributa na katerega je vzmet povezana. Večjo kot ima podatek vrednost pri nekem atributu, večja je sila vzmeti, s katero vleče podatek. Podatek tako



Slika 2.4: Primer vizualizacije podatkov s sedmimi atributi z metodo Radviz.



Attr 1	Attr 2	Attr 3	Attr 4	Class
0.75	0.75	0.1	0.3	a
0.2	0.1	0.3	0.95	b

Slika 2.5: Primer vizualizacije Radviz z do risanimi navideznimi vzmetmi. Vir: [22], stran: 86

leži bližje atributom, pri katerih ima visoko vrednost, saj ga tiste vzmeti vlečejo močneje.

Pred vizualizacijo so vrednosti atributov normalizirane na vrednosti med 0 in 1, da imajo vsi atributi enak vpliv. Opazimo, da je poleg izbire atributov pomembna tudi njihova razporeditev atributov na krožnici, saj s spremembo vrstnega reda atributov dobimo nove, drugačne vizualizacije. Z izborom n atributov tako dobimo $\frac{(n-1)!}{2}$ različnih vizualizacij. Čeprav je z metodo Radviz možno upodobiti poljubno število atributov v vizualizaciji, jih več kot 10 ponavadi ne, saj postanejo vizualizacije nepregledne [16].

Tako kot pri razsevnih diagramih lahko tudi pri metodi Radviz točke obarvamo in s tem predstavimo še razred, v katerega primer spada.

2.4 Zanimivost vizualizacij

Kot smo že omenili, je glavni namen vizualizacije informacij pridobivanje novega znanja o podatkih. V grafu lahko opazimo gruče točk, ki imajo podobne lastnosti in se pojavijo skupaj, lahko vidimo vzorce, preko katerih odkrijemo relacije med podatki, lahko opazimo osamljene točke in podobno. Takim vizualizacijam pravimo, da so zanimive. V njih opazimo nekaj nenaključnega. Naloga opazovalca je, da razišče, zakaj se v grafu pojavijo te zanimivosti, in preko tega pridobi novo znanje o podatkih.

Po drugi strani je prav tako možno, da v vizualizaciji ne vidimo ničesar uporabnega. Primer tega bi bil graf, ki bi ga želeli uporabljati za klasifikacijo novih primerov, a bi izgledal, kot da so vse točke naključno nametane. S pomočjo takega grafa bi težko klasificirali nove primere, zato bi bil za nas neuporaben.

V praksi se želimo izogniti pregledovanju neuporabnih grafov, saj je pregledovanje grafov zamudno opravilo. V nadaljevanju se ukvarjamo z ocenjevanjem vizualizacij in predstavimo metodo, ki omogoča avtomatsko iskanje zanimivih vizualizacij.

2.5 Metoda VizRank

2.5.1 Ocenjevanje vizualizacij

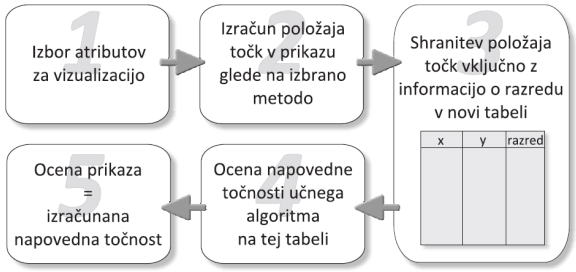
Kot omenjeno v poglavju 2.4, vse vizualizacije niso koristne. Število različnih vizualizacij za neko vrsto podatkov je odvisno od števila atributov, ki jih želimo upodobiti. Če imamo podatke z x atributi in nas zanimajo vizualizacije z y uporabljenimi atributi, je možnih izborov atributov $\binom{x}{y}$. Pri metodah, kjer je pomembna postavitev atributov (primer Radviz), lahko za vsak izbor atributov s preurejanjem pridobimo še $\frac{(y-1)!}{2}$ različnih vizualizacij [16]. Število različnih možnih vizualizacij tako narašča zelo hitro.

Različnih možnih vizualizacij je že pri majhnem številu atributov občutno preveč, da bi vse pregledali ročno. Prav tako je pregledovanje naključnih vizualizacij zamudno opravilo. Zaradi omenjenih razlogov si pri iskanju primernih vizualizacij pomagamo z metodo VizRank [17]. Metoda nam oceni vizualizacije, ne da bi jih sami morali ročno pregledovati. Rezultat metode je spisek vizualizacij urejenih po ocenah. Ocena predstavlja zanimivost vizualizacije. Večja ko je ocena vizualizacije, večja je verjetnost, da bo opazovalec v vizualizaciji opazil vzorec. S pomočjo metode VizRank tako ročno pregledamo samo dobro ocenjene vizualizacije in ne izgubljamo časa s pregledovanjem tistih, za katere lahko že metoda določi, da so neuporabne (podatki izgledajo naključno nametani, ni opaznih vzorcev in podobno). V nadaljevanju sledi podrobnejši opis delovanja metode VizRank.

2.5.2 Opis metode VizRank

Metoda VizRank je primerna za iskanje zanimivih vizualizacij pri vizualizacijskih metodah, kjer so primeri predstavljeni kot točke v prostoru, njihov položaj pa je odvisen od vrednosti atributov, ki jih uporabimo pri vizualizaciji. Vizualizacijske metode teh vrst so razsevni diagram, Radviz, Polyviz [9] in splošne linearne projekcije. Ker v diplomskem delu uporabljamo razsevne diagrame in metodo Radviz, je primerna tudi uporaba metode VizRank.

Metodi VizRank podamo podatke, ki morajo biti take oblike, da primeri pripadajo različnim razredom (class-labeled data), ta pa nam vrne najbolj zanimive projekcije, narejene iz podanih podatkov. To naredi s ponavljanjem petih korakov,



Slika 2.6: Postopek delovanja metode VizRank. Vir: [16], stran: 54

predstavljenih v sliki 2.6. Koraki so podrobneje opisani v naslednjih podpoglavjih. Metoda najprej izbere podmnožico atributov za vizualizacijo. Ko ima izbrane atribute, metoda izračuna položaj točk v projekciji glede na izbrano vizualizacijsko metodo. Izračunane položaje točk shrani v tabelo in zraven doda še razred, ki mu podatek pripada. V naslednjem koraku izračuna oceno napovedne točnosti učnega algoritma na izdelani tabeli. V zadnjem koraku metoda poda vizualizaciji oceno in ponovi postopek z naslednjo vizualizacijo.

2.5.3 Izbor atributov

Prvi korak metode je izbor podmnožice atributov za vizualizacijo. Kot smo opisali v poglavju 2.5.1, število projekcij narašča eksponentno s številom izbranih atributov za vizualizacijo. Zaradi tako hitrega naraščanja števila možnih vizualizacij ne more metoda VizRank niti pri običajni velikosti podatkov pregledati vseh možnih vizualizacij. Procent zanimivih vizualizacij je tako majhen, da če bi izbirali atribute naključno, bi trajalo več ur, preden bi metoda našla kakšno zanimivo rešitev [16]. Zaradi omenjenih težav VizRank uporablja dve hevristiki pri izbiri atributov za vizualizacijo. Z uporabo hevristik hitreje pridemo do prostora zanimivih projekcij kot pri naključnem iskanju. Prav tako se izognemo preiskovanju velike večine prostora projekcij in še vedno z visoko verjetnostjo najdemo najzanimivejše vizualizacije. Prva hevristika je splošna, namenjena vsem točkovnim vizualizacijskim metodam, druga pa je izboljšava splošne, za primere vizualizacije na metodi Radviz.

Opis uporabljene hevristike

Pri splošni hevristiki metoda najprej oceni "kvaliteto" vseh atributov, ki nastopajo v podatkih. To stori z izračunom informativnosti atributov z uporabo metode Relief [15]. Pri postopku se opira na dejstvo, da dobre projekcije sestavljajo atributi, ki so informativni pri ločevanju med različnimi razredi. Na drugi strani neinformativni atributi projekcijam ne prinesejo nič. Oceno projekcije sedaj sestavi z vsoto atributov, ki so v projekciji uporabljeni. Tako izračuna ocene vseh projekcij in jih uredi od najvišje do najnižje. Višja ko je ocena projekcije, bolj verjetno je, da bo projekcija zanimiva. Da se metoda pri izbiri atributov izogne determinizmu (škodljivost determinizma pri izbiri je opisana na strani 59 v [16]), vključi pri postopku izbiranja še naključnost. Projekcije sedaj izbira naključno, a večjo prednost daje projekcijam z dobrimi ocenami. Opisana hevristika je primerna za vse vizualizacijske metode, ki jih podpira VizRank.

Za metodo Radviz se da postopek izbire prilagoditi in tako še izboljšati ter pohitriti ocenjevanje projekcij. Podroben opis obeh hevristik je v [16] v poglavju 3.2.

Izračun točk in zapis v tabelo

Drugi korak metode VizRank je izračun položaja točk v prikazu glede na izbrano metodo. Metoda za vsak primer v podatkih izračuna položaj koordinat x in y. Podatke shrani v tabelo in doda razred, v katerega podatek spada. Po končanem postopku metoda uporablja samo še podatke iz dobljene tabele. Pri ocenjevanju metoda torej "vidi" le tiste podatke, katere vidi človek, ko opazuje vizualizacijo.

Ocena napovedne točnosti vizualizacije

Dobljeno vizualizacijo mora metoda VizRank oceniti čim bolj podobno človeški oceni vizualizacije. Zaradi tega je po empiričnih raziskavah [16] najbolj primerna izbira učnega algoritma metoda k-najbližjih sosedov (k-NN, k-nearest neighbours). k-NN deluje tako, da napove vrednost razreda za nov primer glede na to, kakšne razrede ima njegovih k najbližjih sosedov. Vsak od k sosedov glasuje svoj razred, njihov glas pa je utežen z razdaljo do primera, ki ga ocenjujemo. Metoda primer nato klasificira v razred, za katerega je večja verjetnost, da vanj spada glede na

dobljene informacije sosedov. Po raziskavi je za izbiro števila k najbolj primerna formula $k = \frac{N}{c}$, kjer je N število podatkov in c število razredov v podatkih.

Potrebno je izbrati tudi cenilno funkcijo. Ker je k-NN verjetnostni klasifikator, lahko za oceno celotne vizualizacije izračunamo povprečno vrednost, ki jo klasifikator določi pravilnim razredom. Povprečna vrednost se je po empiričnih raziskavah izkazala za najprimernejšo [16].

Za analizo celotne projekcije metoda VizRank uporabi prečno preverjanje "izloči enega" (leave one out cross validation) z učnim algoritmom k-NN in cenilno funkcijo povprečnih verjetnosti. Dobljeno oceno shrani in vmesti vizualizacijo v seznam ocenjenih vizualizacij, ki so urejeni po oceni. Uporabnik lahko tako kadarkoli metodo prekine in bo dobil spisek do takrat najbolje ocenjenih vizualizacij.

Poglavje 3

Genski podatki

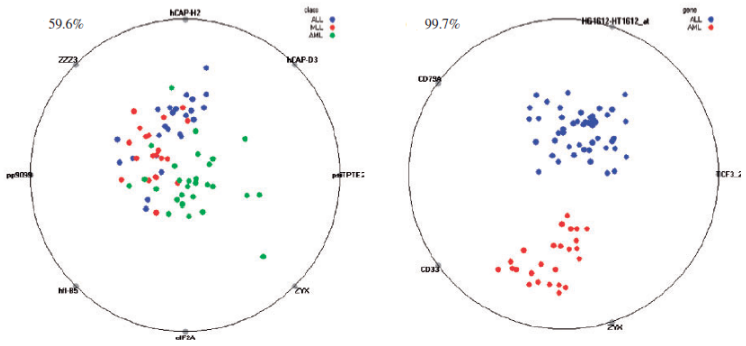
3.1 Splošno

V diplomskem delu se ukvarjamo z vizualizacijo genskih podatkov. Opisali smo že vse, kar je potrebno vedeti o vizualizacijah, manjka nam samo še opis podatkov, ki jih vizualiziramo. Na razpolago imamo podatke dveh vrst. Prvi so genski podatki mikromrež DNA, ki jih vizualiziramo. Namen in postopek vizualizacije teh vrst podatkov je opisan v članku [18]. Podatke vizualiziramo z razsevnim diagramom, kadar za atributa uporabimo dva gena, in z metodo Radviz, kadar uporabimo več atributov.

Druga vrsta podatkov so podatki o genskih skupinah. Ti podatki predstavljajo naše predznanje, ki ga imamo o podatkih mikromrež DNA. Sledi podroben opis obeh vrst uporabljenih podatkov.

3.2 Mikromreže DNA

Podatke, ki jih v diplomskem delu upodabljamo v slikovni obliki, so podatkovne zbirke o različnih vrstah rakavih obolenj. Podatki so pridobljeni z uporabo tehnologije mikromrež DNA in so javno dostopni. Tehnologija mikromrež DNA je eno od najučinkovitejših metod za študije izražanja genoma, saj daje zmožnost merjenja izraženosti več tisočih genov naenkrat [13].



Slika 3.1: Primer slabe in dobre Radviz vizualizacije genskih podatkov. Leva je z metodo VizRank dobila oceno 59.6% in ni koristna, desna pa dobro loči med obema razredoma in je dobila oceno 99.7%. Vir: [18]

3.2.1 Struktura podatkov

V datotekah mikromrež DNA imamo naštete paciente. Za vsakega pacienta imamo izmerjene izraženosti več tisoč genov. Poleg izmerjene izraženosti genov, imamo za vsakega pacienta podan tudi razred, v katerega spada. Razred v večini datotek predstavlja diagnozo pacienta (vrsta tumorja). Značilnost podatkov je, da vsebujejo veliko število atributov (genov, več tisoč) in le majhno število primerov (pacientov, 10 do 100).

Podatke v diplomskem delu vizualiziramo tako, da izmed vseh genov izberemo nekaj genov, ki jih uporabimo kot atribute pri vizualizaciji z razsevnim diagramom ali metodo Radviz. Razrede, v katere primeri spadajo, prikažemo z barvo točk. Zanimive vizualizacije so tiste, ki dobro ločijo med različnimi razredi in jih lahko uporabimo za klasifikacijo novih primerov. Slabe vizualizacije pa so tiste, ki ne ločijo dobro med primeri različnih razredov pacientov (primer slabe in dobre vizualizacije na sliki 3.1).

Poskuse smo opravljali na šestih različnih datotekah mikromrež DNA. Opis tehničnih lastnosti podatkov prikazuje tabela 3.1. Sledi podrobnejši opis vsake od datotek.

DLBLC vsebuje klasifikacijske podatke pacientov. Pacienti spadajo v dva razreda. Prvi imajo difuzni velikocelični limfom B (diffuse large B-cell lymphoma

Datoteka	Velikost vzorca (št. pacientov)	Št. atributov (genov)	Št. razredov
DLBCL.tab	77	7070	2
GDS971.tab	23	9695	2
GDS1059.tab	54	6200	2
GDS1667.tab	36	34700	3
GDS2609.tab	23	34700	2
GSE3726.tab	52	14166	2

Tabela 3.1: Tabela prikazuje lastnosti datotek uporabljenih za vizualizacijo.

- DLBCL), drugi pa folikularno limfomo (follicular lymphoma - FL). Zgodovinski podatki kažejo, da se FL pogosto v kasnejši fazi spremeni v DLBLC. Podatki so podrobneje opisani v članku [20].

GDS971 vsebuje podatke o pacientih z dvema različnima rakavima sarkomoma. Prvi imajo sarkom RMS (rhabdomyosarcoma), drugi pa Ewigov sarkom (Ewig's sarcoma - EWS). Podatki so podrobneje opisani v članku [2].

GDS1059 vsebuje podatke z analizo mononuklearnih (mononuclear) celic 54 pacientov zdravljenih s kemoterapijo. Pacienti so mlajši od 15 let z akutno mieloično levkemijo AML (acute myeloid leukemia). Razred pacienta nam predstavlja podatek o tem, ali se je bolezen v naslednjih letih ponovila ali ne. Podrobneje predstavljeni podatki v članku [24].

GDS1667 vsebuje podatke pacientov z karcinomom na glavi in vratu (head and neck squamous cell carcinoma - HNSCC) ter človeškim papiloma virusom (human papilloma virus - HPV). Podatki prikazujejo učinek virusa HPV na obolenje s HNSCC. Podrobno so opisani v članku [21].

GDS2609 vsebuje podatke o izraženosti genov pri pacientih z rakom na debelem črevesu, kateri pri prednikih nimajo te bolezni. Podrobneje so opisani v [12].

Datoteka	Število skupin
c2.cp.v3.0.symbols	1320
c5.bp.v3.0.symbols	825
c5.mf.v3.0.symbols	396

Tabela 3.2: Tabela prikazuje lastnosti datotek o skupinah genov, uporabljenih kot predznanje podatkov mikromrež DNA.

GSE3726 vsebuje podatke o izraženosti genov pri pacientkah z rakom na dojkah ali rakom na debelem črevesju. Podrobneje so opisani v [6].

3.3 Genske skupine

Druga vrsta podatkov, ki jih imamo na voljo, so genske skupine (gene sets). Skupine imamo podane v datotekah in so v formatu GMT (Gene Matrix Transform file format). Skupine so dobljene iz prosto dostopne baze MSigDB. Služijo nam kot predznanje mikromrežam, ki jih vizualiziramo. Skupine so sestavljene iz raziskav določenih procesov in biološkega znanja. Uporabljamo tri datoteke s podanimi skupinami genov, predstavljene v tabeli 3.2.

Prva datoteka spada v skupine z oznako C2 (currated gene sets), kjer so datoteke sestavljene iz podatkov pridobljenih iz različnih internetnih podatkovnih baz o genih.

Druga in tretja datoteka spadata v skupine C5 (gene ontology gene sets), kjer so skupine sestavljene na podlagi ontologije genov [1].

Kot že omenjeno, so skupine genov naše predznanje o podatkih, ki jih vizualiziramo. V diplomskem delu ugotavljamo, ali lahko z uporabo znanja o genskih skupin pridemo hitreje do dobrih vizualizacij genskih podatkov mikromrež DNA. Želimo ugotoviti, kakšen vpliv imajo skupine na kakovost vizualizacij dobljenih iz podatkov mikromrež.

V naslednjih poglavjih sledi opis izvedenih poskusov in eksperimentiranja na podatkih. Glavni namen našega dela je ugotoviti, ali si lahko pri iskanju dobrih vizualizacij podatkov mikromrež DNA pomagamo s predznanjem o smiselnih skupinah genov.

Poglavje 4

Metode in poskusi

4.1 Uvod

Kot omenjeno, se v diplomskem delu ukvarjamo z vizualizacijami podatkov. Vizualiziramo genske podatke mikromrež DNA opisane v poglavju 3.2. O podatkih imamo tudi predznanje. To so genske skupine, ki so opisane v poglavju 3.3. Podatke vizualiziramo tako, da izmed vseh genov izberemo nekaj teh in jih uporabimo kot atribute pri vizualizacijah z razsevnim diagramom (opisan v poglavju 2.2) ali metodo RadViz (opisana v poglavju 2.3).

4.2 Prvi del eksperimentov

4.2.1 Ideja

Zanima nas, ali lahko z uporabo znanja o smiselnih skupinah genov izboljšamo kakovost vizualizacij. Ideja prvega poskusa, s katerim bi ugotovili vplivanje genskih skupin, je sledeča.

Za vizualizacijo smo naključno izbrali 50 parov genov, kjer se oba gena pojavita skupaj v vsaj eni izmed skupin definiranih v datotekah genskih skupin GMT. Vsak izmed parov predstavlja eno vizualizacijo z razsevnim diagramom. Vsako izmed vizualizacij smo ocenili z metodo VizRank.

Nato smo znova izbrali 50 parov genov. Tokrat smo izbirali samo pare, kjer se gena v nobeni izmed skupin ne pojavita skupaj. Dobili smo novih 50 vizualizacij.

Tudi tokrat smo vsako ocenili z metodo VizRank.

Na koncu smo primerjali ocene obeh skupin parov genov. Naredili smo analizo rezultatov, s katero smo ugotovili, kako pripadnost istim skupinam pri genih vpliva na kakovost dobljenih vizualizacij.

Enak postopek, ki je opisan za pare genov, smo ponovili tudi na trojicah genov.

4.2.2 Potek dela

Za vse poskuse smo uporabljali programski jezik Julia [3]. Edina izjema je uporaba metode VizRank. Ta je vsebovana v programskem paketu Orange [8], ki je modul za programski jezik Python. Poleg samega modula smo uporabljali tudi uporabniški vmesnik programa Orange, ki je namenjen delu z vizualizacijam podatkov.

Branje datotek mikromrež DNA

Prvi korak v našem delu je bilo branje tabelaričnih datotek s podatki mikromrež DNA. Datoteka je v formatu .tab (tab-separated values), kjer so podatki ločeni s tabulatorjem. Za branje smo uporabili odprtokodno knjižnico DataFrames (https://github.com/JuliaStats/DataFrames.jl), ki je namenjena branju in obdelavi tabelaričnih datotek. Knjižnica vsebuje metodo “readtable”, ki ji kot argument podamo separator podatkov. V našem primeru je to tabulator.

```
readtable("GDS971.tab", separator = '\t', header=false);
```

Branje datotek genskih skupin

Naslednji korak je bilo branje datoteke GMT, ki vsebuje podatke o skupinah genov. Vsaka vrstica v datoteki predstavlja eno skupino. Znotraj skupine so podatki ločeni s tabulatorjem. Prvi podatek v vrstici vedno predstavlja ime skupine. Naslednji je URL podrobnejšega opisa skupine, kateremu sledijo našteta imena genov, ki spadajo v to skupino. Za branje datoteke smo napisali svojo metodo.

Listing 4.1: Programska koda za branje datoteke formata .gmt

```
function readGmt(path)
    tab = {};
    open(path, "r") do f
        for line in eachline(f)
            push!(tab, split(line, '\t'));
        end
    end
    return tab;
end
```

Metodi kot parameter podamo pot, vrne pa dvo-dimenzionalno tabelo, ki vsebuje enake podatke kot originalna datoteka. V programu smo tako dobili vse potrebne podatke. Naš naslednji korak je bil izbira genov.

Izbira parov genov

Izbrati smo morali 50 parov genov, ki imajo vsaj eno skupino iz datoteke GMT. V ta namen smo napisali metodo, ki ji kot argument podamo par genov, vrne pa logično vrednost, ali pripadata gena kakšni skupni skupini.

S pomočjo opisanih metod smo lahko izbrali 50 parov genov, ki imajo skupne skupine, in 50 parov genov iz različnih skupin. Za izbiro genov, ki imajo skupne, je bil postopek naslednji:

1. Iz datoteke GMT izberi naključno skupino.
2. Izberi naključna gena iz izbrane skupine.
3. Preveri, če sta oba gena vsebovana v datoteki mikromrež DNA.
   (a) Če sta, dodaj par v polje skupnih skupin.
4. Ponavljaj, dokler v polju ni 50 parov.

Korak 3 je potreben, ker imamo v podatkih o skupinah genov tudi gene, ki jih v kakšni izmed tabelaričnih datotek ni. Velja tudi obratno: obstajajo geni, ki so v tabelarični datoteki, a jih ni med podatki o skupinah genov.

Za izbiro 50 parov genov, ki nimajo skupnih skupin, pa je bil postopek sledeč:

1. Iz datoteke GMT izberi dve naključni skupini.
2. Iz vsake izmed skupin izberi po en gen.
3. Preveri, ali sta oba gena vsebovana v datoteki mikromrež DNA in se ne pojavita skupaj v nobeni izmed skupin.
   (a) Če velja, dodaj par v polje različnih skupin.
4. Ponavljaj, dokler v polju ni 50 parov.

Rezultata metod sta bili dve polji parov genov. Rezultate smo shranili v tekstovno datoteko.

Izračun ocen dobljenih vizualizacij

Zadnji korak poskusa je bila ocena dobljenih vizualizacij. Kot že omenjeno, smo v ta namen uporabili metodo VizRank iz paketa Orange. V programu Orange smo v programskem jeziku Python napisali skripto, ki nam je iz tekstovne datoteke prebrala pare in za vsak par izračunala oceno vizualizacije. Rezultate smo shranili v novo tekstovno datoteko (programska koda vidna v 4.2).

Dobili smo rezultate vizualizacij. Napisali smo še metodo, ki nam je iz datoteke rezultate prebrala in izračunala povprečje dobljenih rezultatov. Z uporabo odprtokodne knjižnice Gadfly (https://dcjones.github.io/Gadfly.jl/) smo rezultate analizirali tudi s pomočjo grafov.

4.2.3 Rezultati

Rezultati vizualizacij parov

V tabeli 4.1 imamo prikaz rezultatov poskusa parov genov, predstavljenih v tabelarični obliki. Opazimo lahko, da so razlike v dobljenih povprečjih ocen zelo majhne. Povprečne ocene dobljenih vizualizacij ostanejo zelo podobne, ne glede na to, ali pare genov za vizualizacije vzamemo iz skupnih skupin ali različnih. V podatkih povprečnih ocen ne opazimo niti vzorcev (da bi za kakšno datoteko vedno veljalo, da bi bilo boljše vzeti par genov iz iste skupine ali podobno).

Razlogi za majhne razlike v rezultatih so lahko različni. Možno je, da imamo premajhen vzorec (50 parov), saj je različnih možnih vizualizacij zelo veliko. Po

Listing 4.2: Programska koda za izračun ocen vizualizacij

```
import orange
from orngVizRank import *

pairs = [line.strip() for line in open("groupIdx.txt")]
#index pairs
pairs = map((lambda line: line.split(",", 2)),pairs)

vizrank = VizRank(SCATTERPLOT)
classidx = len(in_data[1])-1

results = []
f = open("groupResults.txt", 'w')
for pair in pairs:
    #julia indecies starts with 1
    first = int(pair[0])-1
    second = int(pair[1])-1

    data = in_data.select([first,second,classidx])
    vizrank.setData(data)
    vizrank.evaluateProjections()
    results[len(results):] = [vizrank.results[0][0]]
    f.write(str(vizrank.results[0][0])+"\n")
f.close()

out_data = results
```

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	skupne	različne	skupne	različne	skupne	različne
DLBLC.tab	66.21	66.03	65.50	65.50	64.82	64.93
GDS971.tab	53.93	53.53	54.10	54.15	53.80	54.06
GDS1059.tab	51.02	51.36	51.09	50.82	50.41	51.01
GDS1667.tab	69.33	68.59	68.42	69.00	67.71	66.70
GDS2609.tab	56.96	59.78	60.54	57.78	56.77	61.67
GSE3726.tab	60.02	59.45	61.41	59.54	60.74	60.35

Tabela 4.1: Tabela prikazuje izračunana povprečja ocen parov genov. V stolpcu skupne imamo povprečne ocene, ki so dobljene iz 50 vizualizacij parov s skupnimi skupinami. V stolpcu "različne" so povprečne ocene parov genov iz različnih skupin.

drugi strani je morda podatek o povprečni oceni zavajajoč, saj je pomembna tudi razporeditev ocen vizualizacij. Dva primera imata lahko podobno povprečno oceno, ob tem pa ima en od primerov skoraj vse pare ocenjeno povprečno, drugi pa nekaj slabih, nekaj povprečnih in nekaj zelo dobrih vizualizacij. Razlog je lahko tudi, da skupine genov ne vplivajo opazno veliko na kakovost dobljenih vizualizacij. Poglejmo še podobno tabelo, v kateri imamo vsebovane podatke o oceni najboljše dobljene vizualizacije.

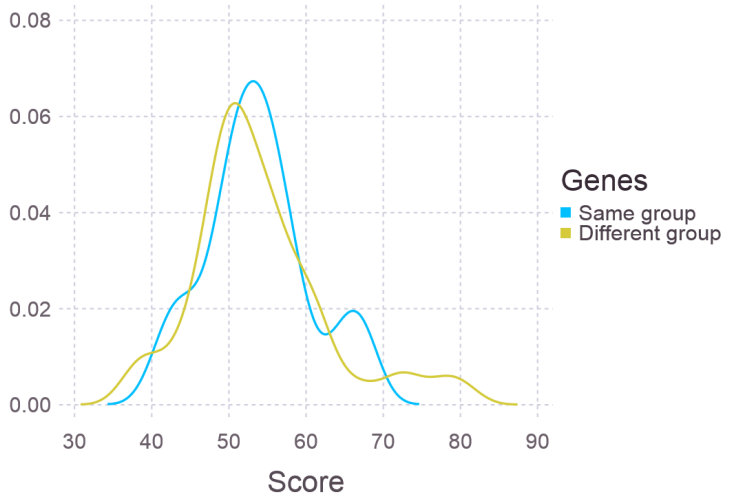
V tabeli 4.2 opazimo, da imajo pari s skupnimi skupinami genov iz skupin datoteke c5.mf vedno slabše ocenjen najboljši par kot pari različnih skupin. Če pogledamo še rezultate tabele 4.1; pri datoteki c5.mf vidimo večinoma slabše povprečne ocene v primeru skupnih skupin. Rezultati namigujejo, da skupine genov v datoteki c5.mf negativno vplivajo na kakovost dobljenih vizualizacij.

Še vedno razlike niso velike in so lahko (še posebej v tabeli najboljših ocen) posledica "posrečene" izbire genov v kombinaciji z razmeroma majhnim vzorcem. Drugih opaznih vzorcev v tabeli ni. Nekaj zanimivih rezultatov si lahko pogledamo tudi v grafični obliki.

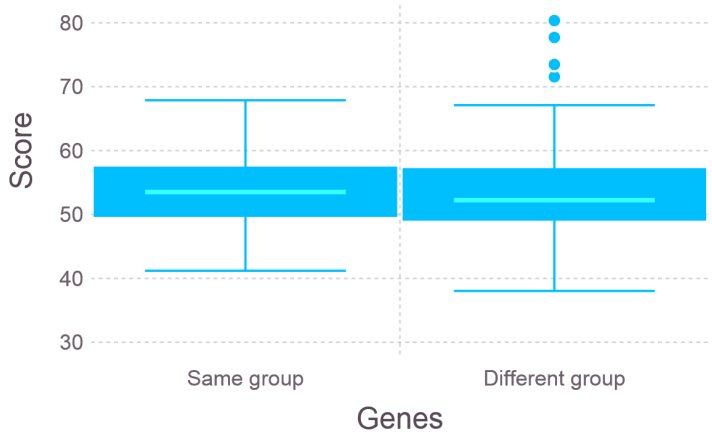
V sliki 4.1 vidimo lep pregled dobljenih ocen, ki bi jih v tabelarični obliki težje razumeli. Opazimo, da imajo geni različnih skupin več dobro ocenjenih vizualizacij

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	skupne	različne	skupne	različne	skupne	različne
DLBLC	77.18	76.61	75.47	71.90	73.63	75.39
GDS971	77.95	78.07	81.51	72.25	67.88	80.36
GDS1059	58.16	61.02	56.45	58.24	56.99	57.38
GDS1667	91.47	82.33	83.68	88.42	79.02	83.84
GDS2609	74.57	91.69	87.29	88.11	81.82	91.96
GSE3726	78.64	74.10	80.95	75.35	76.30	80.46

Tabela 4.2: Tabela vsebuje oceno najboljše vizualizacije, dobljene iz parov s skupnimi in različnimi skupinami.



Slika 4.1: Graf porazdelitve primera GDS971 z uporabljenimi skupinami iz c5mf.



Slika 4.2: Boxplot diagram primera GDS971 z uporabljenimi skupinami iz c5mf.

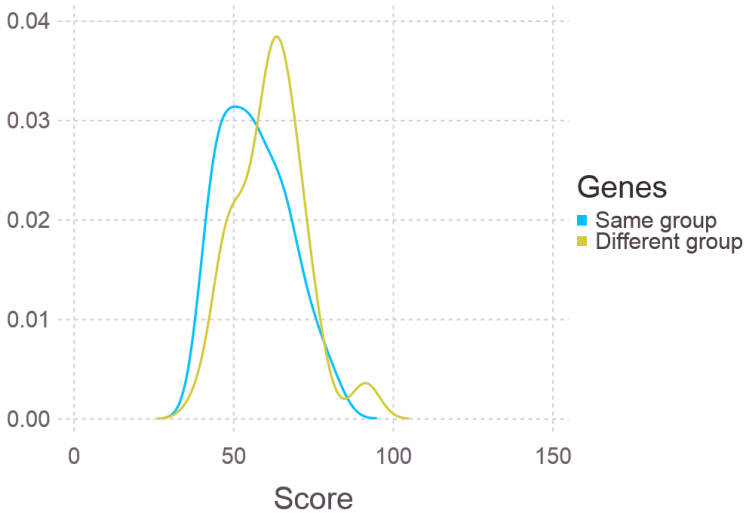
iz datoteke GDS971, kljub temu da je povprečna ocena vizualizacij večja pri genih iz istih skupin. Enako lahko opazimo v sliki 4.2, ki še nazorneje prikaže večje povprečje skupin genov iz enakih skupin, kljub temu da ima druga skupina kar nekaj zelo dobro ocenjenih vizualizacij. Sliki sta odlična primera, zakaj sta lahko povprečna ocena in ocena najboljše vizualizacije varljivi.

Slika 4.3 nam predstavlja primer ocen vizualizacij iz genov GDS2609 z uporabljenimi skupinami genov iz datoteke c5.mf. Vidimo lahko potrditev ugotovitve tabelaričnih podatkov, da nam v tem primeru geni iz različnih skupin prinesejo boljše vizualizacije.

Rezultati vizualizacij trojic

Rezultati poskusa trojic genov (tabela 4.3) so podobni rezultatom pri parih genov. Zopet imamo med podatki majhne razlike. Opazimo lahko, da skupine genov iz datotek c2.cp in c5.cb pri večini datotek vizualizacij pozitivno vplivajo na dobljene rezultate.

Dobljeni rezultati parov in trojic nam ne dajo jasnega odgovora na vprašanje o vplivu skupin genov na kakovost vizualizacij. Pri vseh ugotovitvah je potrebno opozoriti, da so razlike zelo majhne. Zaradi omenjenih razlogov smo se odločili nadaljevati z drugačnimi poskusi, ki so opisani v naslednjih poglavjih.



Slika 4.3: Graf porazdelitve primera GDS2609 z uporabljenimi skupinami iz c5mf.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	skupne	različne	skupne	različne	skupne	različne
DLBLC	66.28	66.59	66.01	65.15	65.85	66.05
GDS971	52.69	50.56	53.57	52.48	51.77	50.22
GDS1059	51.36	50.09	50.68	50.24	50.23	51.06
GDS1667	68.34	67.37	67.65	67.21	67.14	68.13
GDS2609	60.58	56.98	57.67	60.18	59.99	58.56
GSE3726	58.21	59.42	59.63	58.87	58.72	59.03

Tabela 4.3: Tabela prikazuje izračunana povprečja ocen trojic genov. V stolpcu skupne imamo povprečne ocene, ki so dobljene iz 50 vizualizacij trojic s skupnimi skupinami. V stolpcu "različne" so povprečne ocene trojic genov iz različnih skupin.

4.3 Drugi del eksperimentov

4.3.1 Ideja

V drugem delu smo se poskusa lotili drugače. Tokrat smo začeli tako, da smo z metodo VizRank poiskali dobro ocenjene vizualizacije parov genov. Izmed vseh vizualizacij smo izbrali 100 in 1000 najbolje ocenjenih. Za par genov iz skupine najbolje ocenjenih vizualizacij smo preverili, ali imata gena morda kakšno skupno skupino. Število pojavitev smo prešteli in izračunali odstotni delež vizualizacij, kjer sta oba gena iz iste skupine.

Na koncu smo analizirali rezultate in preverili, ali morda pri parih genov najboljših vizualizacij dobimo večji procent takih, kjer sta gena iz iste skupine. Po enakem postopku smo naredili poskus tudi pri trojicah genov.

4.3.2 Potek dela

Iskanje najboljših vizualizacij

Naša prva naloga je bila pridobitev 100 in 1000 najbolje ocenjenih vizualizacij. Uporabili smo metodo VizRank. Ker je različnih možnih vizualizacij izmed več tisoč genov ogromno, smo metodo pognali s časovno omejitvijo 2 uri (zakaj smo vseeno dobili večino najboljših vizualizacij, je opisano v podpoglavju 2.5.3). Tako smo dobili približek najboljših 1000 ocenjenih vizualizacij parov genov. Ker je postopek iskanja najboljših vizualizacij zamuden, smo rezultate shranili v tekstovne datoteke.

Za branje rezultatov iz datoteke smo napisali metodo, s katero preberemo projekcijsko datoteko, ki vsebuje vizualizacije in njihove ocene.

Listing 4.3: Programska koda za branje datoteke najboljših ocenjenih vizualizacij

```
function readProjectionPairs(path)
    pairs = {}
    open(path, "r") do f
        for line in eachline(f)
            m = match(r"'.*'", line)
            if m != nothing
                fst = split(m.match, ',')[1][2:end-1];
                snd = split(m.match, ',')[2][3:end-1];
                push!(pairs, (fst, snd));
            end
        end
    end
    return pairs
end
```

Tako smo iz najboljših vizualizacij dobili 1000 parov genov.

Preverjanje pripadnosti skupnih skupin

Za vsak par genov smo preverili, ali imata gena kakšno skupno skupino ali ne. Skupine smo dobili iz datoteke GMT, kot že opisano v prvem delu poskusov. Iz prejšnjih poskusov smo prav tako imeli že metodo, ki nam za par genov pove, ali imata kakšno skupno skupino ali ne. Za vse pare genov smo preverili, če imata kakšno skupno skupino. Na koncu smo prešteli, koliko parov izmed parov najboljših vizualizacij ima iste skupine.

Trojice genov

Enak poskus kot za pare smo naredili še za trojice genov. Metode. opisane v prvem delu, so napisane splošno, tako da smo lahko večino kode uporabili iz poskusa s pari genov. Dodatno smo morali napisati metodo, ki nam prebere projekcije, saj je bila prejšnja napisana specifično za pare. Zopet smo z VizRank 2 uri iskali najboljše projekcije. Tokrat smo ocenjevali RadViz projekcije s tremi geni.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	1	2	9	8	2	2
GDS971	1	0	8	8	0	2
GDS1059	0	0	2	10	0	5
GDS1667	2	1	1	0	0	0
GDS2609	1	0	1	0	0	0
GSE3726	3	2	13	3	5	0

Tabela 4.4: Tabela prikazuje število parov genov, ki spadajo v skupne skupine izmed najboljših 100 vizualizacij in izmed naključno izbranih parov.

Pri analizi dobljenih trojic smo pogledali, kolikokrat se pojavijo trojice v istih skupinah. Analizo smo naredili na dva različna načina. Prvič smo šteli trojice, kjer se vsi trije geni pojavijo v isti skupini. Drugič pa smo dovolili, da se le dva gena iz trojice pojavita skupaj v eni izmed skupin.

4.3.3 Rezultati

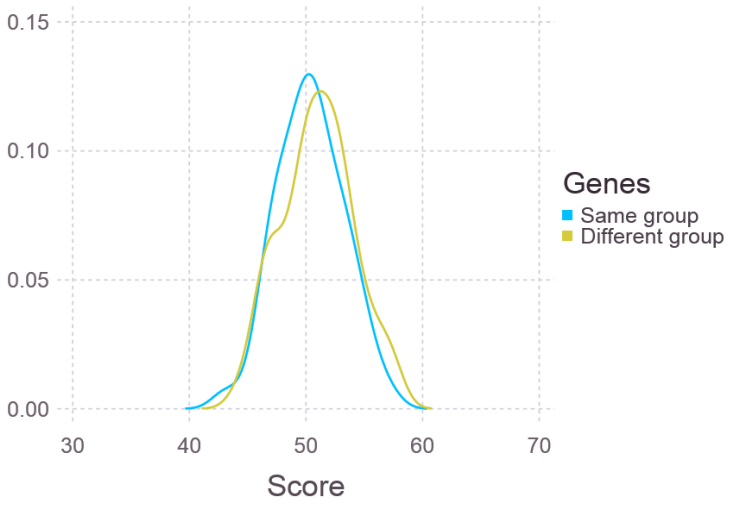
Rezultati vizualizacij parov

Poskus za pare genov smo najprej naredili za najboljših 100 vizualizacij. Rezultati so prikazani v tabeli 4.4. Izmed vseh rezultatov izstopajo štirje. To so: pri datoteki GDS1059 pari iz skupin c5.cb in c5.mf in pri datoteki GSE3726 prav tako pari iz skupin c5.cb in c5.mf.

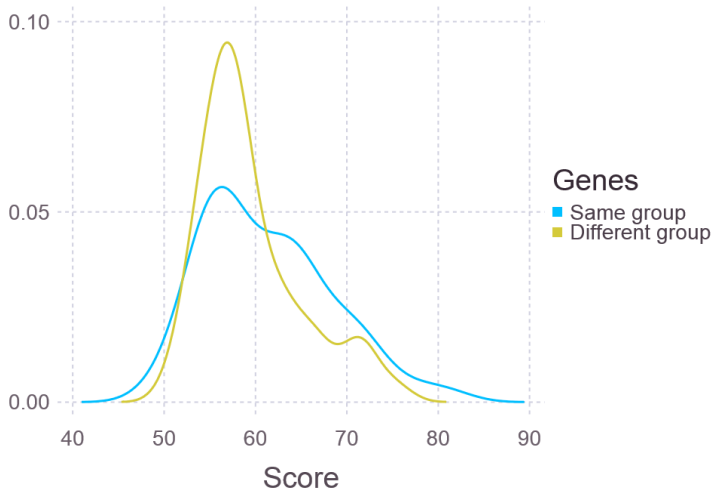
Če smo pare izbirali naključno, smo pri datoteki GDS1059 in skupin iz datotek c5.cb in c5.mf dobili več parov, ki spadajo v iste skupine, kot jih je bilo med najboljšimi stotimi vizualizacijami. Iz tega lahko sklepamo, da v tem primeru pari iz skupnih skupin negativno vplivajo na kakovost dobljenih vizualizacij. Rezultate lahko primerjamo z rezultati iz prvega poskusa in se ujemajo. V prvem poskusu smo v primerih GDS1059 z datotekama c5.cb in c5.mf dobili boljše povprečne ocene vizualizacij pri parih iz različnih skupin kot pri parih iz skupnih skupin 4.1, kar se vidi v grafu 4.4.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	20	14	77	45	24	9
GDS971	19	19	79	58	8	15
GDS1059	11	17	51	78	9	19
GDS1667	21	1	20	6	2	0
GDS2609	10	2	34	6	8	1
GSE3726	7	2	90	31	21	11

Tabela 4.5: Tabela prikazuje število parov genov, ki spadajo v skupne skupine izmed najboljših 1000 vizualizacij in izmed naključno izbranih parov.



Slika 4.4: Graf porazdelitve primera GDS1059 z uporabljenimi skupinami iz c5mf iz eksperimentov opisanih v podpoglavju 4.2.3.



Slika 4.5: Graf porazdelitve primera GSE3726 z uporabljenimi skupinami iz c5bp iz eksperimentov opisanih v podpoglavju 4.2.3.

Ravno obratno velja za pare iz skupin GSE3726. Tam smo dobili veliko več parov v istih skupinah pri parih najboljših 100 vizualizacij. Tudi tukaj se rezultati ujemajo z rezultati prvega dela poskusov, kar se vidi v grafu 4.5.

Ker smo pri večini rezultatov dobili premajhna števila za primerjanje, smo poskus ponovili še z najboljšimi 1000 izbranimi vizualizacijami. Rezultati so v tabeli 4.5. Tukaj opazimo, da dobimo pri datotekah DLBLC, GDS1667, GDS2609 in GSE3726 vedno boljše rezultate pri parih iz najboljših 1000 vizualizacij. Iz tega lahko sklepamo, da pri teh datotekah skupine genov pozitivno vplivajo na kakovost dobljenih vizualizacij. Pri datoteki GDS1059 opazimo podobno kot v prvem delu tega poskusa, in sicer, da pripadnost skupnim skupinam negativno vpliva na kakovost rezultatov.

Rezultati vizualizacij trojic

Pri trojicah genov smo naredili dve analizi. Prvič smo prešteli, koliko pojavitev trojic je takih, kjer so vsi trije geni iz trojice vsebovani skupaj v kakšni izmed skupin. Drugič pa smo prešteli, koliko je takih, kjer sta le 2 izmed trojice genov vsebovana skupaj v kakšni skupini.

Tabela 4.6 prikazuje rezultate prve analize. Pričakovano so števila zelo majhna,

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	0	0	0	0	0	1
GDS971	0	0	0	1	0	1
GDS1059	0	0	0	0	0	0
GDS1667	0	0	0	0	0	0
GDS2609	0	0	3	0	0	0
GSE3726	0	0	4	0	0	0

Tabela 4.6: Tabela prikazuje število trojic genov, kjer geni vsi trije geni spadajo v skupne skupine izmed najboljših 100 vizualizacij in izmed naključno izbranih trojic.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	0	0	12	5	0	0
GDS971	0	0	4	2	0	1
GDS1059	0	1	9	10	0	0
GDS1667	0	0	0	0	0	0
GDS2609	4	0	4	0	1	0
GSE3726	0	1	27	6	0	0

Tabela 4.7: Tabela prikazuje število trojic genov, kjer geni vsi trije geni spadajo v skupne skupine izmed najboljših 1000 vizualizacij in izmed naključno izbranih trojic.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	5	10	17	16	5	4
GDS971	5	2	15	12	2	1
GDS1059	8	7	8	15	2	1
GDS1667	6	1	12	4	0	0
GDS2609	0	0	3	0	8	1
GSE3726	0	3	28	11	4	1

Tabela 4.8: Tabela prikazuje število trojic genov, kjer vsaj dva izmed trojice genov spadata v skupne skupine izmed najboljših 100 vizualizacij in izmed naključno izbranih trojic.

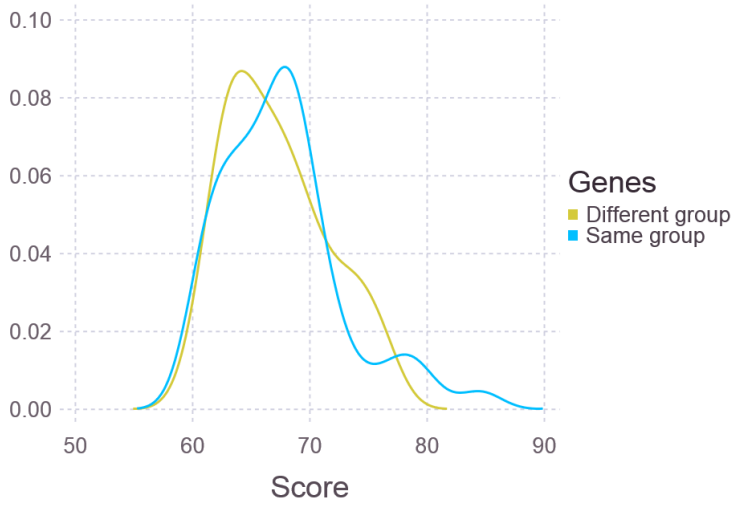
saj je majhna verjetnost, da so trije geni vsebovani v isti skupini. Vseeno lahko opazimo, da pri datotekah GDS2609 in GSE3726 v kombinaciji s skupinami iz c5.cb skupine pozitivno vplivajo na kakovost dobljenih vizualizacij. To se potrdi tudi v drugem delu analize, kar se vidi v tabeli 4.8. Zaradi majhnih števil smo naredili še enak poskus za najboljših 1000 vizualizacij (tabela 4.7), kjer se rezultati ujemajo z drugo vrsto analize, ki sledi.

V drugi analizi pričakovano dobimo večje rezultate. Opazimo lahko, da pri datotekah GDS971, GDS1667, GDS2609 in GSE3726 genske skupine po večini dobro vplivajo na kakovost dobljenih projekcij, pri datoteki GDS971 pa skupine negativno vplivajo na kakovost dobljenih projekcij.

V vseh treh tabelah je opazno tudi, da največje številke dobimo pri skupinah iz datoteke c5.cb. Razlog za veliko število genov ni v velikosti datoteke, saj imamo v datoteki c2.cp več skupin genov kot v c5.cb (3.2). Očitno se geni iz genskih skupin datoteke c5.cb najpogosteje pojavljajo v podatkovnih zbirkah mikromrež DNA.

Rezultati se po večini ujemajo tudi z rezultati prvega dela poskusov. Primer ujemanja je datoteka GDS1667, kjer smo ugotovili, da genske skupine dobro vplivajo na kakovost vizualizacij, kar se vidi tudi v sliki 4.6.

Tudi pri drugem delu poskusov razlike niso bile prepričljive. Nekaj rezultatov dobljenih v drugem delu tudi ni v popolnem ujemanju z rezultati iz prvega dela.



Slika 4.6: Graf porazdelitve trojic primera GSD1667 z uporabljenimi skupinami iz c5bp iz eksperimentov opisanih v podpoglavju 4.2.3.

Kot smo že omenili, je posledica lahko majhen vzorec (predvsem v prvem delu poskusov), majhne razlike v rezultatih ali posrečena izbira genov. Zaradi teh razlogov smo se odločili narediti še tretji del poskusov, ki so podrobno opisani v naslednjih poglavjih.

4.4 Tretji del eksperimentov

4.4.1 Ideja

Tretjega dela poskusov smo se lotili na drugačen način. Najprej smo podobno kot v drugem delu izbrali 1000 najboljših vizualizacij. Tokrat smo izbirali Radviz vizualizacije s 5 geni. Za vsak par genov (iz dobljenih peteric) smo pogledali, kolikokrat se pojavita skupaj v kakšni izmed peteric najboljših 1000 vizualizacij. Iz dobljenih podatkov smo skonstruirali graf. Vsako vozlišče grafa je predstavljalo en gen. Dve vozlišči sta bili povezani, če se par genov, ki jih predstavljata, pojavi skupaj v kakšni izmed 1000 najboljših vizualizacij. Na koncu smo analizirali dobljene grafe.

4.4.2 Potek dela

Iskanje najboljših vizualizacij

Prvi del poskusa je bil podoben tistemu iz drugega dela. Metodo VizRank smo časovno omejili na 2 uri. Ta nam je iskala Radviz vizualizacije s 5 atributi. Dobljenih 1000 najboljših vizualizacij smo shranili v tekstovno datoteko. Iz datoteke, smo kot že v prejšnjih nalogah, prebrali najboljše vizualizacije. Tako smo dobili tabelo, ki je vsebovala 1000 peteric genov in ocene njihovih vizualizacij. Iz tabele peteric smo zgradili novo polje, ki je vsebovalo vse naštete gene. Ponavljajoči se geni so bili vsebovani samo enkrat.

Listing 4.4: Programska koda za kreiranje polja vseh genov iz peteric.

```
function uniqueGenes(array)
    genes = {};
    for line in array
        for g in line
            if (!in(g, genes))
                push!(genes, g);
            end
        end
    end
    return genes;
end
```

Izračun potrebnih podatkov za kreiranje grafov

Iz dobljene tabele genov smo zgradili vse možne kombinacije parov genov. Za vsak dobljen par genov smo prešteli, kolikokrat se gena pojavita skupaj v najboljših 1000 vizualizacijah peteric genov. Metodi podamo dva argumenta. Prvi argument je tabela parov, katerih število pojavitev v skupinah želimo prešteti. Kot drugi argument podamo tabelo skupin genov. Metoda vsakemu paru v tabeli "pripne" število skupnih pojavitev v najboljših petericah vizualizacij.

Listing 4.5: Programska koda, ki prešteje število pojavitev parov v najboljših 1000 dobljenih vizualizacijah.

```
function countPairsInGroups(pairs, groups)
    counters = {};
    for i=1:(length(pairs))
        counter = 0;
        (fst, snd) = pairs[i];
        for j=1:(length(groups))
            if (in(fst,groups[j]) && in(snd,groups[j]))
                counter += 1;
            end
        end
        push!(counters, counter);
    end
    return hcat(counters, pairs);
end
```

Tako smo dobili vse potrebno za kreiranje grafa. Za predstavitev grafa smo uporabili format .net (network), ki ga uporablja program za vizualizacijo podatkov Pajek. Format je prav tako podprt v programu Orange.

Sledi opis formata net, ki je namenjen zapisu grafov v tekstovni obliki.

Oblika NET datotek

V prvo vrstico vstavimo ključno besedo *vertices. Poleg nje, za presledkom ločeno, napišemo število vozlišč, ki jih naš graf vsebuje. Prvi vrstici sledijo vrstice, od katerih vsaka predstavlja eno vozlišče. Vozlišče predstavimo tako, da najprej podamo njegov enolični identifikator (v našem primeru zaporedna številka) in opcijsko ime vozlišča (v našem primeru kratica gena). Po naštetih vseh vozliščih vstavimo vrstico, ki vsebuje ključno besedo *edges. Sledijo vrstice, ki predstavljajo povezave med vozlišči. Povezave predstavimo tako, da s presledkom ločimo ID izvornega vozlišča in ID ponornega vozlišča. Opcijsko lahko dodamo povezavi še težo.

Izdelava grafov

Za predstavitev genov v formatu NET smo napisali pomožne metode.

Listing 4.6: Programske metode, za kreiranja grafa v formatu .net

```
function createGraphNode(id, label)
    return strincg(id, " ", label, "\n");
end

function createGraphEdge(sourceId, targetId)
    return string(sourceId, " ", targetId, "\n");
end

function createGraph(allGenes, allPairs, n=0)
    graph = "";
    counter = 1;
    for gene in allGenes
        graph = string(graph,createGraphNode(counter,gene));
        counter += 1;
    end
    graph = string("*vertices ", (counter-1), "\n", graph);

    graph = string(graph, "*edges\n");
    for i=1:length(allPairs[:,1])
        (numb, (fst, snd)) = allPairs[i,:];
        if (numb > n)
            graph = string(graph,
                createGraphEdge(getIndex(fst, allGenes),
                getIndex(snd, allGenes)));
        end
    end

    return graph;
end
```

Metodi createGraph podamo tri parametre. Prvi je polje vseh genov. Metoda za vsak gen v tem polju ustvari eno vozlišče. Drugi parameter so vsi pari genov. Tretji, opcijski parameter (privzeta vrednost je 0), pa predstavlja, kolikokrat se mora par genov pojaviti skupaj v eni izmed najboljših 1000 vizualizacij genov, da med njima naredimo povezavo.

Dobili smo željen graf, kjer vsako vozlišče predstavlja en gen, povezave med njimi pa obstajajo, če sta dva gena skupaj v vsaj eni izmed 1000 najbolje ocenjenih vizualizacijah s petimi geni.

Poleg osnovnega grafa smo skonstruirali še nekaj podobnih grafov. Pri enem smo povezavo med genoma naredili le v primeru, če se par pojavi vsaj 10-krat skupaj v 1000 najbolje ocenjenih vizualizacijah. V naslednjem primeru pa smo narisali povezavo le, če se pojavita skupaj 20-krat.

Prvo vprašanje, ki smo si ga zastavili, je bilo sledeče. So geni, ki so povezani večkrat iz istih skupin, ali ravno nasprotno, večkrat iz različnih skupin genov?

Iz dobljenega grafa direktno ni bilo mogoče sklepati o odgovoru na vprašanje. V ta namen smo povezavam dodali uteži. Kot že opisano zgoraj, lahko v datotekah formata net povezavam opcijsko dodamo uteži tako, da pri povezavi dodamo še število, ki predstavlja utež povezave. Graf smo predelali tako, da smo povezave, kjer sta gena iz različnih skupin, otežili z majhno utežjo: 1, povezavam genov iz istih skupin pa smo dali utež 10.

Listing 4.7: Potrebne spremembe v programski kodi

```
function createGraphEdge(sourceId, targetId, sameGroup)
    size = (sameGroup == true) ? 10 : 1;
    return string(sourceId," ",targetId," ",size,"\n");
end

graph = string(graph, createGraphEdge(
        getIndex(fst, allGenes),
        getIndex(snd, allGenes),
        isSameGroup(fst, snd)));
```

Iz dobljenih grafov smo lahko hitro ocenili, da je pri dobljenih parih genov več genov iz različnih skupin kot genov iz istih skupin.

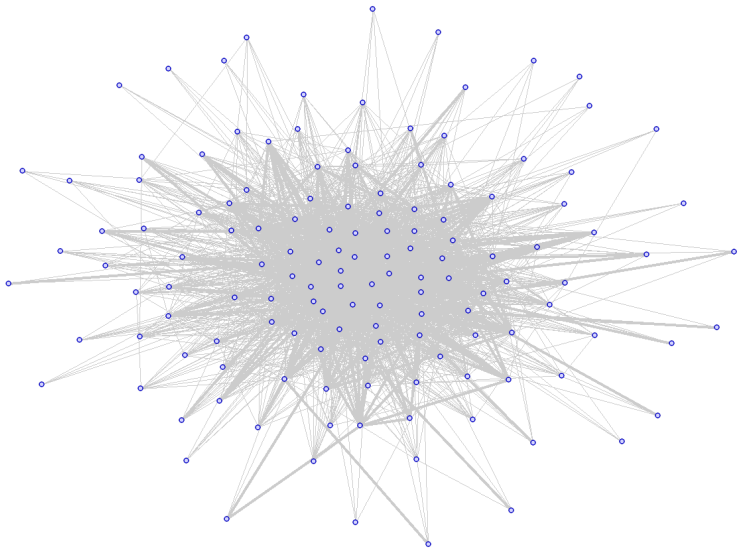
Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	2.65%	2.19%	9.02%	7.87%	1.97%	1.53%
GDS971	1.92%	1.39%	5.62%	6.48%	1.52%	1.25%
GDS1059	1.65%	1.58%	7.56%	6.53%	1.79%	1.37%
GDS1667	1.44%	0.22%	2.85%	0.69%	0.45%	0.09%
GDS2609	1.17%	0.23%	3.34%	1.05%	1.35%	0.23%
GSE3726	0.55%	0.77%	7.34%	3.90%	1.29%	1.11%

Tabela 4.9: Tabela prikazuje procente parov genov, ki spadajo v skupne skupine v grafu parov najboljših 1000 vizualizacij in izmed naključno izbranih parov.

Za natančnejše rezultate smo povezave istih in različnih skupin prešteli še programsko. Za vsak par genov smo uporabili že opisano metodo, ki nam pove, ali ima par kakšno skupno skupino. Izračunali smo delež genov iz istih skupin in delež genov iz različnih skupin. Računsko smo dobili potrditev rezultatov grafične metode. V vseh grafih je več parov genov iz različnih skupin kot iz istih. Dobljeni rezultati niso presenetljivi, saj smo v drugem delu poskusov ugotovili, da je med podatki veliko več parov genov iz različnih skupin kot tistih, ki spadajo v skupne skupine.

Na koncu smo deleža parov genov s skupnimi skupinami med pari najboljših vizualizacij primerjali z deleži pri naključno izbranih genih. Če se razlikujeta, bi s tem lahko pokazali, da pripadnost genov skupnim skupinam vpliva na kakovost dobljenih vizualizacij.

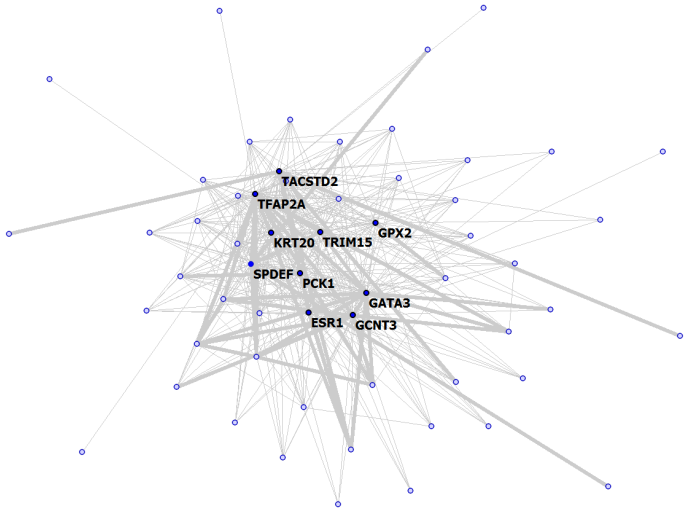
V ta namen smo za vsak graf izbrali toliko naključnih parov genov, kot smo imeli v grafu vozlišč. V ta namen smo uporabili že opisane metode iz prejšnjih nalog. Dobljene pare smo analizirali na enak način kot prejšnje. Izračunali smo delež parov, kjer imata gena skupno skupino in delež parov genov iz različnih skupin. Rezultate smo shranili in jih primerjali z dobljenimi iz najboljših vizualizacij.



Slika 4.7: Primer dobljenega grafa iz GSE3726 iz rezultatov v tabeli 4.9.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	2.47%	2.57%	10.89%	6.43%	3.46%	0.99%
GDS971	2.89%	0.48%	2.89%	9.66%	0.48%	1.45%
GDS1059	3.77%	0.47%	6.60%	7.07%	2.36%	1.89%
GDS1667	6.08%	0%	4.35%	1.63%	0%	0.86%
GDS2609	5.48%	0%	7.32%	0%	4.87%	0%
GSE3726	0%	0.87%	10.91%	4.80%	0.87%	1.75%

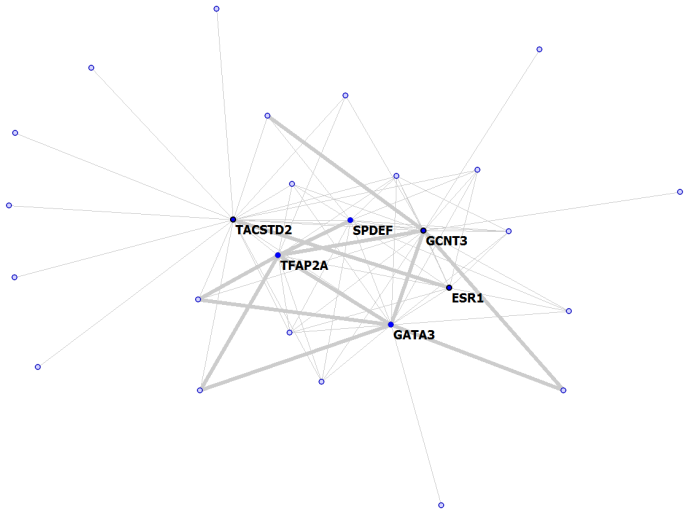
Tabela 4.10: Tabela prikazuje procente parov genov, ki spadajo v skupne skupine v grafu parov najboljših 1000 vizualizacij, kjer imamo povezave samo, če nastopajo skupaj v vsaj 10 skupinah, in izmed naključno izbranih parov.



Slika 4.8: Primer dobljenega grafa iz GSE3726 iz rezultatov v tabeli 4.10. Geni brez povezav so skriti (geni, ki niso z nobenim od ostalih genov vsaj desetkrat v najboljših 1000 vizualizacijah). Označeni so tisti geni, ki imajo nadpovprečno število povezav.

Datoteka	c2.cp.v3.0.symbols		c5.cb.v3.0.symbols		c5.mf.v3.0.symbols	
	najboljši	naključni	najboljši	naključni	najboljši	naključni
DLBLC	2.02%	4.04%	11.89%	6.06%	4.04%	3.03%
GDS971	2.65%	0.88%	3.53%	3.53%	0.88%	2.65%
GDS1059	5.82%	2.91%	5.82%	4.84%	1.94%	0.97%
GDS1667	10.68%	0%	8.51%	0%	0%	0%
GDS2609	9.80%	0%	5.88%	1.96%	11.76%	0%
GSE3726	0%	0%	15.58%	3.89%	1.29%	1.29%

Tabela 4.11: Tabela prikazuje procente parov genov, ki spadajo v skupne skupine v grafu parov najboljših 1000 vizualizacij, kjer imamo povezave samo, če nastopajo skupaj v vsaj 20 skupinah, in izmed naključno izbranih parov.



Slika 4.9: Primer dobljenega grafa iz GSE3726 iz rezultatov v tabeli 4.11. Geni brez povezav so skriti. Označeni so tisti geni, ki imajo nadpovprečno število povezav.

4.4.3 Rezultati

Najprej smo naredili grafe, kjer so povezave med geni, če se gena vsaj enkrat pojavita skupaj v 1000 najboljših vizualizacijah. Rezultati so prikazani v tabeli 4.9. Opazimo, da se nam potrdijo rezultati prejšnjih dveh poskusov. Pri datotekah GDS1776, GDS2609 in GSE3726, še posebej v kombinaciji z skupinami iz c5.cb in c5.mf, dobimo boljše rezultate pri genih, ki spadajo v skupne skupine, kot pri naključni izbiri genov. Ker so razlike majhne, smo naredili poskus še za grafe, kjer so povezave med geni, če spadajo vsaj 10-krat v najboljših 1000 vizualizacij in 20-krat. Rezultati so prikazani v tabelah 4.10 in 4.11.

Večjo kot smo postavili mejo, kolikokrat mora biti par genov vsebovan v najboljših 1000 vizualizacijah, bolj nazorne rezultate smo dobili. Prav tako se rezultati vseh treh poskusov ujemajo.

V tabeli 4.11 opazimo, da ima c5.cb pri večini datotek največje odstotne deleže skupin tako pri naključni izbiri kot pri izbiri parov iz grafa. Enako smo opazili že v drugem delu poskusov, kar se nam je sedaj potrdilo.

Iz vseh treh delov eksperimentov (4.2.3, 4.3.3, 4.4.3) lahko sklepamo, da je vpliv skupin na kakovost dobljenih vizualizacij odvisen od izbire datoteke skupin in genskih podatkov, ki jih vizualiziramo.

Pri datoteki GDS1667 v vseh poskusih ugotovimo, da geni, ki imajo skupne skupine genov, dobro vplivajo na rezultat vizualizacij. To velja še posebej za skupine iz datotek c2.cp in c5.cb. Enako lahko sklepamo za datoteki GDS2609 in GSE3726, pri katerih tudi skupine iz datoteke c5.mf dobro vplivajo na kakovost dobljenih projekcij.

Po drugi strani pri datotekah DLBLC, GDS971 in GDS1059 nismo opazili izboljšanja projekcij pri izbiri genov iz skupnih skupin. Datoteka GDS1059 pri rezultatih izstopa, saj poskusi kažejo, da je morda celo bolje vzeti gene, ki ne spadajo v iste skupine, kot naključne vizualizacije.

Omembe vredna je tudi ugotovitev, da smo pri večini datotek mikromrež DNA največje rezultate dobili pri skupinah genov iz c5.cb. Ti rezultati kažejo, da se geni vsebovani v skupinah c5.cb najbolje ujemajo z vzorcem genov, ki nastopajo v datotekah mikromrež DNA.

V grafih 4.8 in 4.9 lahko tudi označimo gene z nadpovprečnim številom povezav. Tako vidimo imena tistih genov, ki so z največ različnimi geni (z vsakim vsaj desetkrat, dvajsetkrat) vsebovani v najboljših 1000 vizualizacijah. Sklepamo lahko, da so ti geni med tistimi, ki največ pripomorejo h kakovosti dobljenih vizualizacij.

Poglavje 5

Zaključek

Podrobni rezultati poskusov so predstavljeni v podpoglavjih 4.2.3, 4.3.3 in 4.4.3. V poskusih smo ugotovili, da je vpliv genskih skupin na kakovost dobljenih vizualizacij različen za različne datoteke mikromrež DNA. Pri polovici podatkov mikromrež DNA smo ugotovili, da genske skupine pozitivno vplivajo na kakovost dobljenih vizualizacij. Pomembna je tudi izbira genskih skupin. Izkazalo se je, da ena izmed datotek z genskimi skupinami bistveno bolje vpliva na rezultate genskih vizualizacij (pri večini datotek mikromrež izboljša rezultate) kot pa ostali dve (izboljšata samo pri polovici).

Poleg rezultatov genskih skupin so prispevki dela tudi ideje izvedenih poskusov. S poskusi smo uspeli izvedeti, kakšen je vpliv genskih skupin na kakovost vizualizacij genov. Ker so vse ideje poskusov splošne, jih je možno uporabiti tudi za druge vrste podatkov.

V delu se ukvarjamo s sorazmeroma majhnim številom različnih podatkov. Poskuse delamo samo na šestih različnih podatkih mikormrež DNA, skupine genov pa izbiramo samo iz treh različnih datotek dobljenih iz baze MSigDB. V prihodnje bi se izplačalo z opisanimi poskusi preveriti vpliv skupin genov na več različnih podatkih mikromrež DNA. Prav tako bi se izplačalo preveriti vpliv vseh datotek genskih skupin iz baze MSigDB in ugotoviti, katere najbolje vplivajo na kakovosti vizualizacij.

V nadaljnjem delu so možne tudi različne izboljšave metode VizRank. Metodo VizRank bi lahko razširili, da bi za iskanje vizualizacij mikromrež DNA pri iskanju uporabila tudi predznanje genskih skupin.

Raziskati bi bilo vredno tudi druge podobne primere vizualizacij podatkov s predznanjem in metodo VizRank izboljšati tako, da bi ji pri splošnih podatkih lahko podali predznanje, ta bi ga pa sama uporabila pri iskanju vizualizacij.

Literatura

[1] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[2] Claudia Baer, Mattias Nees, Stephen Breit, Barbara Selle, Andreas E Kulozik, Karl-Ludwig Schaefer, Yvonne Braun, Daniel Wai, and Christopher Poremba. Profiling and functional annotation of mrna gene expression in pediatric rhabdomyosarcoma and ewing's sarcoma. *International journal of cancer*, 110(5):687–694, 2004.

[3] Jeff Bezanson, Stefan Karpinski, Viral B Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. *arXiv preprint arXiv:1209.5145*, 2012.

[4] Chris Brunsdon, AS Fotheringham, and ME Charlton. An investigation of methods for visualising highly multivariate datasets. *Case Studies of Visualization in the Social Sciences*, pages 55–80, 1998.

[5] Stuart K Card, Jock D Mackinlay, and Ben Shneiderman. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.

[6] Dondapati Chowdary, Jessica Lathrop, Joanne Skelton, Kathleen Curtin, Thomas Briggs, Yi Zhang, Jack Yu, Yixin Wang, and Abhijit Mazumder. Prognostic gene expression signatures can be measured in tissues collected in rnalater preservative. *The journal of molecular diagnostics*, 8(1):31–39, 2006.

[7] William S Cleveland and Robert McGill. The many faces of a scatterplot. *Journal of the American Statistical Association*, 79(388):807–822, 1984.

[8] Janez Demšar, Blaž Zupan, Gregor Leban, and Tomaz Curk. *Orange: From experimental machine learning to interactive data mining.* Springer, 2004.

[9] Georges Grinstein, Marjan Trutschl, and Urrka Cvek. High-dimensional visualizations. In *Proceedings of Workshop on Visual Data Mining, ACM Conference on Knowledge Discovery and Data Mining*, pages 1–14, 2001.

[10] Hilary M Hearnshaw, David John Unwin, et al. *Visualization in geographical information systems.* John Wiley & Sons Ltd, 1994.

[11] Patrick Hoffman, Georges Grinstein, and David Pinkney. Dimensional anchors: a graphic primitive for multidimensional multivariate information visualizations. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge management*, pages 9–16. ACM, 1999.

[12] Yi Hong, Kok Sun Ho, Kong Weng Eu, and Peh Yean Cheah. A susceptibility gene set for early onset colorectal cancer that integrates diverse signaling pathways: implication for tumorigenesis. *Clinical Cancer Research*, 13(4):1107–1114, 2007.

[13] P Juvan et al. Tehnologija dna mikromrež in njena uporaba v medicini. *11SDMI*, page 2.

[14] Arie Kaufman. Trends in visualization and volume graphics, scientific visualization advances and challenges, 1994.

[15] Igor Kononenko and Edvard Simec. Induction of decision trees using relieff. In *Proceedings of the ISSEK94 Workshop on Mathematical and Statistical Methods in Artificial Intelligence*, pages 199–220. Springer, 1995.

[16] Gregor Leban. *Vizualizacija podatkov s strojnim ucenjem.* PhD thesis, University of Ljubljana, Factulty of Computer and Information Science, 2007.

[17] Gregor Leban, Blaž Zupan, Gaj Vidmar, and Ivan Bratko. Vizrank: Data visualization guided by machine learning. *Data Mining and Knowledge Discovery*, 13(2):119–136, 2006.

[18] Minca Mramor, Gregor Leban, Janez Demšar, and Blaž Zupan. Visualization-based cancer microarray data classification analysis. *Bioinformatics*, 23(16):2147–2154, 2007.

[19] Helen C. Purchase. Effective information visualisation: a study of graph drawing aesthetics and algorithms. *Interacting with computers*, 13(2):147–162, 2000.

[20] Margaret A Shipp, Ken N Ross, Pablo Tamayo, Andrew P Weng, Jeffery L Kutok, Ricardo CT Aguiar, Michelle Gaasenbeek, Michael Angelo, Michael Reich, Geraldine S Pinkus, et al. Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature medicine*, 8(1):68–74, 2002.

[21] Robbert JC Slebos, Yajun Yi, Kim Ely, Jesse Carter, Amy Evjen, Xueqiong Zhang, Yu Shyr, Barbara M Murphy, Anthony J Cmelak, Brian B Burkey, et al. Gene expression differences associated with human papillomavirus status in head and neck squamous cell carcinoma. *Clinical Cancer Research*, 12(3):701–709, 2006.

[22] Miha Štajdohar. *Visualization and analysis of the space of prediction model*. PhD thesis, University of Ljubljana, Factulty of Computer and Information Science, 2012.

[23] Wikipedia. Scatter plot — Wikipedia, the free encyclopedia, 2014. [Online; accessed 21-August-2014].

[24] Tomohito Yagi, Akira Morimoto, Mariko Eguchi, Shigeyoshi Hibi, Masahiro Sako, Eiichi Ishii, Shuki Mizutani, Shinsaku Imashuku, Misao Ohki, and Hitoshi Ichikawa. Identification of a gene expression signature associated with pediatric aml prognosis. *Blood*, 102(5):1849–1856, 2003.