Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Doktorska disertacija

Vizualizacija podatkov s strojnim učenjem

Gregor Leban

Mentorja:
akad. prof. dr. Ivan Bratko
prof. dr. Blaž Zupan

Ljubljana, 2007

Povzetek

Vizualizacija podatkov je izredno pomembno orodje pri odkrivanju znanja v podatkih. Z vizualiziranjem prave podmnožice atributov s pravo vizualizacijsko metodo lahko jasno identificiramo zanimive in potencialno uporabne vzorce v podatkih. Vsi prikazi na žalost niso enako informativni in naloga uporabnika je, da poišče tiste, ki mu bodo omogočili najboljši vpogled v vsebovane zakonitosti.

Zbirke podatkov običajno vsebujejo veliko število atributov, zaradi česar obstaja tudi veliko število prikazov, ki si jih mora uporabnik ogledati. Da bi mu olajšali delo, smo v disertaciji razvili metodo VizRank, ki lahko avtomatsko izračuna oceno zanimivosti vsakega možnega prikaza klasificiranih podatkov. Prikaze lahko na osnovi te ocene rangiramo in nato podrobneje analiziramo zgolj majhno podmnožico najbolje ocenjenih prikazov. VizRank lahko uporabimo pri vsaki vizualizacijski metodi, ki primere vizualizira kot točke, katerih položaj je določen z vrednostjo atributov. Za ocenjevanje zanimivosti prikazov VizRank uporablja algoritem k-najbližjih sosedov (k-NN), s katerim oceni, kako dobro so primeri istega razreda združeni med sabo in ločeni od primerov iz drugih razredov. Klasifikacijsko točnost algoritma k-NN uporabimo za oceno zanimivosti projekcije. Na ta način dobijo projekcije z odlično ločenostjo razredov (visoka klasifikacijska točnost) visoko oceno zanimivosti, projekcije, v katerih se razredi prekrivajo, pa temu primerno nižjo oceno.

V disertaciji smo predstavili tudi metodi, ki omogočata avtomatsko ocenjevanje zanimivosti prikazov z metodo paralelnih koordinat ter mozaičnih diagramov. Prikaze z metodo paralelnih koordinat ocenimo glede na to, kako dobro lahko z njimi ločimo med razredi na osnovi položaja, kjer posamezne črte (primeri) sečejo koordinatne osi, ter na osnovi naklona črt med sosednjimi osmi. Mozaične diagrame ocenimo v dveh delih. Prikaze najprej ocenimo glede na izbor vizualiziranih atributov, s katerim je določena porazdelitev primerov po posameznih celicah diagrama. Večja kot je čistost posameznih celic, bolj je izbor atributov zanimiv. Za vsak izbor atributov lahko nato s preverjanjem, kako blizu ležijo celice s podobno porazdelitvijo razredov, ugotovimo, kateri vrstni red atributov je najprimernejši za vizualizacijo.

Uporabnost razvitih metod smo demonstrirali na različnih problemskih domenah iz repozitorija UCI in na zbirkah podatkov o raku.

Ključne besede: vizualizacija podatkov, strojno učenje, vizualno odkrivanje znanja v podatkih, eksplorativna analiza podatkov.

Abstract

Data visualization is a tool that has an enormous potential for extracting knowledge from data. Visualizing the right set of features in a right way can clearly identify interesting and potentially useful patterns. However, not all data projections are equally interesting and the task of a data miner is to find the most insightful ones.

Real life data sets often contain a large number of attributes which results in a large number of possible projection that the user has to inspect. To help the user we developed a method called VizRank, which can automatically compute an estimate of interestingness for each of possible projections of class labeled data. We can rank projections according to this score and then focus only on a small subset of best ranked projections, that will provide the greatest insight into the data. VizRank can be applied on any visualization method that maps attribute values to the position of a shown symbol. In order to estimate the interestingness of a given projection, VizRank uses k-nearest neighbor (k-NN) method to evaluate how well the points in the projection that have the same class label are grouped together and separated from the points with different labels. The accuracy of the k-NN is then used as the projection score. This way, the projections that show a perfect class separation (the accuracy of k-NN is high) receive the highest score, while the projections where some classes overlap receive correspondingly lower scores.

We also provide algorithms that allow automatic identification of interesting projections using the parallel coordinates method and mosaic plots. We rank parallel coordinate displays based on the ability to discriminate between classes by observing the intersecting position of the lines with the axes and by observing the angle of the lines between the neighbouring axes. The interestingness of mosaic plots is computed in two parts. First, we evaluate the selection of visualized attributes based on the distribution of class values in individual cells of the plot. Different attribute orders are then tested and the order with the best grouping of cells with the same class label is selected.

To demonstrate the usefulness of the developed algorithms we present results on data sets from UCI repository and from cancer microarray data analysis.

Key words: data visualization, machine learning, visual data mining, exploratory data analysis.

Zahvala

Najprej bi se rad zahvalil svojima mentorjema, akad. prof. dr. Ivanu Bratku ter izr. prof. dr. Blažu Zupanu, ki sta me usmerjala skozi celoten podiplomski študij. Njune številne ideje in predlogi so bistveno pripomogli h kvalitetnejši vsebini doktorata. Bila sta mi zgled dobrega raziskovalca in me s svojim entuziazmom ter zanimanjem dodatno motivirala za delo.

Za vso izkazano pomoč se zahvaljujem tudi doc. dr. Janezu Demšarju, ki mi je pomagal s svojimi idejami in temeljitim poznavanjem področja statistike ter strojnega učenja. V izredno pomoč mi je bilo tudi to, da sem lahko pri svojem raziskovanju uporabljal programski paket Orange, ki sta ga razvila skupaj z Blažem Zupanom.

Zahvalil bi se rad tudi vsem ostalim sodelavcem Laboratorija za umetno inteligenco, še posebej Aleksu Jakulinu, Tomažu Curku ter Aleksandru Sadikovu. Poleg prijetnih pogovorov so mi nudili tudi številne koristne nasvete, ki so mi pomagali pri delu.

Zahvala gre vsekakor tudi staršem in prijateljem, ki so mi ves čas pisanja doktorata stali ob strani, me spodbujali ter opominjali na to, kaj je v življenju resnično pomembno.

Nikakor pa ne najdem besed, s katerimi bi se lahko dovolj zahvalil Tini. V času, ko je nastajal doktorat, mi je nudila izjemno moralno podporo in pomoč brez katere bi težko shajal.

Offer to Me your every deed.
Devoid of egotism and desire,
inwardly centered in the soul,
ever calm and free from worries,
be dutifully engaged
in the battle of life.

Bhagavad Gita, 3:30

Kazalo

Poglavje 1

Uvod

Zbiranje podatkov je v zadnjih desetletjih prisotno na vseh področjih človeškega udejstvovanja. Na poslovnem področju naprimer shranjujemo informacije o tem, kaj potrošniki kupujejo, na področju socialnega varstva beležimo raznovrstne informacije o državljanih, v medicini o bolnikih, v genetiki pa podatke o tem, kako se pri različnih pogojih izraža tisoče genov v organizmu. V nekaterih primerih je zbiranje podatkov namenjeno zgolj evidenci, pogosteje pa jih nato uporabimo v analizi, s katero želimo odkriti zanimive in nepričakovane vzorce v podatkih. Tovrstni analizi podatkov pravimo odkrivanje znanja iz podatkov (ang. *data mining*), uporablja pa postopke iz strojnega učenja, razpoznavanja vzorcev, statistike, vizualizacije ter baz podatkov. Cilj analize je pridobivanje novega znanja (ang. *knowledge discovery*), ki nam omogoča boljše razumevanje podatkov ter vsebovanih zakonitosti.

Za odkrivanje znanja iz podatkov so nam na voljo številni postopki. Med drugim jih lahko delimo glede na to, kako aktiven in pomemben je uporabnik v procesu analize. Eno skrajnost v taki klasifikaciji predstavljajo avtomatski postopki iz strojnega učenja in statistike, ki delujejo samodejno, brez sodelovanja z uporabnikom. V primeru klasificiranih podatkov oziroma nadzorovanega učenja – t.j. učenja, pri katerem imamo v podatkih poleg atributov definiran tudi razred, katerega vrednosti se želimo naučiti čim pravilneje napovedati na osnovi vrednosti atributov – so primeri takih algoritmov nevronske mreže, metode najbližjih sosedov ter metode podpornih vektorjev. Njihova velika slabost je, da so dobljeni modeli človeku pogosto netransparentni in težko razumljivi – za človeka je praktično nemogoče razumeti pomen in vpliv posameznih uteži pri naučenih nevronskih mrežah ali pa si predstavljati, kako izgleda hiperravnina v visokodimenzionalnem prostoru. Modeli so v tem primeru črne škatle, v njihove napovedi pa je potrebno slepo zaupati.

Drugo skrajnost predstavljajo interaktivni postopki analize, pri katerih ima glavno besedo uporabnik, temeljijo pa na grafičnem prikazu podatkov. Ljudje imamo izredno razvite vizualno-kognitivne sposobnosti, ki nam omogočajo hitro in enostavno detekcijo

vzorcev. Z ustrezno vizualizacijo podatkov je zato mogoče odkriti različne zakonitosti, kot naprimer gruče primerov, osamelce, trende ter relacije med spremenljivkami. Tovrsten pristop k analizi je bistveno manj občutljiv na nehomogenost ter šum v podatkih in omogoča izkoriščanje dodatnega uporabnikovega znanja o problemski domeni. Interaktivna analiza podatkov je za uporabnika intuitivna in ne zahteva posebnega razumevanja kompleksnih matematičnih ter statističnih algoritmov in parametrov. Glavna prednost uporabe vizualizacije v primerjavi z avtomatskimi postopki pa je vsekakor njena interpretabilnost – odkrite zakonitosti lahko dejansko vidimo, zaradi česar je njihovo razumevanje neprimerno boljše.

Interaktivna analiza podatkov ima vsekakor tudi pomembne šibke točke. Različni prikazi podatkov so različno zanimivi in naloga uporabnika je, da ročno poišče najzanimivejše prikaze, ki bodo nudili najboljši vpogled v vsebovane zakonitosti. Ker je uporabnik pri iskanju zanimivih prikazov prepuščen samemu sebi, je tovrstno odkrivanje znanja v podatkih običajno zelo časovno potratno. Težava je namreč v tem, da imajo zbirke podatkov dandanes tipično veliko število atributov, zaradi česar obstaja ogromno število načinov, kako lahko z izbrano vizualizacijsko metodo vizualiziramo podatke. Že pri enostavnem razsevnem diagramu, s katerim prikažemo zgolj dva atributa hkrati, obstaja za podatke z n atributi $n(n-1)/2$ različnih diagramov, pri večdimenzionalnih vizualizacijskih metodah, s katerimi hkrati vizualiziramo večje število atributov, pa je možnih prikazov še bistveno več. Z večanjem števila možnih prikazov se zmožnost ročnega iskanja zanimivih prikazov manjša, zaradi česar se morajo uporabniki najpogosteje zadovoljiti zgolj z uporabo enostavnih (in omejenih) eno- in dvodimenzionalnih vizualizacijskih metod.

1.1 Cilj disertacije

Cilj disertacije je olajšati identifikacijo zanimivih prikazov z uporabo avtomatskih algoritmov, ki temeljijo na postopkih strojnega učenja. V ta namen smo razvili metodo VizRank, s katero je mogoče avtomatsko oceniti zanimivost različnih prikazov klasificiranih podatkov. Metoda je primerna za uporabo pri vsaki točkovni vizualizacijski metodi, pri kateri vrednosti vizualiziranih atributov določajo položaj prikazanega simbola v prikazu. Primera takih metod sta razsevni diagram ter metoda radviz (glej sliko 1.1). Pri klasificiranih podatkih je zanimivost prikaza odvisna od tega, kako dobro so v njem ločeni primeri različnih razredov – prikazi z dobro ločenostjo razredov omogočajo vizualno razpoznavanje pravil za ločevanje med razredi, zaradi česar so bolj zanimivi za podrobnejšo analizo kot prikazi s slabšo ločenostjo razredov. Za ocenjevanje tovrstnih prikazov VizRank najprej sestavi novo zbirko podatkov, ki vsebuje zgolj koordinate primerov v dvodimenzionalnem prikazu ter njihove vrednosti razredov – torej zgolj informacije, ki so uporabniku na voljo ob ogledu prikaza. Na tej zbirki podatkov nato z izbranim učnim algoritmom oceni napovedno točnost in jo uporabi kot oceno zanimivosti prikaza. Takšno ocenjevanje prikazov je smiselno, ker bo v prikazih z dobro ločenimi razredi točnost učnega algoritma visoka (in s tem visoka tudi ocena zanimivosti prikaza), v prikazih s slabšo ločenostjo

razredov pa temu primerno nižja. Z uporabo algoritma VizRank tako uporabniku zanimivih prikazov ne bo več potrebno iskati ročno, temveč se bo lahko osredotočil zgolj na majhno skupino najbolje ocenjenih prikazov.

V disertaciji bomo predstavili tudi načine, kako najbolje ocenjene prikaze uporabiti za pridobivanje dodatnega znanja o podatkih. Eden od primerov uporabe je naprimer ocenjevanje pomembnosti posameznih atributov za ločevanje med razredi. Pomembni atributi pomembno vplivajo na ločenost razredov v prikazu, zato lahko njihovo pomembnost enostavno ocenimo glede na to, kako pogosto so zastopani v najboljših prikazih. Takšno ocenjevanje atributov ni kratkovidno, saj upošteva potencialne interakcije med atributi v prikazu, hkrati pa ni preveč daljnovidno, saj upošteva zgolj prikaze s podmnožico atributov. Seznam najboljših prikazov lahko uspešno uporabimo tudi za iskanje osamelcev – izkaže se namreč, da taki primeri v najboljših prikazih zelo pogosto ležijo na robu svoje gruče primerov ali celo med primeri drugega razreda.

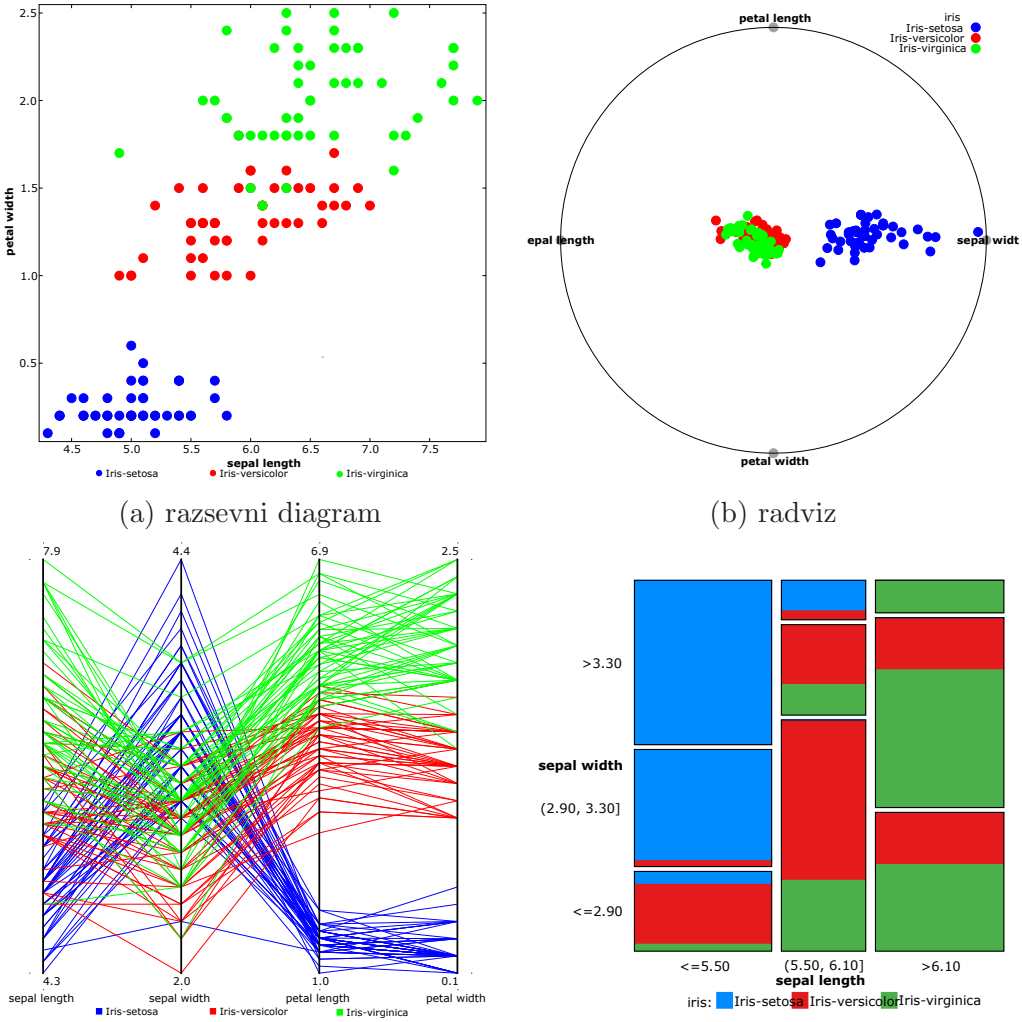
Poleg ocenjevanja prikazov s točkovnimi metodami,bomo v disertaciji predstavili tudi postopke za ocenjevanje mozaičnih diagramov ter prikazov s paralelnimi koordinatami (slika 1.1.c in 1.1.d). Pri paralelnih koordinatah so atributi predstavljeni kot paralelne osi, primeri pa kot črte, ki potekajo med prvo in zadnjo osjo. Najbolj so pri taki vizualizaciji opazne relacije med sosednjimi atributi, zato je bistvenega pomena, kako atribute izberemo in uredimo. Predstavili bomo dva algoritma za iskanje zanimivih prikazov – prvi temelji na uporabi razsevnih diagramov, drugi pa na prikazih radviz. Za vizualizacijo diskretnih atributov so najprimernejši mozaični diagrami, ki različne vrednosti vizualiziranih atributov prikažejo s celicami, njihova velikost pa ustreza številu primerov z dano kombinacijo vrednosti atributov. V primeru klasificiranih podatkov lahko posamezne celice dodatno pobarvamo glede na porazdelitev primerov po razredih. Ker so najzanimivejši tisti mozaični diagrami, v katerih so posamezne celice čim bolj “čiste” (vsebujejo zgolj primere enega razreda), smo razvili algoritem, ki prikaze oceni glede na ta kriterij. Poleg tega bomo v disertaciji predstavili tudi postopek, ki za dani izbor atributov pregleda različne postavitve teh atributov v diagramu in predlaga tisto postavitev, iz katere je najlaže opaziti skupne zakonitosti, ki veljajo za različne podskupine celic.

1.2 Vsebina disertacije

Organizacija disertacije je naslednja.

V 2. poglavju bomo predstavili področje vizualizacije podatkov. Podali bomo taksonomijo vizualizacijskih metod ter na kratko opisali pomembnejše med njimi. Predstavili bomo obstoječe postopke, ki omogočajo avtomatsko iskanje zanimivih linearnih projekcij podatkov. Izbranih šest vizualizacijskih metod bomo uporabili na petih problemskih domenah ter ocenili njihovo uspešnost pri nalogah, s katerimi se pri analizi klasificiranih podatkov najpogosteje srečujemo.

V 3. poglavju bomo predstavili metodo VizRank, s katero lahko ocenimo zanimivost prikazov, generiranih s poljubno točkovno vizualizacijsko metodo. Opisali bomo pomem-



(a) razsevni diagram (b) radviz

(c) paralelne koordinate (d) mozaični diagram

Slika 1.1: Primer štirih vizualizacijskih metod, ki bodo obravnavane v disertaciji, za podatke iris.

bne parametre metode ter njihov vpliv na ocenjevanje prikazov. Predstavili bomo hevristiko, s katero v primeru velikega števila možnih prikazov zelo učinkovito identificiramo najzanimivejše prikaze. Prikazali bomo rezultate eksperimenta, ki je pokazal veliko ujemanje med človeškim rangiranjem prikazov ter rangiranjem, ki ga določi VizRank. Predstavili bomo načine, s katerimi lahko projekcije uporabimo za ocenjevanje pomembnosti posameznih atributov, odkrivanje interakcij med atributi ter za iskanje osamelcev. Opisali bomo tudi rezultate poskusa, kjer smo projekcije uporabili kot napovedne modele in z njimi dosegli zelo visoko napovedno točnost.

V 4. poglavju bomo princip rangiranja prikazov glede na njihovo zanimivost razširili tudi na metodo paralelnih koordinat. Zaradi pomembnih dualnih lastnosti z evklidskim

prostorom, bomo zanimivost prikaza definirali kot vsoto zanimivosti vseh sosednjih parov atributov. Definirali bomo različne kriterije zanimivosti parov atributov ter opisali algoritem, s katerim lahko poiščemo najzanimivejše prikaze.

V 5. poglavju bomo predstavili postopek, ki omogoča ocenjevanje zanimivosti mozaičnih diagramov. Najprej bomo podali kriterije, s katerimi lahko na različne načine ocenimo, kako uspešno so razredi v diagramu ločeni pri vizualizaciji različnih podmnožic atributov. Opisali bomo tudi postopek, ki za dani izbor atributov poišče najprimernejši vrstni red atributov.

V zaključnem poglavju bomo izpostavili ključna dognanja disertacije ter opisali nekaj možnosti za nadaljnje delo.

1.3 Prispevki k znanosti

Disertacija vsebuje izvirne prispevke na področjih vizualizacije podatkov ter odkrivanja znanja v podatkih. Razvili smo postopke, s katerimi je mogoče avtomatsko oceniti zanimivost prikazov z različnimi podmnožicami atributov. Ti postopki so bili razviti za vizualizacijske metode, ki so se pri analizi podatkov izkazale kot najuspešnejše. Opisali smo tudi različne uporabe najboljših najdenih prikazov za odkrivanje dodatnega znanja.

Glavni prispevki disertacije so naslednji:

- Izdelali smo podrobno analizo pomembnejših vizualizacijskih metod ter ocenili njihovo uspešnost pri prikazu različnih zakonitosti v podatkih.
- Za različne vizualizacijske metode smo razvili postopke za avtomatsko ocenjevanje zanimivosti prikazov pri analizi klasificiranih podatkov. Prikaze je mogoče urediti glede na njihovo zanimivost, kar uporabniku omogoča, da zanimivih prikazov ne išče ročno, ampak se omeji na podrobnejšo analizo zgolj majhne podmnožice najbolje ocenjenih prikazov, ki nudijo najboljši vpogled v vsebovane zakonitosti.
- Razvili smo različne hevristike, ki omogočajo zelo uspešno preiskovanje prostora možnih prikazov in identifikacijo najzanimivejših prikazov.
- Predstavili smo postopek za ocenjevanje pomembnosti atributov, odkrivanje interakcij med atributi ter identifikacijo osamelcev na podlagi najboljših prikazov podatkov.
- Predstavili smo način za uporabo vizualnih prikazov kot napovednih modelov za klasifikacijo novih primerov. V nasprotju s številnimi metodami strojnega učenja, ki zgradijo težko interpretabilne modele, predstavljajo prikazi točne in hkrati intuitivno razumljive modele.
- Vizualizacijske metode ter vse postopke namenjene iskanju zanimivih prikazov smo implementirali v sistemu Orange [25].

Poglavje 2

Vizualizacija podatkov

V tem poglavju bomo predstavili področje vizualizacije podatkov. Podali bomo taksonomijo obstoječih vizualizacijskih metod ter opisali pomembnejše metode. Za primer linearnih projekcij bomo opisali množico obstoječih postopkov, ki omogočajo iskanje zanimivih prikazov podatkov. Nazadnje bomo za šest izbranih vizualizacijskih metod ocenili njihovo uporabnost na petih problemskih domenah iz strojnega učenja.

2.1 Uvod

Ljudje smo običajno zelo neuspešni pri analizi podatkov, če so le-ti prikazani v tekstovni ali numerični obliki. Le s težavo odkrijemo zanimive lastnosti podatkov, pa naj si bodo to osnovne statistične informacije o spremenljivkah, relacije med spremenljivkami, da iskanja osamelcev ali potencialnih podskupin primerov niti ne omenjamo. Veliko bolje se človek odreže pri teh in številnih drugih nalogah, če podatke prikažemo v grafični obliki. Področju, ki se ukvarja z načini za tako predstavitev podatkov, pravimo vizualizacija podatkov in ga običajno opredelimo kot "uporaba računalniško-podprte, interaktivne, grafične reprezentacije podatkov za izboljšanje kognicije" [14]. Glavni namen vizualizacije je torej uspešno izkoristi izredne človeške sposobnosti percepcije in prikazati podatke na tak način, da bodo postali pomembni vzorci v podatkih nemudoma vidni.

Celotno področje vizualizacije podatkov lahko podrobneje razdelimo na dve podpodročji: znanstveno vizualizacijo ter vizualizacijo informacij (ang. *scientific/information visualization*). Pri znanstveni vizualizaciji gre za vizualizacijo grafičnih modelov, ki so običajno konstruirani iz izmerjenih ali simuliranih podatkov in predstavljajo objekte ali koncepte povezane s pojavi v fizikalnem svetu. Relacije med atributi so v tem primeru dobro znane in razumljive. Primeri znanstvene vizualizacije so vizualizacija vremenskih podatkov, gibanja tekočin, geoloških/geofizikalnih podatkov, molekularnih struktur ter medicinskih podatkov (CTG, ultrazvok). Vizualizacija informacij se na drugi strani

ukvarja s prikazom abstraktnih podatkov, za katere zelo pogosto ne obstaja nobena predstavitev v fizikalnem svetu. Relacije med atributi so v tem primeru razumljive slabo ali pa sploh niso. Primeri takih podatkov so podatki o gibanju tečajev delnic, prodanih avtomobilih ter izraženosti posameznih genov v organizmu. V področje vizualizacije informacij spada tudi vizualizacija abstraktnih struktur, kot so naprimer drevesa in grafi. V disertaciji se bomo omejili zgolj na vizualizacijo informacij, saj je ta smer vizualizacije edina primerna za vrsto podatkov, s katerimi se bomo pri analizi srečevali.

Vizualizacijo informacij lahko uspešno uporabimo v različnih fazah analize podatkov. V začetni (raziskovalni) fazi analize nam naprimer vizualizacija omogoča, da uporabnik med interaktivnim spreminjanjem prikaza pridobi splošen vpogled v podatke ter postavi nove hipoteze. V potrditveni fazi nam vizualizacija lahko pomaga pri potrditvi ali zavračanju hipotez, ki smo jih pridobili z različnimi avtomatskimi postopki iz strojnega učenja ter statistike. Vizualizacijo lahko uspešno uporabimo tudi v fazi same predstavitve, kjer je cilj na čim bolj jasen in razumljiv način prikazati neko znano informacijo, ki jo podatki vsebujejo.

2.1.1 Zgodovinski razvoj

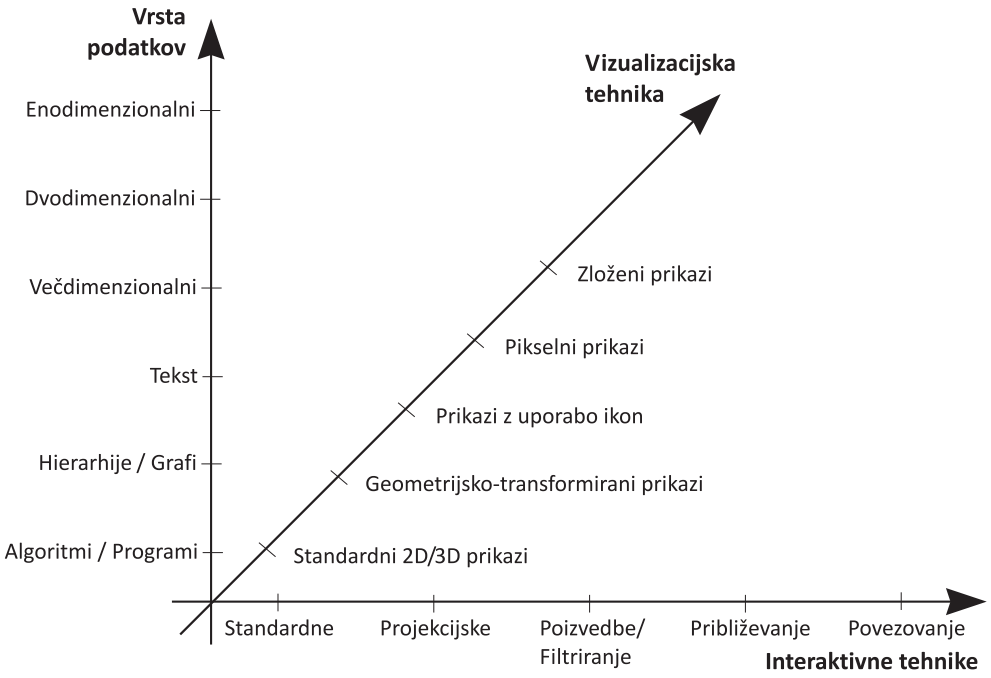
Zgodovinsko gledano se razvoj področja vizualizacije informacij deli na štiri faze [110]. Mejnike tem fazam predstavljajo izdaja knjige *Exploratory data analysis* [106] leta 1977, NSF delavnica na temo vizualizacije [79] leta 1987 ter *IEEE* konferenca na temo vizualizacije [81] leta 1991.

V prvem obdobju so se z vizualizacijo v glavnem ukvarjali statistiki. Prikazi so bili običajni *x-y* diagrami, izrisani na papirju z barvnimi svinčniki. Za to obdobje so značilni prikazi majhne količine podatkov. Vizualizacija je namenjena prikazu glavnih značilnosti podatkov, pomagala naj bi pri izbiri statističnih metod za obdelavo podatkov ter uspešno prikazala zaključke.

Z izdajo Tuckeyeve knjige *Exploratory data analysis* [106] postane vizualizacija več kot le grafična predstavitev podatkov – postane način za odkrivanje zakonitosti v podatkih. Prihod osebnih računalnikov ter barvnih monitorjev omogoči analitikom interaktivno analizo podatkov v realnem času. Večinoma so še vedno v uporabi le dvodimenzionalni prikazi podatkov.

Naslednji mejnik je delavnica *NSF Workshop on Scientific Computing* [79] leta 1987, kjer se je tudi uradno potrdila nujnost večdimenzionalnih vizualizacijskih metod. Večina raziskav se v tem obdobju preusmeri od statističnih metod raziskovalne analize k razvoju novih, računsko zahtevnih postopkov za generiranje interaktivnih vizualizacij. Razvijejo se številne nove vizualizacijske metode, ki na različne načine omogočajo vizualizacijo večjega števila atributov.

V zadnjem obdobju, ki poteka od leta 1992 dalje, se raziskovalci izogibajo razvoju novih vizualizacijskih metod. Glavni cilj v tem obdobju je ocenitev obstoječih vizualizacijskih metod s stališča njihove pravilnosti, učinkovitosti in uporabnosti pri analizi podatkov.



Slika 2.1: Kriteriji za klasifikacijo vizualizacijskih metod

2.2 Taksonomija vizualizacijskih metod

Za vizualizacijo informacij je bilo v preteklih letih razvitih veliko metod. Da bi bolje razumeli podobnosti in razlike med njimi, so bili narejeni številni poskusi klasifikacij metod glede na njihove lastnosti. Keim [60] je predlagal razvrstitev metod glede na tri kriterije: glede na vrsto podatkov, ki jih lahko vizualiziramo, glede na tip vizualizacijske metode ter glede na interaktivne tehnike, ki jih lahko uporabimo. Izbrani kriteriji so še posebej smiselni zato, ker jih lahko smatramo kot medsebojno ortogonalne – različne vrste vizualizacijskih metod lahko poljubno kombiniramo z različnimi interaktivnimi tehnikami in jih apliciramo na različnih vrstah podatkov.

Na sliki 2.1 je prikazana klasifikacija vizualizacijskih metod glede na omenjene tri kriterije. Sledi krajši opis skupin metod za vsakega od kriterijev.

2.2.1 Klasifikacija glede na vrsto podatkov

Podatki, ki jih želimo vizualizirati, običajno vsebujejo večje število primerov, pri čemer je vsak primer predstavljen z določenim številom spremenljivk oziroma atributov. Take podatke ločimo na eno-, dvo- ter večdimenzionalne. Poleg teh poznamo tudi kompleksnejše tipe podatkov, kot so naprimer tekst in grafi. Vizualizacijske metode lahko torej ločimo glede na naslednje vrste podatkov:

Enodimenzionalni podatki. Tipičen primer enodimenzionalnih podatkov so časovni podatki. Za vsako časovno točko obstaja ena ali več vrednosti. Primer prikaza je

vizualizacija gibanja borznih podatkov z uporabo linijskega diagrama, v katerem so povezane sosednje časovne točke.

Dvodimenzionalni podatki. Primer dvodimenzionalnih podatkov so zemljepisni podatki, pri katerih sta atributa zemljepisna širina in dolžina. Tipična metoda za vizualizacijo takih podatkov so x-y diagrami.

Večdimenzionalni podatki. Večina baz podatkov, s katerimi se srečujemo danes, vsebuje več kot dva atributa. Primer izrazito visokodimenzionalnih podatkov so podatki iz mikromrež, ki tipično vsebujejo več tisoč atributov. V takih primerih ne obstaja enostavna preslikava vseh atributov v dve dimenziji, zato so potrebne naprednejše vizualizacijske metode. Primer take metode je metoda paralelnih koordinat, pri kateri so atributi predstavljeni kot osi, postavljene paralelno ena ob drugi.

Besedila. V današnji dobi interneta so besedila vse pomembnejši tip podatkov. Ker teksta v originalni obliki ni mogoče vizualizirati, je najprej potrebno uporabiti različne postopke za njegovo pretvorbo v obliko opisnih vektorjev, ki jih nato najpogosteje vizualiziramo z uporabo analize osnovnih komponent ali večdimenzionalnega skaliranja.

Hierarhije/Grafi. Pogosto so podatki predstavljeni v obliki relacij med objekti. Primer takih podatkov so povezave med internetnimi stranmi ter struktura direktorijev in datotek v računalniku. Tovrstni podatki so običajno prikazani z grafi. Grafi so sestavljeni iz vozlišč, ki predstavljajo objekte, ter povezav, ki predstavljajo relacije med objekti.

Algoritmi/Programi. Obvladovanje velikih programskih projektov je zahtevno opravilo. Vizualizacijo lahko v tem primeru uporabimo za prikaz pretoka podatkov v programu, skokov in klicev posameznih funkcij ter strukture in starosti izvorne kode [105]. Na sliki 2.2 je prikaz iz programa SeeSoft [27], kjer je vsaka datoteka vizualizirana kot stolpec. Vsaka vrstica kode je prikazana v stolpcu s tanko črto, katere barva odraža, kako pogosto se vrstica izvede pri izvajanju programa – temneje obarvane vrstice se izvedejo pogosteje kot svetleje obarvane. S to vizualizacijo enostavno identificiramo dele kode, ki jih je smiselno dodatno pohitriti.

2.2.2 Klasifikacija glede na tip vizualizacijske metode

Različne vizualizacijske metode uporabljajo različne principe za preslikavo podatkov v grafično obliko. Glede na uporabljen princip ločimo med naslednjimi tipi vizualizacijskih metod:

Standardni 2D/3D prikazi. V to skupino spadajo dobro poznane in pogosto uporabljene metode, kot so linijski diagrami, histogrami, razsevni diagrami, itd. Večina od teh metod izvira iz statistike.

Slika 2.2: Vizualizacija programske kode

Geometrijsko-transformirani prikazi. Te metode uporabljajo različne transformacije podatkov, s pomočjo katerih se doseže vizualizacija večjega števila atributov. Primeri takih metod so metoda paralelnih koordinat [54], radviz [50] ter matrika razsevnih diagramov [91]. Poleg teh spadajo v to skupino tudi metode za zmanjševanje dimenzionalnosti podatkov, kot naprimer analiza glavnih komponent [51], večdimenzionalno skaliranje [19, 8] ter projekcijsko iskanje [70, 104]. Nekatere od teh metod so podrobneje opisane v razdelku 2.4.

Prikazi z uporabo ikon. Pri teh metodah so primeri vizualizirani kot "ikone", vrednosti atributov pa so prikazane z različnimi oblikami in barvami posameznih delov teh ikon. Ikone lahko definiramo poljubno – znani so primeri metod, kjer so kot ikone uporabljene različne oblike obrazov [16], zvezd [15], paličnih figur [84, 44] ter barvnih pravokotnikov [76]. V primeru paličnih figur je vsak primer predstavljen kot lik, sestavljen iz manjšega števila povezanih palic, pri čemer kot med posameznimi palicami odraža vrednost določenega atributa.

Pikselni prikazi. Ti prikazi so namenjeni vizualizaciji velikega števila podatkov. Celotno področje prikaza se običajno najprej razdeli na manjša področja, tako da vsakemu atributu pripada del prikaza. Znotraj posameznega področja se nato vrednost vsakega primera ponazori z barvanjem natanko enega piksla na zaslonu. Znana primera pikselnih prikazov sta metoda rekurzivnih vzorcev ter metoda krožnih segmentov (glej razdelek 2.3.4).

Skladovni prikazi. Pri skladovnih prikazih se v en koordinatni sistem vstavi nov koordinatni sistem, s čemer se poveča število vizualiziranih atributov. Podatki so tako prikazani v hierarhični obliki, zaradi česar je pomembno, da pazljivo izberemo vrstni red vizualiziranih atributov. Primera takih metod sta dimenzijsko skladanje ter svetovi-znotraj-svetov (glej razdelek 2.3.8).

2.2.3 Klasifikacija glede na interaktivne tehnike

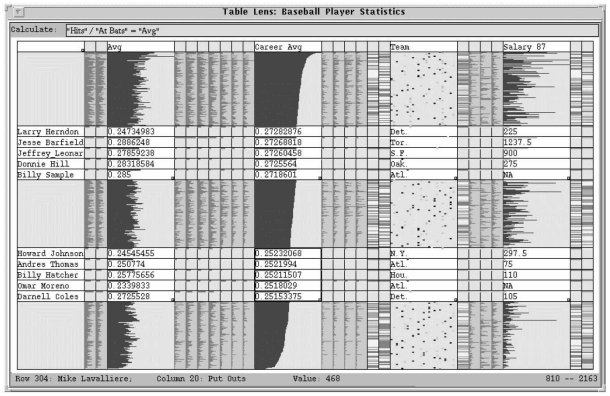
Interaktivne tehnike so postopki, s katerimi lahko uporabnik dinamično vpliva na prikaz podatkov. Te tehnike omogočajo izboljšanje uporabnikove percepcije prikazane informacije, zato so vsaj nekatere od teh tehnik na voljo pri večini vizualizacijskih metod. Uporaba interaktivnih tehnik bistveno zmanjša slabosti posameznih metod, še posebej tistih slabosti, ki so posledica prekrivanja primerov. Posledično lahko zaradi tega uspešneje vizualiziramo večje število primerov.

Standardne. Standardne interaktivne tehnike omogočajo uporabnikom enostavno izbiranje in urejanje vizualiziranih atributov.

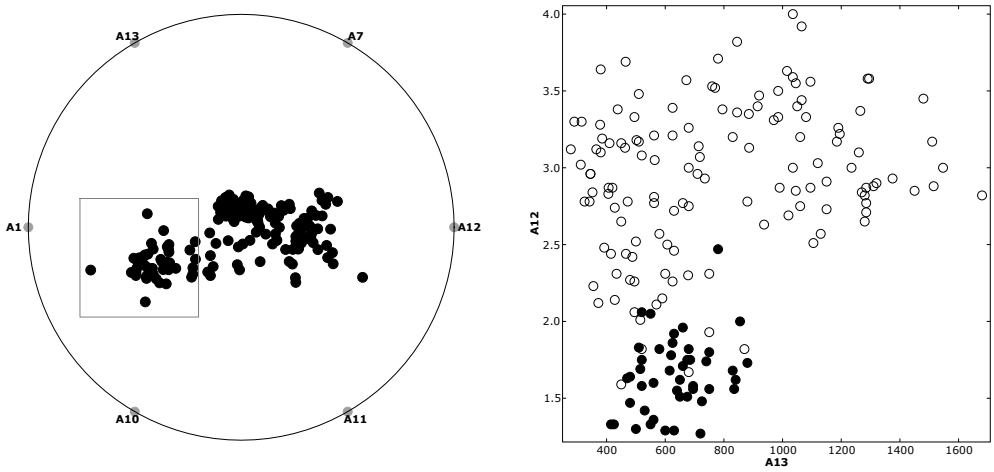
Projekcijske. Projekcijske tehnike omogočajo dinamično spreminjanje projekcij z namenom odkrivanja zanimivih pogledov dane zbirke podatkov. Tipičen primer projekcijske tehnike je metoda Grand Tour [2] (glej razdelek 2.3.9), ki podatke prikaže v x-y diagramu, v katerem je vsaka od osi utežena linearna kombinacija vseh atributov. Uteži pri posameznih atributih se dinamično spreminjajo, s čimer se zvezno spreminja tudi položaj točk v projekciji. Na ta način je v projekcijah mogoče odkriti gruče točk, osamelce ter druge strukture v podatkih. Spreminjanje uteži pri atributih je lahko ročno, naključno ali pa usmerjeno, z namenom odkrivanja točno določene strukture v podatkih. Primeru usmerjenega spreminjanja uteži pravimo projekcijsko iskanje in je podrobneje opisano v razdelku 2.4.5.

Poizvedbe in filtriranje (ang. querying & filtering). Pri analizi velikih zbirk podatkov pogosto želimo izbrati neko podmnožico primerov in jih podrobneje analizirati. Filtriranje je operacija, ki nam omogoča, da podmnožico primerov izberemo interaktivno s pomočjo vizualizacije. Poizvedba je sorodna operacija, pri kateri cilj ni odstranitev neizbranih primerov, ampak prikaz dodatne informacije za izbrane primere. Primer poizvedb so magične leče (ang. *Magic Lenses*) ter tabelarične leče (ang. *Table Lenses*) [101]. Magične leče uporabljajo povečevalno steklo, s katerim se podatki pod povečevalnim steklom prikažejo drugače kot preostali podatki. Pri tabelaričnih lečah pa so podatki prikazani z uporabo stolpičnih diagramov, hkrati pa lahko za izbrane podmnožice primerov prikažemo dejanske vrednosti primerov (glej sliko 2.3).

Približevanje (ang. zooming). Približevanje omogoča postopno odkrivanje podrobnosti v posameznih delih prikaza. Še posebej je pomembno pri vizualizaciji velike količine podatkov, kjer se primeri prekrivajo. V praksi je pogosto koristno, če približevanje ne omogoča zgolj povečave določenega dela prikaza, ampak glede na stopnjo približanja prilagodi tudi količino prikazane informacije. Temu pravimo logično približevanje (ang. *logical zooming*) in smo ga še posebej vajeni pri vizualizaciji zemljevidov. Ljubljana je videti kot pika na zemljevidu Evrope, kot mnogokotnik na zemljevidu Slovenije, pri dodatnem približevanju pa lahko pridemo vse do vizualizacije posameznih ulic v mestu.



Slika 2.3: Tabelarične leče



Slika 2.4: Primer povezovanja med prikazom radviz in razsevnim diagramom na podatkih wine. Primeri, ki jih je uporabnik izbral v prikazu radviz (levo), so označeni tudi v razsevnem diagramu (desno).

Povezovanje (ang. linking). Povezovanje je tehnika, pri kateri izbor enega ali več primerov v enem prikazu povzroči označitev teh primerov tudi v drugih prikazih. Primere, ki so nam zaradi nekega razloga zanimivi, lahko tako podrobneje analiziramo še z uporabo drugih vizualizacijskih metod ter z drugimi vizualiziranimi atributi. Primer povezovanja je na sliki 2.4, ki prikazuje dva prikaza zbirke podatkov wine. V prikazu radviz (levi prikaz) je uporabnik izbral podskupino primerov, ti pa se nato samodejno označijo tudi v razsevnem diagramu (desni prikaz).

2.3 Opis pomembnejših vizualizacijskih metod

V tem razdelku bomo opisali vizualizacijske metode, ki se pogosto pojavljajo v literaturi. Omejili se bomo na metode, ki so namenjene vizualizaciji dvo- in večdimenzionalnih podatkov in se izognili metodam za vizualizacijo grafov, hierarhij in ostalih vrst podatkov. Največ poudarka bomo namenili metodam, ki bodo uporabljene v nadaljevanju disertacije – torej razsevnim diagramom, metodi radviz, metodi paralelnih koordinat ter mozaičnemu diagramu.

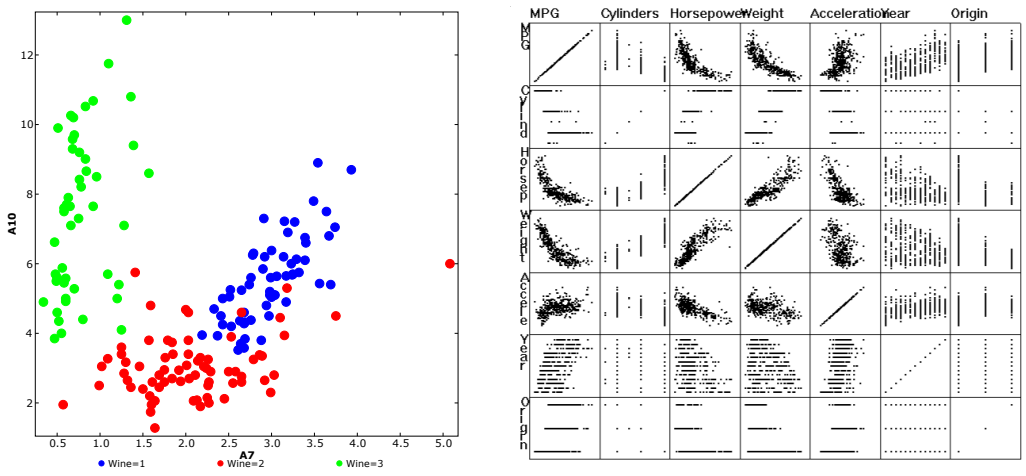
2.3.1 Razsevni diagram

Razsevni diagram (ang. *scatterplot*) je ena najbolj znanih in najpogosteje uporabljenih vizualizacijskih metod. V osnovni obliki je namenjena prikazu relacije med dvema atributoma, pri čemer je en atribut prikazan na x osi, drugi pa na y osi. Podatki so prikazani kot točke, katerih položaj je določen z vrednostmi vizualiziranih atributov. Zaradi načina prikaza je metoda zelo primerna za odkrivanje gruč, trendov, osamelcev ter korelacij med atributi.

Število vizualiziranih atributov je pri razsevnem diagramu mogoče povečati tako, da vrednosti dodatnih atributov prikažemo z velikostjo, barvo in obliko simbolov, ki ponazarjajo primere. Pri tem se je potrebno zavedati, da je človeška percepcija teh značilk bistveno slabša od percepcije položaja točk v prikazu [12, 13]. Dodatno težavo predstavlja dejstvo, da pri vizualizaciji pogosto pride do prekrivanja posameznih točk, kar dodatno oteži razpoznavanje teh značilk. Zaradi omenjenih razlogov je smiselno, da najpomembnejša atributa vedno vizualiziramo na x in y osi, ter da tudi sicer ostanemo zmerni pri uporabi omenjenih lastnosti simbolov za vizualizacijo dodatnih atributov.

Na sliki 2.5.a je primer razsevnega diagrama za dva atributa iz zbirke podatkov wine [80], ki vsebujejo rezultate kemične analize vin iz določene italijanske regije. Vrednost diskretnega razreda je prikazana z barvo točk. V prikazu so različne vrste vina dokaj dobro ločene, zaradi česar lahko enostavno induciramo pravila za ločevanje med njimi.

Ena od enostavnih razširitev metode se imenuje matrika razsevnih diagramov in je namenjena hkratnemu prikazu razsevnih diagramov z vsemi pari atributov. Matrika je dimenzij $n \times n$, pri čemer je n število atributov v zbirki podatkov. Vsak element (i, j) te matrike je razsevni diagram, kjer sta na x in y osi i-ti in j-ti atribut. Matrika razsevnih diagramov je primerna za hitro odkrivanje korelacij med različnimi atributi, težava metode pa je v tem, da je primerna le pri zelo majhnem številu atributov (< 10) saj postanejo sicer posamezni razsevni diagrami premajhni. Na sliki 2.5.b je matrika razsevnih diagramov za podatke o avtomobilih. Iz prikaza je lepo vidna pozitivna korelacija med močjo (*Horsepower*) in težo (*Weight*) avtomobila ter negativna korelacija med številom milj na galono (*MPG*) ter močjo in težo avtomobila.



(a) (b)

Slika 2.5: (a) Razsevni diagram za podatke wine. (b) Matrika razsevnih diagramov za podatke o avtomobilih.

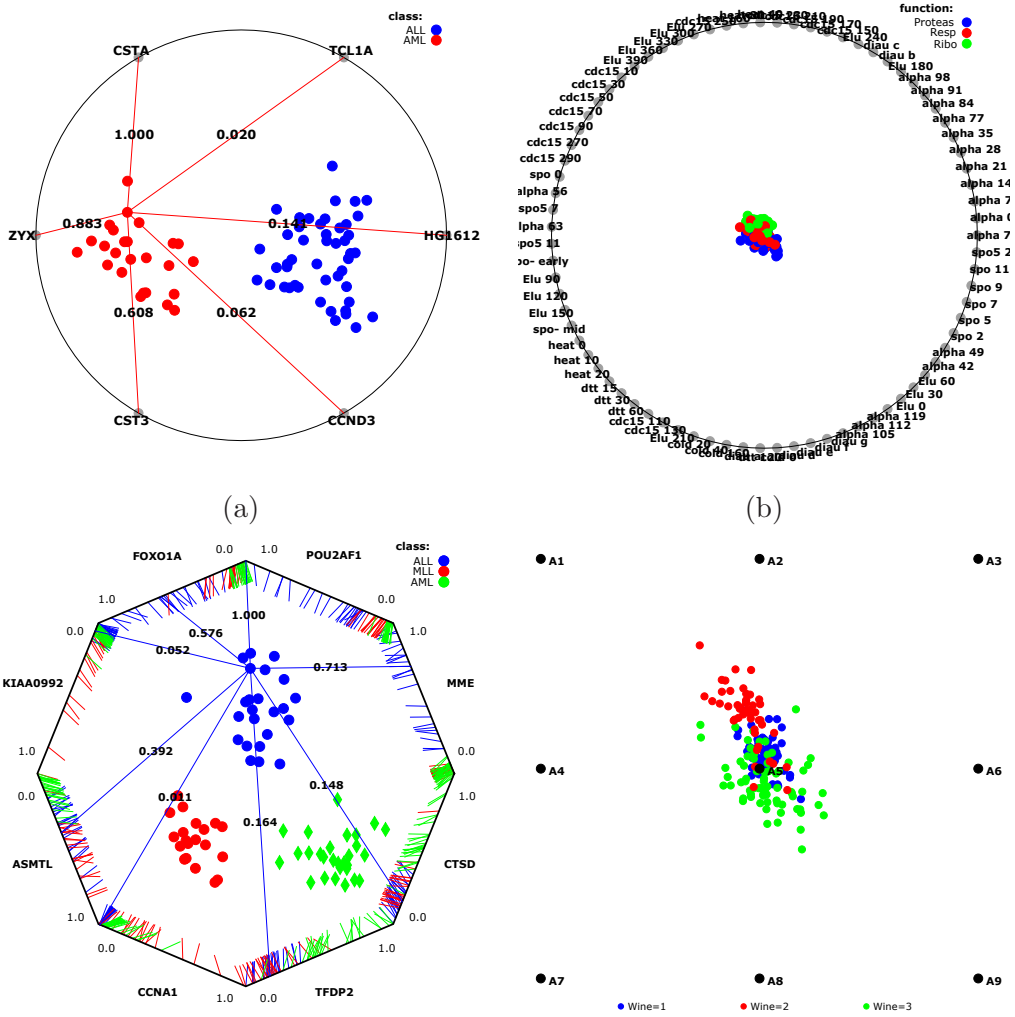
2.3.2 Radviz

Radviz [50, 45, 10] je okrajšano ime za *radial coordinate visualization*. Metoda omogoča vizualizacijo poljubnega števila atributov, pri čemer so atributi predstavljeni kot točke, enakomerno razporejene po obodu enotske krožnice. Tem točkam pravimo dimenzijska sidra (ang. *dimensional anchors*). Primeri so prikazani kot točke, ki ležijo znotraj krožnice. Za razumevanje, kako je določen položaj primerov, je smiselno uporabiti prispodobo z več vzmetmi. Recimo, da želimo vizualizirati n atributov. Vsakemu primeru pripada n vzmeti, ki so na eni strani pritrjene naprimer, na drugi strani pa na enega od dimenzijskih sider (atributov). Trdnost vzmeti (v smislu Hookovega zakona) je določena z vrednostjo atributa, na katerega je vzmet pritrjena – večja kot je vrednost atributa pri primeru, večja je trdnost vzmeti. Vsak atribut torej privlači primer s silo, ki je sorazmerna vrednosti atributa pri tem primeru. Točka, ki ustreza primeru, je nato prikazana tam, kjer je vsota vseh sil nanjo enaka nič.

Natančnejša matematična definicija za določanje položaja primerov je naslednja. Imejmo n atributov, predstavljenih z dimenzijskimi sidri $S_i = (S_{i,x}, S_{i,y})$, za $i = 1..n$. Vrednosti $S_{i,x}$ in $S_{i,y}$ predstavljata x in y koordinati i-tega sidra. Vzemimo primer v z vrednostmi $(v_1, v_2, ..., v_n)$, ki bi ga radi vizualizirali. Njegovi x in y koordinati izračunamo po naslednji formuli:

$$x = \sum_{i=1}^{n} S_{i,x} \cdot \frac{v_i}{\sum_{i=1}^{n} v_i} = \frac{\sum_{i=1}^{n} S_{i,x} \cdot v_i}{\sum_{i=1}^{n} v_i}$$

$$y = \sum_{i=1}^{n} S_{i,y} \cdot \frac{v_i}{\sum_{i=1}^{n} v_i} = \frac{\sum_{i=1}^{n} S_{i,y} \cdot v_i}{\sum_{i=1}^{n} v_i}.$$



Slika 2.6: Prikaza z metodo radviz s šestimi (a) ter 79 atributi (b). Prikaza z metodo polyviz (c) ter gridviz (d).

Položaj točke je torej utežena srednja vrednost položajev sider S_i, pri čemer so uteži določene tako, da je njihova vsota enaka 1. Zaradi te normalizacije je preslikava iz $\mathbb{R}^n \rightarrow \mathbb{R}^2$ nelinearna.

Primer prikaza radviz s šestimi atributi je na sliki 2.6.a. Za izbran primer so simbolično prikazane tudi vzmeti ter njihove trdnosti (vrednosti atributov). Iz položaja primerov v prikazu lahko sklepamo, da imajo primeri iz razreda *ALL* velike vrednosti pri atributih *CCND3*, *HG1612* ter *TCL1A* in majhne vrednosti pri *CSTA*, *ZYX* ter *CST3*, med tem, ko velja za primere iz razreda *AML* ravno obratno.

Pred vizualizacijo je običajno potrebno podatke normalizirati. Tipično se v ta namen uporablja lokalna normalizacija, pri kateri se vrednosti vsakega atributa preslikajo na

interval med 0 in 1. S tem se doseže, da imajo vsi vizualizirani atributi enak vpliv pri določanju položaja točk v prikazu. V primeru, da podatkov ne bi normalizirali, bi imeli atributi z velikim razponom vrednosti večji vpliv na položaj točk kot atributi z majhnim razponom. Poleg tega bi negativne vrednosti atributov lahko povzročile, da primeri ne bi ležali znotraj temveč zunaj krožnice.

Glede na način, kako določimo položaje primerov, lahko povzamemo nekatere lastnosti preslikave:

- Točke s približno enakimi vrednostmi atributov (po normalizaciji) ležijo blizu središča kroga.
- Točke s približno enakimi vrednostmi atributov, ki si ležijo na krožnici nasproti, tudi ležijo blizu središča kroga.
- Točke, ki imajo pri enem ali dveh atributih večje vrednosti, kot pri ostalih, ležijo bližje dimenzijskim sidrom, ki predstavljajo ta atributa.
- Sfera se preslika v elipso.
- n-dimenzionalna črta se preslika v črto.
- n-dimenzionalna ravnina se preslika v poligon.

Položaj primerov v prikazu radviz ni odvisen samo od izbora vizualiziranih atributov, ampak tudi od njihove ureditve na krožnici. Možnih postavitev n atributov na n dimenzijskih sider je $n!$, vendar so si nekatere postavitve ekvivalentne (generirajo prikaze, ki so rotirane ali zrcaljene različice drugih prikazov). Tako naprimer za vsako postavitev atributov obstaja n ekvivalentnih postavitev, ki jih dobimo tako, da vsa dimenzijska sidra rotiramo za $i \cdot 360/n$ stopinj, pri čemer je i lahko med 1 in n. Ekvivalentno postavitev atributov dobimo tudi v primeru, da obrnemo vrstni red, v katerem so atributi na krožnici urejeni. Pri n izbranih atributih je tako število različnih prikazov, ki jih lahko dobimo z različno postavitvijo atributov, $\frac{n!}{n \cdot 2} = \frac{(n-1)!}{2}$.

Čeprav metoda radviz omogoča vizualizacijo poljubnega števila atributov, je uporabnost prikaza z večjim številom atributov (> 10) vprašljiva. Interpretacija takega prikaza je zelo zahtevna, saj je iz položaja točk v prikazu težko sklepati o vrednostih posameznih atributov in o lastnostih, ki naj bi veljale za posamezne gruče točk. Poleg tega z večanjem števila vizualiziranih atributov eksponentno narašča tudi število možnih postavitev atributov, ki jih je potrebno pregledati. Dodatno neprijetnost predstavlja tudi dejstvo, da je pri večini prikazov z večjim številom atributov skupen vpliv vseh atributov na posamezen primer približno nič, zaradi česar ležijo vse točke blizu središča kroga. Primer neinformativnega prikaza radviz, ki prikazuje 79 atributov, je na sliki 2.6.b.

Poleg metode radviz obstajajo še druge sorodne metode, ki na različne načine uporabljajo paradigmo vzmeti. Ena takih je metoda polyviz [50, 45]. Pri tej metodi krožnico nadomestimo z n-kotnikom, pri čemer n ustreza številu vizualiziranih atributov. Vsak

atribut je predstavljen s stranico mnogokotnika, krajišči stranice pa ustrezata minimalni ter maksimalni vrednosti atributa. Pri vizualizaciji vsakega primera je potrebno najprej na vsaki od stranic mnogokotnika poiskati točko, ki ustreza vrednosti atributa pri tem primeru. Ta točka nato predstavlja dimenzijsko sidro, ki kot pri metodi radviz privlači primer s silo, ki je sorazmerna vrednosti atributa. Vrednost vsakega atributa se torej upošteva dvakrat: najprej za določitev položaja dimenzijskega sidra na stranici mnogokotnika, nato pa še za določanje jakosti privlačne sile. Od položaja na stranici, ki ustreza dimenzijskemu sidru za posamezen primer, se običajno nariše tudi krajšo črto v smeri končnega položaja primera. S pomočjo teh črt dobimo vpogled v porazdelitev vrednosti pri posameznih atributih. Pri vizualizaciji večjega števila atributov postanejo stranice mnogokotnika krajše in postavitev točk postane podobna kot pri metodi radviz.

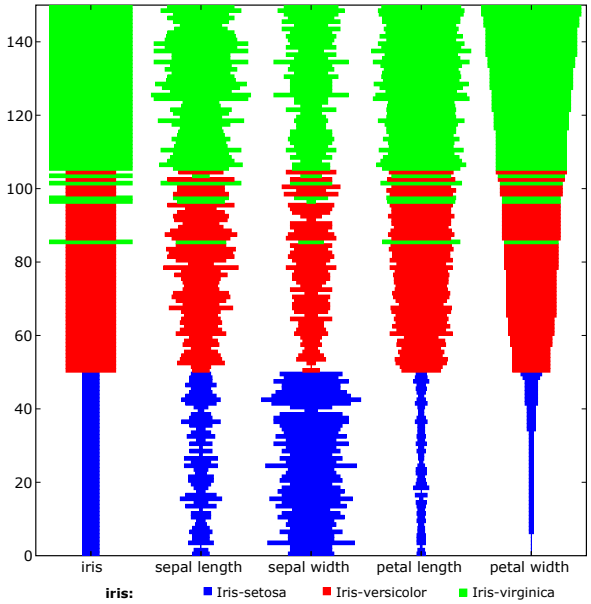
Podobna razširitev je tudi metoda gridviz [50], ki najprej sestavi kvadratno mrežo in nato v vsakega od robov mreže postavi dimenzijsko sidro. Avtorji metode trdijo, da lahko na ta način učinkoviteje vizualiziramo večje število atributov (tudi do $50 \times 50 = 2500$ atributov), ker so le-ti bolj enakomerno razporejeni po prostoru. Pri velikem številu atributov nastanejo težave s prikazom imen atributov, vprašljiva pa je tudi zmožnost interpretacije prikaza, saj so atributi razporejeni po celem prostoru, zaradi česar je teže sklepati o vplivu posameznih atributov na točke.

2.3.3 Pregledni diagram

Pregledni diagram (ang. *survey plot*) [77] je razširitev enostavnih stolpičnih grafov. V tem primeru so atributi predstavljeni z vertikalnimi osmi, vzdolž katerih so prikazani posamezni primeri. Vsak primer je prikazan s stolpci (en stolpec na vsak vizualiziran atribut), ki ležijo pravokotno na vsako od osi. Vsak stolpec je prikazan simetrično na obe strani osi, njegova širina pa je sorazmerna vrednosti primera pri danem atributu. V primeru klasificiranih podatkov je vrednost razreda običajno ponazorjena z barvo stolpca.

Vrstni red, v katerem vizualiziramo primere, je lahko poljuben. Če primere uredimo glede na vrednosti posameznega atributa, nam to omogoča opazovanje koreliranosti med tem in ostalimi atributi. V primeru, da primere uredimo glede na vrednosti razreda, lahko iz prikaza ocenimo uporabnost posameznih atributov za ločevanje med razredi. Pri vizualizaciji diskretnih atributov je primere mogoče urediti glede na več atributov hkrati, s čimer postane metoda uspešnejša za vizualno odkrivanje odločitvenih pravil za ločevanje med razredi.

Primer preglednega diagrama za podatke iris [80] je na sliki 2.7. Primeri so urejeni glede na naraščajočo vrednost atributa *petal width*, zaradi česar je lepo vidna korelacija tega atributa z atributom *petal length*. Očitna je tudi ločenost primerov iz različnih razredov, kar nakazuje, da sta atributa zelo primerna za ločevanje med razredi.

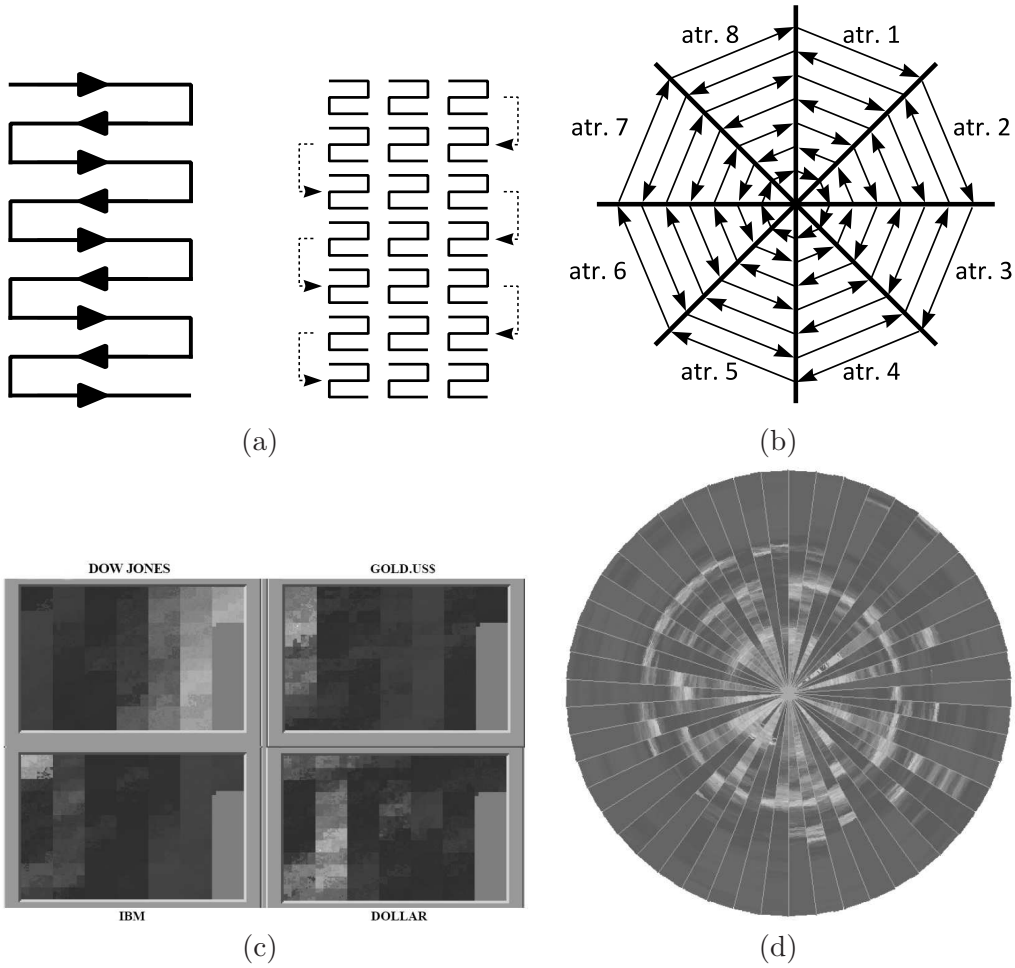


Slika 2.7: Pregledni diagram za podatke iris.

2.3.4 Metoda rekurzivnih vzorcev in krožnih segmentov

Obe metodi spadata v skupino pikselnih metod in sta zato namenjeni vizualizaciji večjih količin podatkov ter večjega števila atributov. Prvič sta bili predstavljeni znotraj sistema VisDB [61]. Metodi sta še posebej primerni za vizualizacijo podatkov, ki imajo neko naravno urejenost (naprimer za vizualizacijo časovnih vrst).

Pri metodi rekurzivnih vzorcev vizualiziramo vsak atribut v svojem podoknu. Znotraj posameznega podokna je vrednost vsakega primera pri tem atributu prikazana z enim pikslom, katerega barva ponazarja vrednost atributa. Metoda rekurzivnih vzorcev je dobila ime zaradi načina, kako so posamezni primeri znotraj vsakega podokna urejeni. Vrednosti primerov so prikazane v tako imenovanih osnovnih rekurzivnih elementih (ang. *recursive base elements*). Osnovni rekurzivni element je vzorec pravokotne oblike, višine h_1 in širine w_1 (vrednosti h_1 in w_1 določi uporabnik), ki prikazuje vrednosti za $h_1 \times w_1$ primerov. Primeri so znotraj elementa urejeni po vrsticah: v prvi vrstici je urejenost primerov od leve proti desni, v drugi od desne proti levi, v tretji spet od leve proti desni itd. Ker je primerov tipično več kot jih gre v en osnovni element, kreiramo več elementov in jih postavimo v obliki pravokotnika višine h_2 in širine w_2. Elemente uredimo v takem vrstnem redu, kot so urejeni posamezni primeri znotraj elementa (najprej od leve proti desni, nato od desne proti levi itd.). Na ta način lahko postopek rekurzivno ponavljamo. Osnovni rekurzivni vzorec in skupina vzorcev velikosti 7×3 sta prikazana na sliki 2.8.a. Slika 2.8.c prikazuje primer vizualizacije vrednosti borznih indeksov delnic za Dow Jones, IBM, zlato ter ameriški dolar za skoraj 7 zaporednih let. Podatki so prikazani tako, da vsakemu letu ustreza en stolpec, ki se nato deli še vertikalno na 12 delov, pri čemer vsak



Slika 2.8: Metoda rekurzivnih vzorcev in krožni segmenti. (a) Vrstni red primerov znotraj osnovnega rekurzivnega vzorca ter ena od možnih ureditev skupine vzorcev. (b) Vrstni red vizualizacije primerov pri metodi krožnih segmentov. Vizualizacija borznih indeksov z metodo rekurzivnih vzorcev (c) in krožnih segmentov (d).

del (osnovni rekurzivni vzorec) ustreza enemu mesecu. Izbrana barvna paleta preslika velike vrednosti atributov v svetle barve, majhne vrednosti pa v temne barve. Iz prikaza je lepo vidno, da je bila cena zlata zelo nizka v petem letu, da so delnice IBM-a po dveh mesecih hitro padle, da je imel ameriški dolar največjo vrednost v osmem mesecu drugega leta itd.

Za razliko od metode rekurzivnih vzorcev pri metodi krožnih segmentov vrednosti atributov niso prikazane v podoknih, ampak znotraj posameznih segmentov kroga. V primeru, da želimo vizualizirati n atributov, krog razdelimo na n segmentov, pri čemer vsakemu atributu pripada en segment. Vrednosti atributov pri posameznih primerih prikažemo tako, da se pomikamo od središča kroga navzven (kot je prikazano na sliki 2.8.b). Slika 2.8.d prikazuje podatke o petdesetih različnih borznih indeksih. Velike vrednosti

Eye	**Hair Color**				
Color	Black	Brown	Red	Blond	**Total**
Green	5	29	14	16	64
Hazel	15	54	14	10	93
Blue	20	84	17	94	215
Brown	68	119	26	7	220
Total	108	286	71	127	592

Tabela 2.1: Frekvenčna tabela za zbirko podatkov o barvi las in oči.

atributov so preslikane v svetle barve, majhne vrednosti pa v temne. Zaradi krožne postavitve je mogoče zlahka opaziti, da ima večina borznih indeksov zelo podobne trende obnašanja, medtem, ko se obnašanje nekaterih indeksov bistveno razlikuje.

2.3.5 Sieve in mozaični diagram

Pri analizi diskretnih ali diskretiziranih podatkov le-te pogosto pretvorimo v obliko frekvenčnih tabel. To so tabele, ki za različne vrednosti enega ali več atributov vsebujejo število primerov v zbirki podatkov, ki vsebuje to vrednost (ali kombinacijo vrednosti). V primeru, da tabela vsebuje podatke o vrednostih enega atributa, govorimo o enosmernih frekvenčnih tabelah, ko vsebuje podatke o kombinaciji vrednosti dveh ali več atributov, pa o dvo- ali večsmernih tabelah. Primer dvosmerne tabele so podatki v tabeli 2.1, ki jih je zbral Snee [99] in vsebuje podatke o barvi las in oči za 592 ljudi.

Za prikaz enosmernih frekvenčnih tabel obstajajo številne enostavne vizualizacijske metode – tipično se uporabljajo stolpični (ang. *bar chart*), točkovni (ang. *dot chart*) ali tortni diagram (ang. *pie chart*). Pri vizualizaciji dvo- ali večsmernih frekvenčnih tabel pa je pomembno, da konstruiramo take diagrame, ki nam prikažejo dejanske frekvence n_{ij} v posameznih celicah tabele v razmerju s frekvencami m_{ij}, ki bi jih pričakovali v primeru veljavnosti nekega predpostavljenega ničelnega modela (naprimer, da sta dva atributa neodvisna). Za primer dveh atributov lahko ob predpostavki neodvisnosti pričakovano frekvenco m_{ij} izračunamo kot produkt skupnega števila primerov v i-ti vrstici in j-tem stolpcu, deljeno s številom vseh primerov:

$$m_{ij} = \frac{n_{i+}n_{+j}}{n_{++}}.$$

Izračunana razlika med m_{ij} in n_{ij} nakazuje odstopanje od neodvisnosti med tema dvema vrednostima atributov. Poglejmo si naprimer primer iz tabele 2.1. Število ljudi s črnimi lasmi in rjavimi očmi (n_{41}) je 68. Po zgornji formuli lahko izračunamo, da je pričakovano število takih ljudi samo $220 \times 108/592 = 40.1$, iz česar lahko sklepamo, da sta vrednosti močno pozitivno korelirani.

Statistična mera, ki se pogosto uporablja za testiranje odvisnosti med atributi, je Pearsonov χ^2. Za posamezne kombinacije vrednosti atributov izračunamo standardizirane

Pearsonove residuale d_{ij} kot

$$d_{ij} = \frac{n_{ij} - m_{ij}}{\sqrt{m_{ij}}}.$$

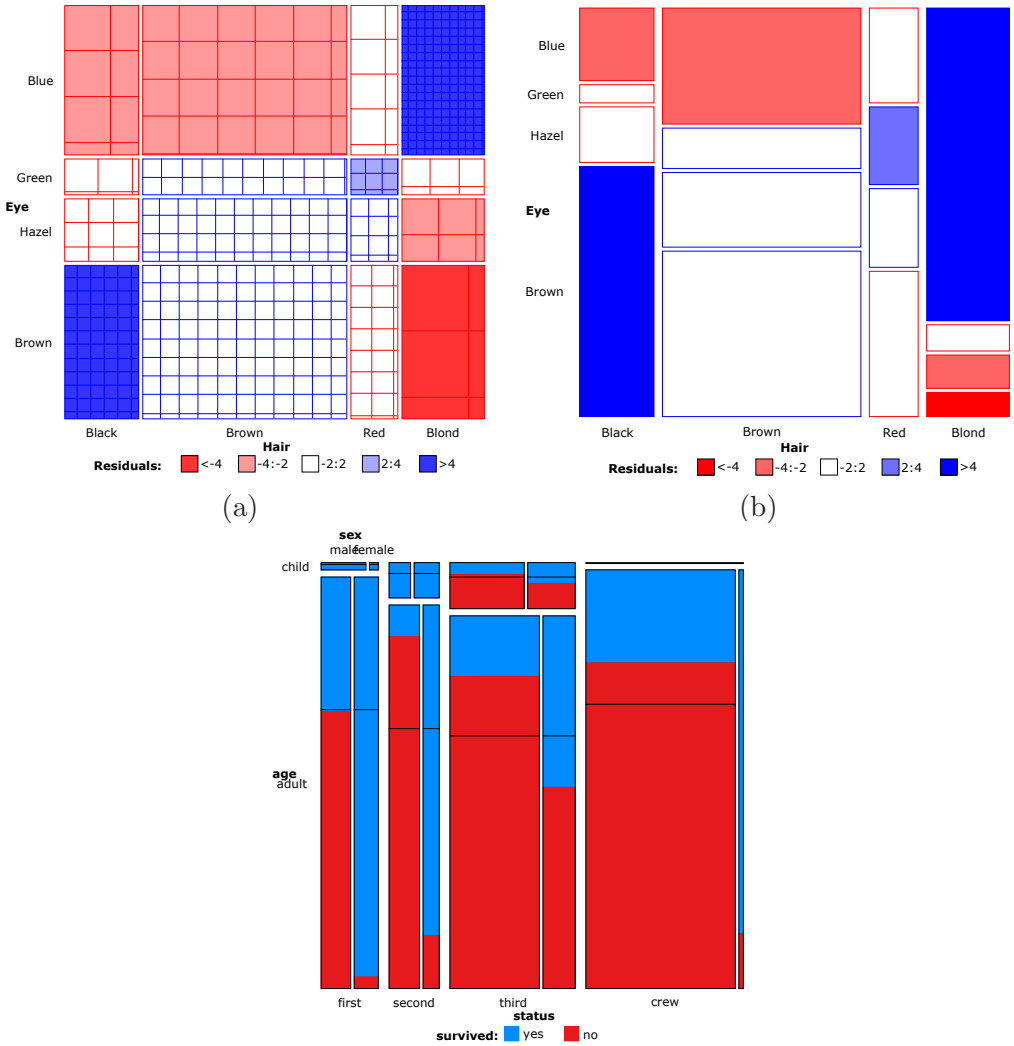
Pearsonov χ^2 je nato definiran kot vsota kvadratov posameznih residualov d_{ij}, oziroma $\chi^2 = \sum_i \sum_j d_{ij}^2$. Pearsonov χ^2 za podatke iz tabele 2.1 je 138.3. Glede na to, da podatki vsebujejo devet prostostnih stopenj, lahko v statističnih tabelah odčitamo, da je za tak χ^2 verjetnost, da sta atributa neodvisna, manjša od 0.001%.

Najbolj znani metodi za vizualizacijo dvo- ali večsmernih frekvenčnih tabel sta parketni diagram (ang. *sieve/parquet diagram*) [87, 88] ter mozaični diagram (ang. *mosaic plot*) [47, 48].

Parketni diagram sestavimo tako, da predpostavimo neodvisnost atributov. Podatke iz tabele predstavimo kot pravokotnike, katerih širina je sorazmerna skupnemu številu primerov v posameznem stolpcu (n_{+j}), višina pa skupnemu številu primerov v posamezni vrstici (n_{i+}) – ploščina pravokotnikov je tako sorazmerna pričakovanim frekvencam m_{ij}. Notranjost posameznih pravokotnikov uporabimo za prikaz odstopanja med pričakovanimi m_{ij} in dejanskimi frekvencami n_{ij}. Pobarvamo jih glede na predznak Pearsonovega residuala d_{ij} – pravokotnike, kjer je $d_{ij} > 0$ z modro, tiste, z $d_{ij} < 0$ pa z rdečo barvo. Za vsako barvo se običajno uporabljajo dva ali trije odtenki. Vrednost residuala prikažemo tudi z gostoto mreže v pravokotniku, kar omogoča lažje primerjanje z drugimi pravokotniki. Slika 2.9.a prikazuje parketni diagram za podatke iz tabele 2.1. Iz diagrama je lepo vidna močna pozitivna korelacija med črnimi lasmi in rjavimi očmi, med svetlimi lasmi in modrimi očmi ter močna negativna korelacija med svetlimi lasmi in rjavimi očmi.

Za razliko od parketnega diagrama je pri mozaičnem diagramu ploščina posameznih pravokotnikov sorazmerna dejanskim in ne pričakovanim frekvencam. Diagram generiramo tako, da začnemo s kvadratom, ki ga vertikalno razdelimo tako, da so širine dobljenih pravokotnikov sorazmerne s frekvencami posameznih vrednosti atributa, ki je prikazan na x osi. V naslednjem koraku vsakega od pravokotnikov ponovno razdelimo (tokrat horizontalno) tako, da višine pravokotnikov ustrezajo *pogojnim* frekvencam posameznih vrednosti atributa prikazanega na y osi. Primer mozaičnega diagrama je na sliki 2.9.b in predstavlja vizualizacijo podatkov iz tabele 2.1. Kot vidimo so širine posameznih pravokotnikov enake kot v parketnem diagramu, razlikujejo pa se njihove višine, saj predstavljajo pogojne frekvence. Tako je naprimer mogoče iz diagrama razbrati, da ima od črnolasih ljudi več kot polovica rjave oči, od svetlolasih ljudi pa več kot dve tretjini modre oči.

Mozaični diagrami so zaradi načina generiranja prikaza primerni tudi za vizualizacijo večjega števila atributov. Deljenje pravokotnikov izvajamo rekurzivno, pri čemer v vsakem koraku rekurzije trenutni pravokotnik razdelimo glede na pogojne frekvence vrednosti že uporabljenih atributov. Kadar nas zanimajo odvisnosti med vrednostmi atributov, lahko kot pri parketnem diagramu notranjost pravokotnikov pobarvamo glede na vrednosti Pearsonovih residualov. Friendly [39, 40] je predlagal tudi preurejanje stolpcev in



Slika 2.9: Sieve (a) in mozaični (b) diagram pri prikazu odvisnosti med barvo las in barvo oči. Mozaični diagram za prikaz verjetnosti preživetja na Titaniku glede na starost, spol ter status potnikov (c).

vrstic diagrama z namenom izboljšanja percepcije asociativnosti med kombinacijami vrednosti – vrstice in stolpce je smiselno urediti tako, da so večja odstopanja od neodvisnosti prikazana na robovih diagrama.

V primeru nadzorovanega učenja lahko namesto asociativnosti v notranjosti pravokotnikov prikažemo porazdelitev primerov po razredih. Primer takega diagrama je na sliki 2.9.c in prikazuje kako so spol, starost ter status potnikov na Titaniku vplivali na njihove možnosti za preživetje. V vsakem od pravokotnikov je prikazana tudi črta, ki ponazarja apriorno verjetnost preživetja (umrlo je 1490 od 2201 potnikov, zaradi česar je črta v vsakem pravokotniku na 67.7% višine) in omogoča primerjavo pogojne verjetnosti

preživetja z apriorno verjetnostjo. Iz diagrama je razvidno, da med posadko ni bilo otrok, da so preživeli vsi otroci v prvem in drugem razredu, da so imele ženske veliko prednost pri reševanju (še posebej če so bile v prvem ali drugem razredu) in podobno.

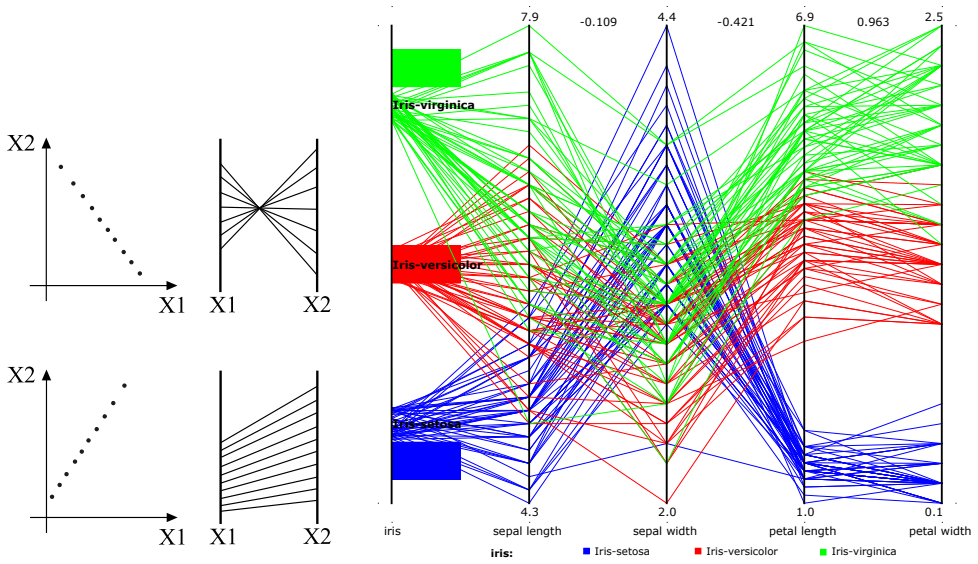
2.3.6 Paralelne koordinate

Paralelne koordinate (ang. *parallel coordinates*) so metoda, ki jo je leta 1985 razvil Inselberg [54]. Ugotovil je, da je težava pri običajnih kartezičnih koordinatah to, da sta osi postavljeni pravokotno ena na drugo, zaradi česar lahko hkrati vizualiziramo samo dva atributa. Predlagal je prostorsko veliko bolj učinkovito metodo, pri kateri so atributi predstavljeni z osmi, ki so postavljene vertikalno ter vzporedno ena ob drugi. Zgornji in spodnji rob vsake koordinatne osi običajno določata maksimalno ter minimalno vrednost atributa. Podatke nato vizualiziramo tako, da za vsak primer poiščemo na posameznih koordinatnih oseh točke, ki ustrezajo vrednostim atributov, ter sosednje točke med sabo povežemo. Posamezen primer je tako v paralelnih koordinatah prikazan kot lomljena črta, ki poteka med prvo in zadnjo koordinatno osjo. V primeru klasificiranih podatkov lahko primere pobarvamo in na ta na način prikažemo vrednost razreda.

Čeprav se metoda na eleganten način znebi omejitev kartezičnega sistema, vseeno ni primerna za vizualizacijo velikega števila atributov (> 20). Pri večjem številu atributov so namreč osi atributov postavljene blizu ena drugi, zaradi česar postane prostor med posameznimi osmi nasičen s črtami, kar onemogoča učinkovito percepcijo posameznih primerov. Težave pa metodi ne povzroča samo število atributov ampak tudi število primerov v zbirki podatkov. Z večanjem števila vizualiziranih primerov se namreč povečuje gostota črt med osmi, kar ponovno pripelje do problema percepcije primerov.

Metoda paralelnih koordinat je bila uporabljena že na širokem spektru večdimenzionalnih problemov, naprimer v robotiki, upravi, računski geometriji, reševanju konfliktov pri nadzoru zračnega letenja, procesni kontroli, kemiji, diferencialnih enačbah ter statistiki [55]. Primerna je za odkrivanje številnih struktur in lastnosti podatkov. Iz prikaza so lepo vidne porazdelitve primerov vzdolž posameznih atributov, kar v primeru klasificiranih podatkov omogoča hitro identifikacijo pomembnih atributov. Enostavno lahko odkrivamo osamelce, za katere je hitro opazen netipičen potek črt v primerjavi z ostalimi primeri. Lepo so vidne tudi gruče primerov, saj primeri iz iste gruče v prikazu ležijo blizu in potekajo med osmi pod podobnimi koti. Metoda je znana tudi po tem, da zelo jasno prikaže korelacije med atributi. V primeru pozitivne korelacije med dvema atributoma potekajo v paralelnih koordinatah črte med osmi približno vzporedno, v primeru negativne korelacije pa se črte sečejo v točki med osmi. Primer pozitivne in negativne korelacije je na sliki 2.10.a. Omeniti je potrebno, da je korelacija vidna samo v primeru, ko sta si osi koreliranih atributov prikazani kot sosednji; če so med njima še osi drugih atributov, korelacija ni vidna.

Na sliki 2.10.b je vizualizacija podatkov iris z metodo paralelnih koordinat. Numerične vrednosti na robovih osi atributov predstavljajo najmanjšo ter največjo vrednost tega atributa, vrednost med sosednjima zgornjima robovoma atributov pa predstavlja

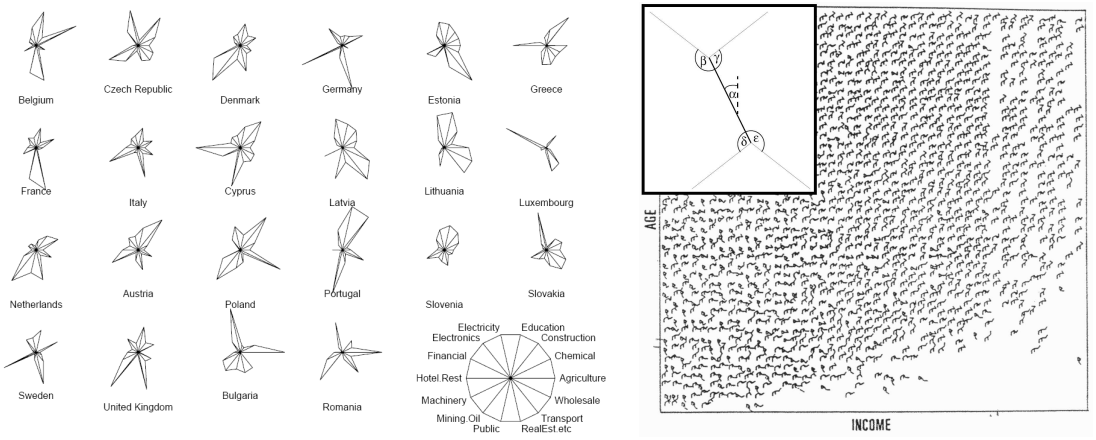


(a) (b)

Slika 2.10: (a) Izgled negativno in pozitivno koreliranih atributov v paralelnih koordinatah. (b) Vizualizacija podatkov iris z metodo paralelnih koordinat.

korelacijo med atributoma. Kot je bilo vidno že pri preglednem diagramu, je največja korelacija med atributoma *petal length* in *petal width*, med katerima potekajo črte skoraj vzporedno. Lepo je opazna tudi ločenost gruče primerov vrste *iris-setosa*, medtem, ko se primeri vrst *iris-virginica* in *iris-versicolor* rahlo prekrivajo.

Zaradi popularnosti metode je bilo razvitih veliko njenih razširitev in izboljšav. Mnoge od njih so različne interaktivne tehnike, ki omogočajo učinkovitejše in uspešnejše izvajanje vizualnega odkrivanja znanja iz podatkov. Mednje spadajo možnost interaktivnega preurejanja vrstnega reda atributov ter možnost približevanja in obračanja osi [97]. Pomembne so tudi različne tehnike filtriranja, ki omogočajo interaktivno izbiranje podmnožice primerov glede na različne kriterije. Poleg filtriranja glede na vrednosti posameznih atributov obstaja tudi kotno filtriranje (ang. *angular brushing*) [49], pri katerem lahko primere izbiramo glede na naklon črt med osmi. Graham [43] je za vizualizacijo primerov namesto lomljenih črt predlagal uporabo zlepkov, kar naj bi bistveno olajšalo sledenje posameznim primerom preko koordinatnih osi. Za boljše odkrivanje gruč je Wegman [109] predlagal kombiniranje metode paralelnih koordinat z metodo Grand Tour (opisano v razdelku 2.3.9). Osi v tem primeru niso posamezni atributi temveč utežene linearne kombinacije atributov. Uteži atributov se dinamično spreminjajo, zaradi česar je prikaz neke vrste animacija, ki naj bi po besedah avtorja olajšala detekcijo gruč.



(a) (b)

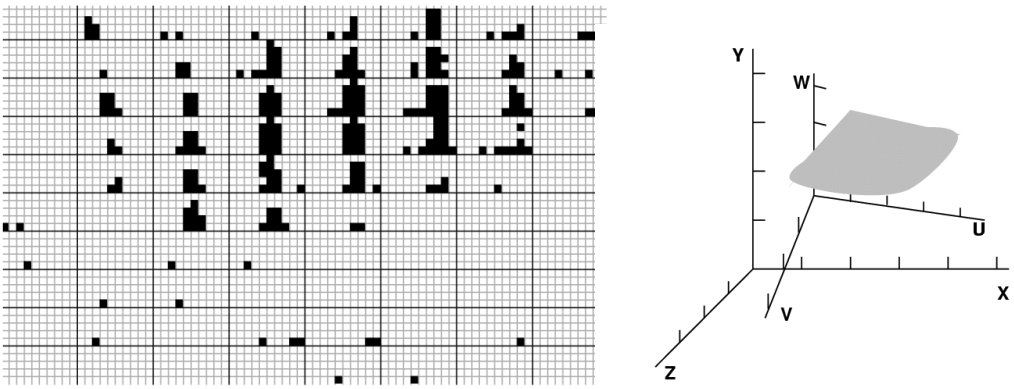
Slika 2.11: Primer zvezdnih (a) in paličnih figur (b).

2.3.7 Metode s figurami in ikonami

Pri tej družini metod se uporabljajo figure (ang. *glyphs*) ali ikone (ang. *icons*) kot osnovni gradniki, s katerimi prikažemo posamezne primere. Vrednosti posameznih atributov so prikazane z določanjem različnih oblik, kotov ali barv različnim delom teh gradnikov. Najbolj znan primer tovrstne metode (čeprav je njena dejanska uporabnost zelo vprašljiva) je metoda Chernoff faces [16], pri kateri so kot gradniki uporabljeni človeški obrazi, vrednosti posameznih atributov pa se preslikajo na različne značilke obraza (naklon in velikost oči, velikost nosu, oblika ust ipd.). Razlog za smiselnost takega načina prikaza podatkov naj bi bila v tem, da smo ljudje zelo učinkoviti pri razpoznavanju obrazov. Slabost te metode je med drugim ta, da imajo različne značilke obraza zelo različen vpliv na našo percepcijo, zaradi česar je zelo pomembno, kako preslikamo atribute v značilke obraza.

Ena od pogosto uporabljenih vrst figur so zvezdne figure (ang. *star glyphs*) [15, 108]. V zvezdnem diagramu so atributi predstavljeni s črtami, ki potekajo iz skupnega centra, pri čemer je črta vsakega atributa usmerjena pod drugačnim kotom. Vrednost atributa je predstavljena z dolžino črte – večja kot je vrednost, daljša je črta. Zunanje robove črt nato povežemo. Ker vsakemu primeru pripada ena zvezdna figura, je metoda primerna za vizualizacijo večjega števila atributov in majhnega števila primerov. S primerjanjem oblik posameznih figur lahko relativno uspešno odkrivamo podobnosti in razlike med posameznimi primeri. Primer zvezdnih figur je na sliki 2.11.a in prikazuje različne podatke o evropskih državah.

Ko imamo večje število primerov ter manjše število atributov, je bolj primerna uporaba paličnih figur (ang. *stick figures*) [84, 44]. V tem primeru je posamezna figura sestavljena iz različnih povezanih palic, pri katerih rotacija posameznih palic figure predstavlja vrednosti atributov. Na zgornjem levem robu slike 2.11.b je primer take figure, s katero lahko vizualiziramo vrednosti petih atributov. Posamezne figure, ki predstavljajo primere, so prikazane zelo blizu ena drugi, kar uporabniku omogoča lažjo percepcijo skupnega vzorca.



(a) (b)

Slika 2.12: Primer dimenzijskega zlaganja (a) ter metode svetovi znotraj svetov (b).

Uspešnost prikaza je zelo odvisna od izbranega načina preslikave posameznih atributov na posamezne palice. Palice so namreč povezane, zaradi česar je naklon neke palice lahko določen relativno glede na naklon drugih palic, na katere je le-ta pritrjena. Primer vizualizacije z uporabo paličnih figur je na sliki 2.11.b, kjer so prikazani različni atributi iz popisa prebivalstva. V prikazu je lepo vidna podobnost primerov v sredini prikaza, medtem ko se primeri na spodnjem robu bistveno razlikujejo od ostalih primerov.

2.3.8 Dimenzijsko zlaganje in svetovi znotraj svetov

Obe metodi sta primera hierarhičnih vizualizacijskih metod. Dimenzijsko zlaganje (ang. *dimensional stacking*) [74, 108] je metoda, primerna za vizualizacijo diskretnih ali diskretiziranih atributov z majhnim številom vrednosti. Najprej sestavimo 2D mrežo dimenzij $m \times n$, kjer sta m in n kardinalnosti atributov, ki bosta predstavljena na x in y osi. V vsako od celic mreže nato vstavimo nov koordinatni sistem z novo mrežo, katere dimenzije ustrezajo kardinalnosti drugih dveh atributov. Postopek rekurzivno ponavljamo, dokler ne prikažemo vseh atributov. Primere nato prikažemo tako, da v mrežah na najglobljem nivoju obarvamo tiste celice, ki ustrezajo vrednostim atributov pri posameznih primerih. Pomemben vpliv na percepcijo ima vrstni red vizualiziranih atributov. Priporoča se, da se pomembnejši atributi vizualizirajo na zunanjih oseh. Na sliki 2.12.a so z uporabo te metode prikazani podatki o črpanju nafte. Na zunanjih oseh sta prikazana zemljepisna dolžina in širina, na notranjih oseh pa kvaliteta in globina nafte.

Če paradigmo dimenzijskega zlaganja razširimo na tri dimenzije, dobimo metodo, ki sta jo Feiner in Besherrs imenovala svetovi znotraj svetov (ang. *worlds within worlds*) [30, 29]. Namenjena je prikazu n-dimenzionalnih matematičnih funkcij. V zunanjem 3D koordinatnem sistemu uporabnik izbere točko, s čimer določi vrednosti prvih treh atributov. V tej točki se nato pojavi nov koordinatni sistem z naslednjimi tremi atributi. Postopek ponavljamo, dokler ne določimo vrednosti $n - 3$ atributom. V zadnjem koordinatnem sistemu lahko nato prikažemo 3D obliko funkcije. Težava pri tem načinu prikaza je, da

mora uporabnik vnaprej vedeti kaj išče, saj je vizualizacija vidna šele pri zadnji postavitvi koordinatnih osi. Primer prikaza šestparametrske funkcije je na sliki 2.12.b.

2.3.9 Grand Tour

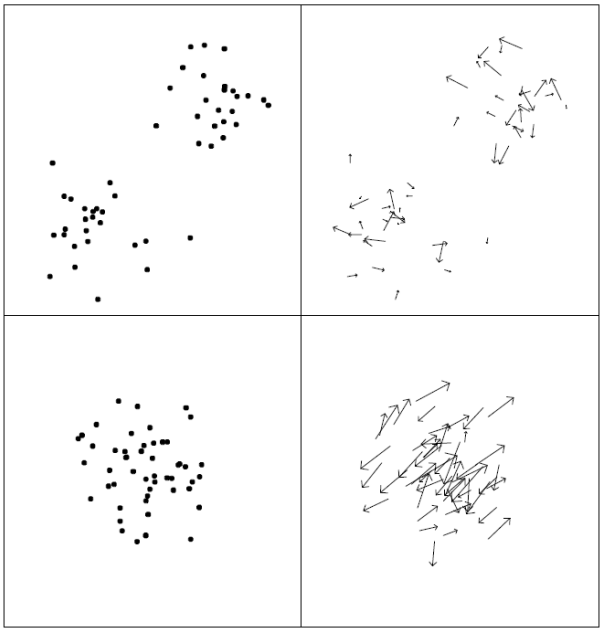
Grand Tour [2] je primer vizualizacijske metode, ki spada v skupino dinamičnih projekcijskih metod. Prikaz je v tem primeru k-dimenzionalna projekcija, kjer je vrednost parametra k lahko 1, 2 ali 3. Najpogosteje se uporablja $k = 2$, s čimer dobimo običajno 2D projekcijo, kjer vsaka od koordinatnih osi predstavlja uteženo linearno kombinacijo originalnih atributov. Naj bosta w_x in w_y vektorja, ki vsebujeta vrednosti uteži za koordinatni osi x in y. Položaj posameznega primera $a_i = (a_{i,1}, a_{i,2}, ..., a_{i,n})$ v projekciji ima tedaj koordinate:

$$x_i = w_x a_i$$
$$y_i = w_y a_i.$$

Za razliko od ostalih opisanih metod prikaz podatkov v tem primeru ni statičen, ampak se s časom spreminja. Spreminjanje prikaza dosežemo s počasnim spreminjanjem uteži posameznih atributov na koordinatnih oseh, kar da analitiku vtis zvezne rotacije prostora. Informacije tako ne dobimo samo iz položaja posameznih točk v projekcijah, ampak tudi iz smeri in hitrosti njihovega spreminjanja. Čeprav je percepcija spremembe položaja slabša od percepcije samega položaja točk, nam skupna informacija, ki jo tako dobimo, da veliko bolj celosten pogled na geometrijo "oblaka" točk v visokodimenzionalnem prostoru. Primer uporabnosti obeh virov informacije lepo demonstrira slika 2.13. Slika prikazuje dve projekciji 4D podatkov, ki vsebujejo dve lepo ločeni gruči točk. Leva stolpca prikazujeta položaje točk v projekcijah, desna stolpca pa smer spreminjanja položaja točk. V zgornji projekciji je ločenost gruč lepo vidna iz položaja točk, ne pa tudi iz smeri njihovega spreminjanja. V spodnji projekciji se gruči popolnoma prekrivata, opazna pa je regularnost v smereh spreminjanja položaja točk; ena skupina puščic kaže v zgornji desni rob, druga pa v spodnji levi rob, kar indicira na prisotnost dveh skupin.

Uspešnost dinamičnega načina prikaza podatkov je zelo odvisna od tega, na kakšen način se spreminjajo uteži posameznih atributov. Eden od bistvenih pogojev je ta, da so spremembe v vrednostih uteži med dvema sosednjima projekcijama dovolj majhne, da ustvarijo vtis zvezne rotacije prostora. Pomembno je tudi, da je spreminjanje uteži regularno in preiskovalno. Za regularno spreminjanje vrednosti velja, da smer spreminjanja ne oscilira, ampak ostaja ista za celo serijo projekcij. To povzroči, da se tudi primeri v prikazu premikajo regularno, s čimer dobimo informacijo o vplivu posameznih atributov na položaj primerov. Za dosego celostnega pregleda nad podatki je pomembno tudi, da nas algoritem vodi po celotnem prostoru možnih projekcij, zaradi česar se morajo uteži atributov spreminjati neodvisno ter znotraj celotnega spektra vrednosti.

Na voljo so različni preiskovalni algoritmi, ki ustrezajo omenjenim zahtevam. Najpomembnejša skupina algoritmov so podatkovnovodeni algoritmi, pri katerih prostora



Slika 2.13: Dve vrsti predstavljata dve projekciji 4D podatkov. Levi stolpec prikazuje položaje primerov, desni pa smer spreminjanja položaja. V zgornji projekciji lahko opazimo dve gruči z opazovanjem položaja primerov, v spodnji projekciji pa z opazovanjem smeri spreminjanja.

projekcij ne preiskujemo naključno, ampak glede na to, kakšno strukturo v podatkih iščemo (gruče, osamelci, odstopanje od normalnosti porazdelitve primerov itd). Uteži atributov v tem primeru spreminjamo tako, da je iskana struktura iz prikaza v prikaz vedno bolj vidna. Področju, ki se ukvarja z določitvijo različnih mer za iskanje zanimivih projekcij podatkov, pravimo projekcijsko iskanje in je podrobneje opisano v razdelku 2.4.5.

2.4 Postopki za iskanje zanimivih linearnih projekcij

Cilj vizualnega odkrivanja znanja iz podatkov je odkrivanje različnih struktur in zakonitosti v podatkih s pomočjo vizualizacije. Ta cilj pa ni enostavno dosegljiv. Četudi je neka struktura v podatkih prisotna, je namreč njena opaznost v prikazu (ne glede na izbrano vizualizacijsko metodo) zelo odvisna od tega, katere atribute vizualiziramo in na kakšen način. Če želimo, naprimer, v razsevnem diagramu videti primere ločene v različne gruče, je to pogosto možno samo takrat, ko vizualiziramo "pravo" kombinacijo atributov. Podobno velja za splošne linearne projekcije, kjer postanejo različne strukture vidne zgolj v izjemnih primerih, ko imajo uteži atributov točno določene vrednosti.

Za nekatere od pogosto iskanih struktur so bile razvite metode, ki omogočajo identifikacijo takih linearnih projekcij, ki to strukturo (v kolikor je v podatkih prisotna) kar se

da dobro prikažejo. Za nekatere od teh metod obstajajo analitične rešitve, za druge pa je potrebno uporabiti različne optimizacijske algoritme. Ti postopki so uporabni tudi kot tehnike za zmanjševanje dimenzionalnosti podatkov (ang. *dimensionality reduction techniques*), saj podatke preslikajo v nižje dimenzionalni prostor ter hkrati ohranijo zanimivo informacijo.

Glede na to, da je cilj postopkov za iskanje zanimivih linearnih projekcij enak cilju te disertacije, bomo v nadaljevanju na kratko predstavili pomembnejše med njimi. Najbolj znana med njimi sta analiza glavnih komponent ter diskriminantna analiza, ki poiščeta prikaze s čim večjo varianco ter prikaze s čim boljšo ločenostjo razredov. Opisali bomo tudi projekcijsko iskanje, ki najde prikaze s čim bolj *ne*normalno porazdelitvijo primerov, ter FreeViz, ki pri iskanju zanimivih projekcij uporablja fizikalni princip privlačnosti in odbojnosti med posameznimi primeri.

2.4.1 Analiza glavnih komponent

Analiza glavnih komponent (ang. *principal component analysis, PCA*) [51, 42] je statistična metoda, katere cilj je zmanjšanje števila dimenzij v podatkih z odstranjevanjem odvisnosti med atributi. Atributi v neki zbirki podatkov običajno niso neodvisni, ampak so medsebojno korelirani. Del informacije, ki jo doprinese posamezen atribut, je tako redundanten, saj je vsebovan že v drugih atributih. Da bi se izognili podvajanju informacij, lahko z analizo glavnih komponent najdemo tako linearno transformacijo (projekcijo, rotacijo), ki z manjšim številom atributov zajame čim več vsebovane informacije.

Informacija je v podatkih vsebovana v obliki variance; atribut, ki ima pri vseh primerih enako vrednost (varianca je v tem primeru enaka nič), je neinformativen in neuporaben. Cilj analize glavnih komponent je zato poiskati manjše število takih neodvisnih vektorjev, ki bodo pojasnili večino variance v podatkih.

Postopek iskanja takih vektorjev je naslednji. Vzemimo, da imamo na voljo n-dimenzionalno zbirko podatkov. Vsak od N primerov je vektor oblike:

$$x_i = [a_{i,1},\ a_{i,2},\ ...,\ a_{i,n}]^T, \qquad i = 1...N.$$

Podatke najprej normaliziramo tako, da izračunamo srednjo vrednost μ_i za vsakega od atributov in jo odštejemo od vsakega primera. Normalizirane primere predstavimo v obliki matrike $\mathbf{X}$:

$$\begin{aligned} x'_i &= [\ a_{i,1} - \mu_1,\ a_{i,2} - \mu_2,\ ...,\ a_{i,n} - \mu_n\ ]^T \\ \mathbf{X} &= [x'_1,\ x'_2, ...,\ x'_N]. \end{aligned}$$

Za normalizirane podatke nato izračunamo kovariančno matriko Σ, pri čemer element $\sigma^2_{m,n}$ te matrike predstavlja kovarianco med m-tim in n-tim atributom:

$$\begin{aligned}\mathbf{\Sigma} &= \frac{1}{N}\mathbf{X}^T\mathbf{X} \\ \sigma^2_{m,n} &= \frac{1}{N}\sum_{i=1}^{N}(v_{i,m}-\mu_m)(v_{i,n}-\mu_n).\end{aligned}$$

Z reševanjem sistema $\mathbf{\Sigma}v = \lambda v$ lahko nato kovariančni matriki $\mathbf{\Sigma}$ poiščemo n pripadajočih lastnih vektorjev v_i in lastnih vrednosti λ_i. Vektorji so medsebojno ortogonalni ($v_i v_j = 0$) ter normalizirani ($\|v_i\|_2 = 1$). Uredimo jih lahko po pomembnosti glede na lastne vrednosti, tako da velja $\lambda_i \geq \lambda_j, 1 \leq i \leq j \leq n$. Lastni vektor v_1, ki pripada največji lastni vrednosti λ_1, tako predstavlja smer v originalnem prostoru, vzdolž katere je varianca največja. Vsak naslednji lastni vektor je pravokoten na vse prejšnje vektorje in kaže v smeri največje *preostale* variance. Delež celotne variance, ki jo predstavlja lastni vektor v_i, je $\lambda_i / \sum_{j=1}^{n}\lambda_j$. Podatke lahko učinkoviteje predstavimo tako, da obdržimo samo prvih k lastnih vektorjev ($k \ll n$) ter tako ohranimo $\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{n}\lambda_i}$ celotne variance. Novo k-dimenzionalno zbirko podatkov $\mathbf{Y}$ dobimo tako, da originalne podatke pomnožimo z matriko izbranih lastnih vektorjev: $\mathbf{Y} = \mathbf{XV}; \mathbf{V} = [v_1, v_2, ..., v_k]$.

Čeprav je analiza glavnih komponent uporabna na širokem spektru domen (kompresija podatkov, regresija, analiza slik, razpoznavanje vzorcev), ima metoda vendarle nekaj omejitev. Ena njenih večjih težav je velika občutljivost na osamelce. Ker metoda išče podprostor, ki bo ohranil čim več variance, lahko prisotnost osamelcev bistveno vpliva na smer dobljenih lastnih vektorjev. Metoda prav tako ni primerna za analizo klasificiranih podatkov, saj je smer maksimalne variance pogosto popolnoma nepovezana s smerjo, ki bi omogočala dobro razlikovanje med različnimi razredi. Ena od razširitev osnovne metode, s katero lahko odpravimo omenjene omejitve, je utežena analiza glavnih komponent.

2.4.2 Utežena analiza glavnih komponent

Utežena analiza glavnih komponent je ena novejših izboljšav analize glavnih komponent, ki sta jo predlagala Koren in Karmel [68, 69]. Avtorja sta pokazala, da analiza glavnih komponent poišče tak podprostor dimenzije k, v katerem je vsota kvadratov razdalj med vsemi pari primerov maksimalna. Če razdaljo med i-tim in j-tim primerom v projeciranem prostoru dimenzije k označimo kot d_{ij}, potem lahko trdimo, da analiza glavnih komponent poišče take vektorje $v_1, ..., v_k$, za katere velja:

$$\begin{aligned}&\max_{v_1,...,v_k} \sum_{i<j} d_{ij}^2 \\ &\text{pri pogoju: } v_i v_j = \delta_{ij}, \quad i,j = 1,...,k\end{aligned} \tag{2.1}$$

kjer je δ_{ij} Kronekerjev delta, ki ima vrednost 1 pri $i = j$ in 0 sicer. Ker je cilj maksimizacija *kvadrata* razdalj med primeri, posveti metoda veliko pozornosti ohranjanju

velikih razdalj, pogosto na račun slabšega ohranjanja manjših razdalj. Da bi zmanjšali občutljivost metode na prisotnost osamelcev, bi bilo zato koristno, če bi lahko razdalje med primeri utežili in poiskali rešitev spremenjenega maksimizacijskega problema:

$$\begin{aligned} &\max_{v_1,...,v_k} \quad \sum_{i<j} w_{ij} \cdot d_{ij}^2 \\ &\text{pri pogoju:} \quad v_i v_j = \delta_{ij}, \quad i,j = 1,...,k \end{aligned} \tag{2.2}$$

Utež w_{ij} bi v tem primeru določala, kako pomembno je, da ležita primera i in j v novem prostoru kar se da narazen. Vprašanje je le, kako algoritem iskanja glavnih komponent prilagoditi, da bo upošteval vrednosti uteži.

Avtorja sta rešitev našla v uporabi Laplaceove matrike. Laplaceova matrika $\mathbf{L}$ je vsaka simetrična pozitivno semidefinitna matrika dimenzij $N \times N$, za katero velja, da je vsota vsakega stolpca in vsake vrstice enaka 0. Posebna vrsta take matrike je *enotska* Laplaceova matrika $\mathbf{L}^u$, katere elementi imajo vrednosti $L^u_{ij} = \delta_{ij} \cdot N - 1$. Zanjo velja, da je matrika $\mathbf{X}^T\mathbf{L}^u\mathbf{X}$ enaka kovariančni matriki $\mathbf{\Sigma}$ do pozitivnega multiplikativnega faktorja, $\mathbf{X}^T\mathbf{L}^u\mathbf{X} = N^2\mathbf{\Sigma}$. Ker množenje matrike s pozitivno konstanto ne spremeni njenih lastnih vektorjev, je iskanje glavnih komponent s pomočjo kovariančne matrike $\mathbf{\Sigma}$ torej ekvivalentno iskanju s pomočjo matrike $\mathbf{X}^T\mathbf{L}^u\mathbf{X}$.

Bolj impresivno in uporabno pa je dejstvo, da lahko definiramo tako Laplaceovo matriko $\mathbf{L}$, v kateri lastni vektorji matrike $\mathbf{X}^T\mathbf{L}\mathbf{X}$ ne predstavljajo rešitve problema iz enačbe (2.1), ampak uteženo različico iz enačbe (2.2). Matrika $\mathbf{L}$ mora v tem primeru biti definirana kot

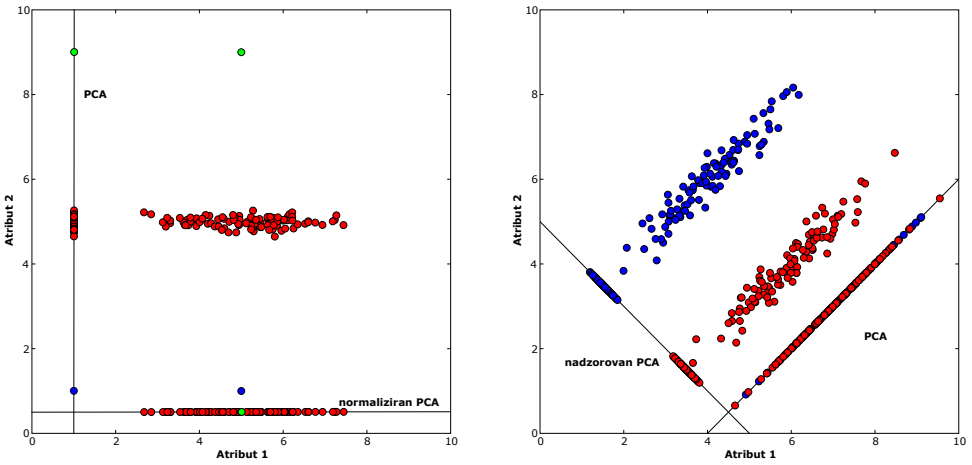
$$L_{ij} = \begin{cases} \sum_{j=1}^{N} w_{ij} & i = j \\ -w_{ij} & i \neq j. \end{cases} \tag{2.3}$$

Kot rečeno, lahko uteži w_{ij} določimo poljubno, glede na naš cilj. Če želimo naprimer zmanjšati vpliv osamelcev, potem avtorja predlagata, da uteži definiramo kot

$$w_{ij} = \frac{1}{D_{ij}},$$

pri čemer je D_{ij} razdalja med i-tim in j-tim primerom v originalnem, n-dimenzionalnem prostoru. Tej različici pravita *normalizirana* analiza glavnih komponenent. Primerjava z osnovno analizo glavnih komponent je na sliki 2.14.a. Kot je razvidno iz slike, se prvi glavni komponenti obeh metod popolnoma razlikujeta; osnovna analiza popolnoma zgreši smer zaradi dveh osamelcev, medtem, ko njena normalizirana različica kaže v smeri vzdolž katere je varianca večine primerov največja.

Ideja uteževanja razdalj med posameznimi primeri je tako splošna, da lahko z njo metodo priredimo celo za probleme nadzorovanega učenja. V tem primeru želimo videti



(a) (b)

Slika 2.14: Primerjava analize osnovnih komponent (PCA) z normalizirano (a) ter nadzorovano analizo osnovnih komponent. Na slikah je za vsako metodo prikazana zgolj prva (najpomembnejša) komponenta.

primere istega razreda skupaj in lepo ločene od primerov, ki pripadajo drugim razredom. To lahko dosežemo tako, da v Laplaceovi matriki $\mathbf{L}$ določimo vrednosti utežem w_{ij} kot

$$w_{ij} = \begin{cases} t \cdot \frac{1}{D_{ij}} & i \text{ in } j \text{ pripadata istemu razredu} \\ \frac{1}{D_{ij}} & \text{sicer.} \end{cases}$$

Spremenljivka t, $0 \leq t \leq 1$, določa, kako pomembno je, da dobljene glavne komponente prikažejo tudi varianco med primeri z isto vrednostjo razreda. Tipično se uporablja vrednost $t = 0$, kar pomeni, da cilj ni razpršiti primere z istim razredom, ampak samo čim bolje ločiti posamezne razrede med sabo. Primer te *nadzorovane* različice metode je na sliki 2.14.b. Glavni vektor, ki ga najde analiza osnovnih komponent, je popolnoma neuporaben za ločevanje med razredoma, medtem ko uspe spremenjeni postopek odlično ločiti oba razreda med sabo.

2.4.3 Linearna diskriminantna analiza

Analiza glavnih komponent poišče smeri v prostoru, vzdolž katerih je varianca največja in pri tem ne upošteva potencialne informacije o tem, kateremu razredu posamezen primer pripada. V nasprotju s tem linearna diskriminantna analiza (ang. *linear discriminant analysis, LDA*) išče smeri, ki omogočajo kar se da dobro ločevanje med razredi. Metodo je za dvorazredne probleme razvil Fisher [31] (Fisherjeva diskriminantna analiza), za večrazredne probleme pa sta jo kasneje razširila Rao [86] in Bryan [11] (multipla diskriminantna analiza). Metoda temelji na predpostavki, da so primeri iz vsakega razreda porazdeljeni po normalni porazdelitvi in da imajo enako kovariančno matriko.

Naj bo $\mathbf{X}$ zbirka podatkov, ki vsebuje N primerov, pri čemer vsak primer pripada enemu od c razredov. Za izračun diskriminant (vektorjev v originalnem prostoru, ki kar se da dobro ločijo med razredi) metoda uporabi dve matriki: $\mathbf{S}_W$ in $\mathbf{S}_B$. Matrika $\mathbf{S}_W$ je vsota kovariančnih matrik po posameznih razredih in vsebuje informacijo o tem, kako so primeri razpršeni znotraj posameznih razredov:

$$\mathbf{S}_W = \sum_{i=1}^{c} N_i \cdot \mathbf{\Sigma}_i = \sum_{i=1}^{c} \sum_{x \in \mathcal{D}_i} (x - \mu_i)(x - \mu_i)^T,$$

pri čemer je $\mathcal{D}_i$ množica primerov, ki pripadajo i-temu razredu, N_i pa moč te množice. Srednja vrednost μ_i primerov znotraj razreda i je

$$\mu_i = \frac{1}{N_i} \sum_{x \in \mathcal{D}_i} x.$$

Na drugi strani pa matrika $\mathbf{S}_B$ vsebuje informacijo o tem, kako dobro so razpršene srednje vrednosti posameznih razredov μ_i in je definirana kot

$$\mathbf{S}_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T,$$

pri čemer je μ srednja vrednost vseh primerov, $\mu = 1/N \sum_{x \in \mathbf{X}} x$. Projekcije, ki bi jih želeli najti, imajo majhno razpršenost primerov znotraj posameznih razredov (koherentnost skupin primerov) in hkrati veliko razpršenost srednjih vrednosti μ_i (dobra ločenost skupin). Formalno lahko ta kriterij zapišemo kot

$$J(\mathbf{V}) = \frac{\mathbf{V}^T \mathbf{S}_B \mathbf{V}}{\mathbf{V}^T \mathbf{S}_W \mathbf{V}}. \tag{2.4}$$

Čeprav je iskanje matrike $\mathbf{V}$, ki maksimizira kriterij $J(\cdot)$, zapleteno, se na srečo izkaže, da je rešitev relativno enostavna. Stolpci optimalne matrike $\mathbf{V}$ so lastni vektorji v_i, ki ustrezajo lastnim vrednostim enačbe

$$\mathbf{S}_B v_i = \lambda_i \mathbf{S}_W v_i.$$

Problem je torej zelo podoben problemu, ki smo ga reševali pri analizi glavnih komponent – potrebno je samo najti lastne vektorje in lastne vrednosti matrike $\mathbf{S}_W^{-1}\mathbf{S}_B$. Ker je rang matrike $\mathbf{S}_B$ največ $c - 1$, bo največ $c - 1$ lastnih vrednosti v_i neničelnih. Za c razredni problem lahko tako najdemo maksimalno $c-1$ diskriminant, ki ustrezajo lastnim vektorjem z neničelnimi lastnimi vrednostmi. Kot pri analizi glavnih komponent lahko te diskriminante uredimo po pomembnosti glede na pripadajoče lastne vrednosti, projekcijo z najboljšo ločenostjo razredov pa dobimo, če vizualiziramo diskriminante z največjimi lastnimi vrednostmi.

2.4.4 Utežena linearna diskriminantna analiza

Linearna diskriminantna analiza je v osnovi namenjena klasifikaciji podatkov, zaradi česar ima, gledano s stališča vizualizacije, dve pomankljivosti. Kriterij maksimizacije razdalje med gručami povzroči, da je metoda občutljiva na pristotnost osamelcev, zaradi česar raje prikaže malo močno ločenih gruč, kot pa več slabše ločenih gruč primerov. Druga, še večja težava pa je, da z minimiziranjem variance znotraj gruč metoda popolnoma ignorira njihovo obliko in velikost. Ne glede na to, ali je gruča homogena ali heterogena, ali je podolgovata ali sferična, LDA vedno poskuša prikazati gručo kot majhno sfero. To je mogoče primerno, če metodo uporabljamo za klasifikacijo, onemogoča pa nam, da bi uspešno ocenili vizualne lastnosti gruče.

Zaradi podobnosti med analizo glavnih komponent in linearno diskriminantno analizo je mogoče tudi pri tej metodi uporabiti isti princip kot pri uteženi analizi glavnih komponent. Pokazati je mogoče, da je kriterij, ki ga optimizira linearna diskriminantna analiza (izraz (2.4)), ekvivalenten kriteriju

$$J(\mathbf{V}) = \frac{\mathbf{V}^T\mathbf{X}^T\mathbf{L}^B\mathbf{X}\mathbf{V}}{\mathbf{V}^T\mathbf{X}^T\mathbf{L}^u\mathbf{X}\mathbf{V}},$$

če je matrika $\mathbf{L}^u$ enotska Laplaceova matrika, elementi matrike $\mathbf{L}^B$ pa so definirani kot

$$L_{ij}^B = \begin{cases} -1 + \frac{N}{N_g} & \text{primera } i \text{ in } j \text{ pripadata razredu } g \\ -1 & \text{primera } i \text{ in } j \text{ pripadata različnima razredoma.} \end{cases}$$

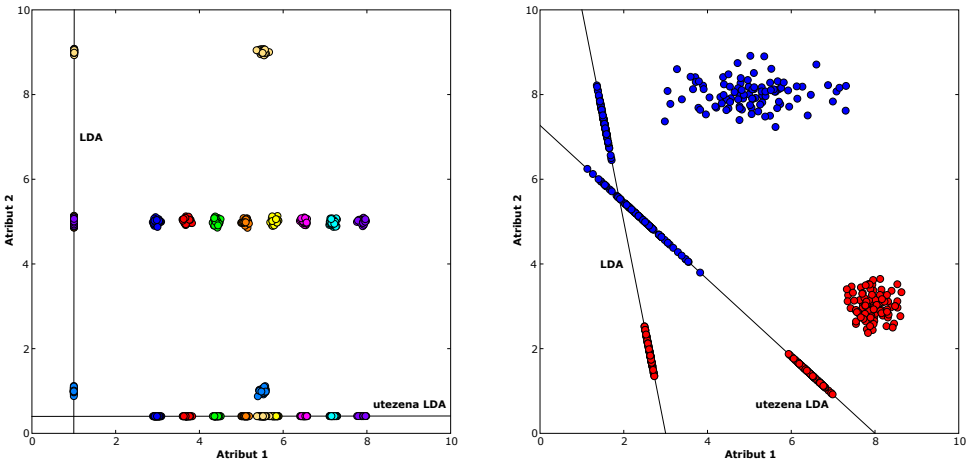
Za odpravo pomankljivosti metode lahko tudi pri linearni diskriminantni analizi definiramo ustrezne Laplaceove matrike. Avtorja Koren in Karmel [68, 69] sta za zmanjšanje dominantnosti velikih razdalj predlagala definiranje matrik $\mathbf{L}^B$ in $\mathbf{L}^W$ (nadomestek matrike $\mathbf{L}^u$) tako, da so uteži w_{ij} iz izraza (2.3) definirane kot

$$w_{ij}^B = \begin{cases} \frac{1}{D_{ij}} & \text{primera } i \text{ in } j \text{ pripadata istemu razredu} \\ 0 & \text{primera } i \text{ in } j \text{ pripadata različnima razredoma} \end{cases}$$

$$w_{ij}^W = \begin{cases} 0 & \text{primera } i \text{ in } j \text{ pripadata istemu razredu} \\ \frac{1}{D_{ij}} & \text{primera } i \text{ in } j \text{ pripadata različnima razredoma} \end{cases}$$

pri čemer je D_{ij} razdalja med i-tim in j-tim primerom v originalnem prostoru.

Dva primera, ki prikazujeta razliko med originalno in uteženo različico metode, sta na sliki 2.15. Levi primer (slika 2.15.a) vsebuje 10 ločenih gruč, od katerih vsaka vsebuje 50 primerov. Položaj dveh gruč je bistveno drugačen od ostalih. Kot je vidno na sliki, bi enodimenzionalna projekcija, dobljena z običajno LDA, poudarila oddaljeni gruči ter združila vseh ostalih osem gruč (večino primerov) v eno. Modificirana različica metode s tem ne bi imela težav in bi uspešno prikazala ločenost večine gruč. Primer na sliki 2.15.b pa vsebuje samo dve gruči primerov, pri čemer ima ena gruča krožno, druga pa podolgovato obliko. Osnovna različica metode bi poiskala projekcijo, kjer bi bili obe gruči čim



(a) (b)

Slika 2.15: Dva primera osnovne linearne diskriminantne analize (LDA) v primerjavi z uteženo linearno diskriminantno analizo. Na slikah je za vsako metodo prikazana zgolj prva (najpomembnejša) komponenta.

bolj enako razpršeni, medtem ko bi utežena različica prikazala razliko v razpršenosti ter hkrati ohranila ločenost skupin.

2.4.5 Projekcijsko iskanje

Projekcijsko iskanje (ang. *projection pursuit, PP*) je posebna vrsta metode Grand Tour. Ker je prostor možnih projekcij običajno zelo velik, je iskanje zanimivih struktur v podatkih z neusmerjenim preiskovanjem prostora zelo neučinkovito. Da bi bilo iskanje uspešnejše, je potrebno neusmerjeno preiskovanje nadomestiti z ustreznim podatkovnovodenim preiskovanjem, s čimer dosežemo, da nas postopek z vsako naslednjo projekcijo vodi v tisti del prostora, ki vsebuje vedno več iskane strukture.

Idejo samega iskanja zanimivih projekcij sta prva predstavila Kruskal [70] in Switzer [104], prvo uspešno implementacijo pa sta razvila Friedman in Tukey [38], ki sta tudi predlagala ime projekcijsko iskanje. Uporabljeno matematično notacijo, ki je postala osnova za nadaljnjo statistično analizo, je poenotil Huber [52]. Številni avtorji so v naslednjih letih razvili mnoge razširitve, s katerimi je mogoče projekcijsko iskanje uporabiti tudi v regresiji [35, 36], klasifikaciji [85, 75] ter pri ocenjevanju gostote primerov (ang. *density estimation*) [36, 37, 52].

Postopek projekcijskega iskanja sestavljata dva ključna elementa: projekcijski indeks (ang. *PP index*) in algoritem za iskanje projekcij. Projekcijski indeks $I(\alpha)$ je mera, ki ocenjuje zanimivost projekcije α pri podatkih $\mathbf{X}$ (mera je implicitno odvisna od podatkov, $I(\alpha) = I(\alpha|\mathbf{X})$). Večja kot je vrednost indeksa, zanimivejša je projekcija α. Algoritem za iskanje projekcij pa je postopek za numerično optimizacijo, ki išče maksimum indeksa preko vseh možnih projekcij α.

Katera projekcija podatkov je zanimiva in katera ni? Odgovor je vsekakor odvisen od področja uporabe in cilja analize. Eno od splošnih definicij zanimive projekcije sta postavila Diaconis in Friedman [38], ki sta razmišljala takole: projekcija je nezanimiva če je naključna in nestrukturirana. Glede na to, da lahko naključnost merimo z entropijo, lahko trdimo, da je projekcija nezanimiva, če ima veliko entropijo. Če se omejimo na projekcije, ki podatkov ne skalirajo, potem imajo maksimalno entropijo tiste projekcije, v katerih so primeri razporejeni po normalni porazdelitvi. Kot zanimivo lahko tako smatramo tisto projekcijo, katere porazdelitev primerov v projekciji se čim bolj razlikuje od normalne porazdelitve.

Na podlagi zgornje in raznih drugih definicij zanimivosti so različni avtorji definirali številne indekse, ki na različne načine ocenjujejo zanimivost projekcij. Kot primer indeksa vzemimo Legendrov indeks, ki ga je razvil Friedman [33]. Naj bo $z_i = \alpha x_i$ položaj i-tega primera v projekciji α. Položaje v projekciji najprej preslikamo s transformacijo R:

$$R = 2\Phi(Z) - 1, \qquad \Phi(Z) = \frac{1}{2\pi} \int_{-\infty}^{Z} e^{-\frac{1}{2t^2}}\, dt.$$

V primeru, da je Z porazdeljena normalno, bo R porazdeljena uniformno na intervalu $[-1, 1]$. Vrednost uniformne porazdelitve na tem intervalu bo konstantno $\frac{1}{2}$. Kot mero zanimivosti projekcije je tako Friedman predlagal merjenje razlike med porazdelitvijo transformirane spremenljivke R ter uniformne porazdelitve z uporabo L_2 metrike:

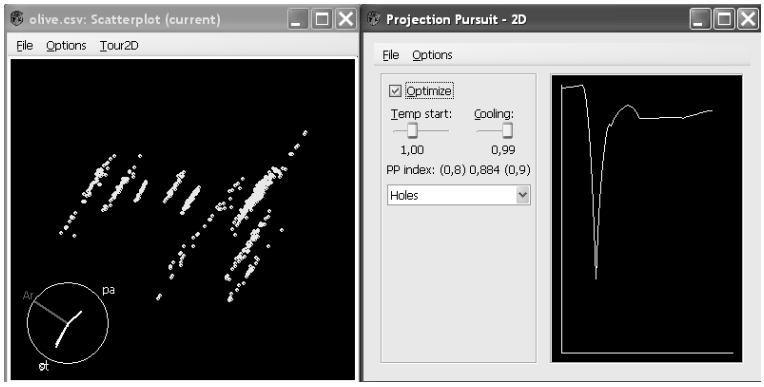
$$I^F(\alpha) = \int_{-1}^{1} \Big\{ p_R(r) - \frac{1}{2} \Big\}^2 dr,$$

kjer je p_R gostota transformirane spremenljivke R. Za računanje funkcije I^F je predlagal, da funkcijo p_R razvijemo z uporabo Legendrovih polinomov in tako zapišemo indeks kot

$$I^F(\alpha) = \frac{1}{2} \sum_{j=1}^{\infty} (2j+1) E_R^2\big\{ P_j(R) \big\} = \frac{1}{2} \sum_{j=1}^{\infty} (2j+1) \Big(\frac{1}{N} \sum_{i=1}^{N} P_j(R) \Big)^2,$$

kjer je $P_j(r)$ Legendrov polinom j-te stopnje. V praksi se običajno izračuna vsoto samo do člena m, $4 \leq m \leq 8$. Indeks je hitro izračunljiv, njegova računska zahtevnost pa narašča linearno za enodimenzionalne in kvadratično za dvodimenzionalne projekcije.

Med drugimi indeksi, ki ocenjujejo *ne*normalnost projekcije, so dobro poznani še Hermitni [46], naravni Hermitni [17], entropijski [59] in momentni indeks [59]. Za analizo klasificiranih podatkov je izbira indeksov bistveno manjša. Prvi tak indeks, namenjen iskanju enodimenzionalnih projekcij, ki ločujejo med razredi, je razvil Posse [85]. Indeks za primere iz vsakega razreda oceni gostoto projiciranih primerov z uporabo jeder in nato uporabi verjetnost napačne klasifikacije kot oceno projekcije. Podobno, a parametrično različico indeksa, je nedavno predstavil Lee [75] in temelji na linearni diskriminantni



Slika 2.16: Primer projekcijskega iskanja v programu GGobi.

analizi. Projekcijo oceni z uporabo Wilksove lambde (testne statistike iz multivariatne analize podatkov), ki predpostavi, da so primeri v projekciji v vsakem razredu porazredeljeni normalno ter da imajo enako kovariančno matriko.

Poleg izbire indeksa je ključen element projekcijskega iskanja tudi optimizacijski algoritem, s katerim iščemo čedalje zanimivejše projekcije. Nekateri indeksi so odvedljivi, zaradi česar lahko uporabimo gradientni postopek optimizacije, pri katerem se pomikamo v smeri največjega odvoda. Težava tovrstne optimizacije je, da imajo indeksi tipično veliko število lokalnih maksimumov, zaradi česar je malo verjetno, da iz naključno izbrane začetne projekcije prispemo do globalnega maksimuma. Kot rešitev je Friedman [33] predlagal optimizacijo v dveh korakih. Z grobim preiskovanjem prostora možnih projekcij naj bi najprej identificirali projekcijo z relativno visoko vrednostjo indeksa, ki bi jo nato uporabili kot začetno projekcijo pri gradientnem postopku. Nekateri avtorji namesto gradientnega postopka predlagajo tudi uporabo genetskih algoritmov [20, 83] ali simuliranega ohlajanja [18, 75].

Najbolj znan programski paket, ki omogoča projekcijsko iskanje, je GGobi [103, 102]. Na voljo ponuja različne indekse, ki jih lahko optimiziramo s simuliranim ohlajanjem. Primer uporabe je na sliki 2.16. Poleg trenutne projekcije (levo okno) je prikazan tudi graf, kako se je vrednost izbranega indeksa s časom spreminjala s spreminjanjem projekcije (desno okno).

2.4.6 FreeViz

FreeViz je novejša metoda, ki jo je razvil Demšar s sodelavci [24] in jo lahko smatramo kot enega od indeksov projekcijskega iskanja. Namenjena je iskanju zanimivih dvodimenzionalnih projekcij klasificiranih podatkov, razširiti pa jo je mogoče tudi na regresijske probleme. FreeViz se za ocenjevanje projekcij zgleduje po fizikalnemu principu privlačnih in odbojnih sil med delci v prostoru. Kot pri fizikalnih delcih se tudi primeri v projekciji privlačijo ali odbijajo. Ker nas zanimajo projekcije z dobro ločenimi razredi, definiramo, da se primeri iz istega razreda privlačijo, primeri iz različnih razredov pa odbijajo. Po-

tencialno energijo projekcije lahko tako definiramo kot vsoto sil med vsemi pari primerov. Ker je položaj primerov v projekciji določen z utežmi atributov na osi x in y, lahko s spreminjanjem uteži spreminjamo tudi potencialno energijo projekcije. Z ustreznim optimizacijskim postopkom lahko na ta način poiščemo projekcijo z minimalno potencialno energijo, ki ima kar se da dobro ločene razrede.

Natančnejša definicija postopka optimizacije je naslednja. Vzemimo, da je a_i i-ti primer v n-dimenzionalni zbirki podatkov. Koordinate $a_i^\alpha = (a_{i,x}^\alpha, a_{i,y}^\alpha)$ tega primera v projekciji $\alpha = (\alpha_x, \alpha_y)$ so:

$$a_{i,x}^\alpha = \alpha_x^T a_i = [\alpha_{1,x}, \alpha_{2,x}, ..., \alpha_{n,x}][a_{i,1}, a_{i,2}, ..., a_{i,n}]^T$$
$$a_{i,y}^\alpha = \alpha_y^T a_i = [\alpha_{1,y}, \alpha_{2,y}, ..., \alpha_{n,y}][a_{i,1}, a_{i,2}, ..., a_{i,n}]^T.$$

Naj bo $F_{j\to i}^\alpha$ dvodimenzionalni vektor, ki v projekciji predstavlja silo, s katero primer a_j^α deluje naprimer a_i^α. Velikost in smer sile sta odvisni od razdalje med primeroma ter od tega, ali pripadata primera istemu razredu ali ne. Rezultanto vseh sil F_i^α, ki delujejo naprimer a_i^α, lahko definiramo kot vsoto sil posameznih primerov:

$$F_i^\alpha = \sum_{j\neq i} F_{j\to i}^\alpha.$$

Če bi položaj ostalih primerov ostal nespremenjen, potem bi lahko s premikom primera a_i^α v smeri vektorja F_i^α dosegli zmanjšanje potencialne energije projekcije. Ker je položaj primerov določen z izbiro projekcije α, lahko dejanski premik primera dosežemo z ustreznim spreminjanjem vrednosti vektorjev α_x in α_y. Za premik primera v dani smeri je potrebno omenjenima vektorjema prišteti poljuben pozitiven večkratnik vektorjev $G_{i,x}$ in $G_{i,y}$, ki ju definiramo kot

$$\mathbf{G}_i = \begin{bmatrix} G_{i,x} \\ G_{i,y} \end{bmatrix} = F_i^\alpha a_i^T = \begin{bmatrix} F_{i,x}^\alpha \\ F_{i,y}^\alpha \end{bmatrix} a_i^T.$$

Z omenjeno spremembo projekcije bi upoštevali samo vpliv enega primera, primera a_i. Ker pa s spreminjanjem projekcije hkrati spremenimo položaje vseh primerov, bi lahko s tem dosegli povečanje napetosti med ostalimi primeri, zaradi česar bi bila potencialna energija take projekcije še večja. Da bi projekciji dejansko zmanjšali potencialno energijo, definiramo ciljno spremembo $\mathbf{G}$ kot vsoto sprememb posameznih primerov $\mathbf{G}_i$, $\mathbf{G} = \sum_{i=1}^N \mathbf{G}_i$.

Opisani postopek optimizacije izvajamo iterativno. Pomembno je, da v vsaki iteraciji projekcijo α spremenimo le malenkostno, saj s tem poleg položaja primerov spremenimo tudi smer in jakost sil med njimi, zaradi česar se spreminja tudi gradientna matrika $\mathbf{G}$. Optimizacijo projekcije ustavimo takrat, ko postane sprememba potencialne energije v nekaj zaporednih korakih optimizacije dovolj majhna (naprimer $< 1\%$).

Sam algoritem optimizacije je neodvisen od izbrane definicije, s katero določamo vrednost sil med primeri. Ker želimo ločevati med razredi, FreeViz definira predznak sile

Podatki	# primerov	# atributov	# razredov
wine	178	13	3
zoo	101 (84)	16	7 (4)
housing	506	13	2
imports-85	201	25	2
monks 3	432	6	2

Tabela 2.2: Osnovne informacije o uporabljenih domenah.

v odvisnosti od tega, ali pripadata primera istemu ali različnima razredoma; v prvem primeru je sila privlačna, v drugem pa odbojna. Jakost sile je nato odvisna od razdalje r med primeroma v projekciji. Če primera pripadata istemu razredu, potem se z manjšanjem razdalje med primeroma manjša tudi sila med njima ($F \sim r$). S tem se več pozornosti posveti primerom, ki ležijo narazen, in manj tistim, ki so že sedaj blizu. V primeru, da primera pripadata različnima razreda, velja obratna sorazmernost; bolj kot sta si primera blizu, večja je odbojna sila med njima ($F \sim 1/r$). Namesto linearne relacije med silo in razdaljo lahko uporabimo tudi kakšno drugačno odvisnost, naprimer kvadratno (r^2) ali eksponentno (e^r), vendar izbira ne vpliva bistveno na končno dobljeno projekcijo.

2.5 Ocenitev izbranih metod za odkrivanje znanja v podatkih

Kot je razvidno iz podpoglavja o vizualizacijskih metodah, imamo za analizo podatkov na voljo pester nabor metod, ki jih lahko uporabimo. Obilnost izbora metod pa je sama po sebi popolnoma nepomemben podatek, saj ne pove ničesar o dejanski uporabnosti teh metod. Brez težav bi se lahko naprimer spomnili neštetih novih vizualizacijskih metod s kombiniranjem principov različnih obstoječih metod, pa s tem ne bi nič doprinesli k uspešnejši analizi podatkov. Bistvenega pomena je sama uporabnost metod – torej njihova sposobnost prikaza podatkov na način, da postanejo različne strukture, ki so vsebovane v podatkih, vidne in razumljive analitiku.

Uporabnost posameznih vizualizacijskih metod se razlikuje glede na vrsto iskane strukture ter vrsto analiziranih podatkov (diskretni/zvezni atributi, nadzorovano/nenadzorovano učenje, dimenzionalnost podatkov ipd). Da bi ugotovili, koliko nam vizualizacija pomaga pri strojnem učenju, bomo v tem razdelku za šest izbranih vizualizacijskih metod ocenili njihovo uspešnost pri odkrivanju različnih struktur in zakonitosti na različnih problemskih domenah. Izbrane vizualizacijske metode so tiste, ki so v literaturi pridobile največ veljave in so hkrati po našem mnenju najprimernejše za analizo podatkov, s katerimi se tipično srečujemo pri nadzorovanem strojnem učenju. Glede na ta kriterija smo za ocenitev izbrali razsevni diagram, metodo radviz, paralelne koordinate, splošne linearne projekcije ter mozaični in pregledni diagram. Linearne projekcije, ki smo jih ocenjevali, so bile dobljene tako z uteženo analizo glavnih komponent kot tudi z metodo FreeViz.

Izbrane vizualizacijske metode smo ocenili na petih različnih problemskih domenah iz repozitorija UCI [80]: wine, zoo, housing, imports-85 ter monks 3. Osnovne informacije o podatkih so v tabeli 2.2. Domene smo izbrali tako, da smo dosegli čim večjo variabilnost podatkov – zbirke tako vsebujejo od 101 do 506 primerov ter dva do sedem razredov. Poleg tega nekatere zbirke vsebujejo samo zvezne atribute, nekatere samo diskretne, nekatere pa zvezne in diskretne atribute.

Pri vsaki zbirki podatkov smo vsako vizualizacijsko metodo ocenili glede na pet pomembnih kriterijev (nalog):

- **Gruče:** Ali je mogoče v prikazih odkriti samostojne gruče primerov? Te gruče lahko vsebujejo primere samo enega razreda ali pa so mešanica različnih razredov. Namen kriterija je oceniti, kako dobro lahko z metodo odkrijemo ločene skupine primerov s podobnimi lastnostmi.
- **Gruče razredov:** Ali lahko najdemo prikaze, v katerih so različni razredi (relativno) dobro ločeni med sabo? Kako enostavno je ta ločenost razredov v prikazu opazna?
- **Osamelci:** Ali je mogoče v različnih prikazih podatkov odkriti primere, pri katerih vrednosti bistveno odstopajo od drugih primerov? Kako enostavno je mogoče te primere odkriti?
- **Pomembnost atributov:** Različni atributi so različno pomembni pri ločevanju med razredi. Ali je mogoče s pomočjo prikazov sklepati o pomembnosti posameznih atributov za ločevanje med razredi?
- **Pravilo/Model:** Ali je mogoče s pomočjo enega ali več prikazov definirati odločitveno pravilo ali model, s katerim bi lahko (relativno) uspešno klasificirali nove primere. Kako enostavno je tako pravilo razbrati iz prikaza in kakšna je njegova interpretabilnost (še posebej pri uporabi projekcijskih vizualizacijskih metod)?

Za ocenjevanje metod smo uporabili tristopenjsko lestvico. V primeru, da metoda ni primerna za dani kriterij, smo v tabeli rezultatov pustili prazen prostor. V primeru, da se je metoda izkazala kot delno uporabna, smo to označili s simbolom '+', če se je izkazala kot odlična, pa s simboloma '++'. Za vsako zbirko podatkov in vsako vizualizacijsko metodo smo si ogledali večje število prikazov, ki so vsebovali različne podmnožice atributov v različnih postavitvah. Primeri teh prikazov so na slikah 2.17 – 2.21. Ocene, ki smo jih dodelili metodam, tako niso določene samo na osnovi teh prikazov, ampak tudi na osnovi ostalih pregledanih prikazov.

2.5.1 Rezultati ocenjevanja

V nadaljevanju sledi krajši opis uspehov in težav, ki so jih vizualizacijske metode imele pri odkrivanju omenjenih struktur in lastnosti na izbranih zbirkah podatkov. Ocene metod so zbrane v tabeli 2.3.

Domena wine

Domena ima 178 primerov, 13 zveznih atributov in tri razrede. Podatki vsebujejo rezultate kemične analize vin iz določene italijanske regije, ki so bila pridelana na tri različne načine.

Na tej domeni je bila večina izbranih vizualizacijskih metod zelo uspešnih. Razsevni diagram in metoda paralelnih koordinat sta se izkazala odlično pri vseh nalogah. Metoda paralelnih koordinat je še posebej uspešna pri določitvi modela – črte, ki ustrezajo primerom iz istega razreda, potekajo zelo regularno preko posameznih osi, zaradi česar lahko za vsak razred definiramo neke vrste kvalitativni model poteka črt.

Metoda radviz in linearne projekcije so se prav tako dobro odrezale. Manj uspešni sta bili samo pri identifikaciji pomembnih značilk ter definiranju interpretabilnega modela. Ker sta obe metodi projekcijski, pri generiranju prikaza izgubimo dejansko informacijo o vrednostih primerov pri posameznih atributih. Čeprav je ločenost razredov v prikazih dobra, je zato težko najti pravilo, ki bi z uporabo originalnih atributov to ločenost pojasnjevalo.

Za vizualizacijo podatkov z mozaičnim diagramom smo atribute diskretizirali z uporabo entropijskega postopka diskretizacije. Iz prikaza lahko ugotovimo, da je posamezne razrede mogoče ločiti, sama diskretizacija pa nam onemogoča sklepanje o tem, ali gre za dobro ločene gruče ali pa si primeri ležijo zelo blizu in je ločenost razredov le posledica dobro izbrane diskretizacijske meje.

Pregledni diagram se je pri analizi teh podatkov odrezal precej povprečno. Vse naloge lahko opravimo le z opazovanjem vsakega atributa posebej – tako je naprimer v prikazu nemogoče identificirati gručo ločenih primerov, če je za to potrebno *hkratno* opazovanje dveh ali več atributov.

Domena zoo

Podatki vsebujejo informacije o 101 vrsti živali. Le-te so opisane s 16 diskretnimi atributi. Razred določa, v katero od sedmih skupin živali spada. Pri vseh vizualizacijskih metodah, razen pri mozaičnem in preglednem diagramu, prikaz primerov s sedmimi različnimi barvami ni bil najuspešnejši, zato smo pri teh metodah odstranili tri številčno najmanj zastopane skupine živali.

Razsevni diagram je imel pri teh podatkih nekaj težav, saj bi moral za uspešno ločevanje med vsemi štirimi razredi hkrati vizualizirati vsaj tri atribute. Ker je osnovna različica razsevnega diagrama omejena na dva atributa, moramo tako uporabiti več prikazov, če želimo ločiti med vsemi razredi.

Metoda paralelnih koordinat je bila ponovno brezhibna pri vseh nalogah. Ker gre za vizualizacijo diskretnih atributov, smo pri vsaki vrednosti atributov prikazali še distribucijo vrednosti razreda, s čimer se še dodatno izboljša percepcija ločenosti razredov. Ena od težav metode je, da je ločenost razredov odvisna od vrstnega reda prikazanih atributov; na sliki 2.18 naprimer ne bi mogli ločiti rib od ostalih živali, če atributa *milk* in *toothed* ne bi bila prikazana kot sosednja.

Linearne projekcije in metoda radviz so se odrezali uspešno, čeprav imajo ponovno težave z ocenjevanjem pomembnosti značilk in definiranjem interpretabilnega modela. Iz prikaza radviz na sliki 2.18 je naprimer brez uporabe interaktivnih tehnik nemogoče sklepati o tem, kaj velja za nevretenčarje v središču kroga (tja bi se namreč preslikali tako primeri z vrednostmi (0,0,0) kot tisti z vrednostmi (1,1,1)).

Metodi, ki sta uspešno prikazali vseh sedem razredov hkrati, sta mozaični in pregledni diagram. Mozaični diagram je bil odličen tudi glede razumljivosti prikazanega modela. Pregledni diagram za razliko od paralelnih koordinat ni omejen z vrstnim redom prikazanih atributov, je pa z njim veliko teže odkriti pravila, ki vključujejo več atributov hkrati (ista težava, kot pri podatkih wine).

Domena housing

Podatki vsebujejo različne informacije za 506 stanovanj v Bostonu. Stanovanja so opisana s 13 zveznimi atributi ter enim diskretnim, binaren razred pa ponazarja ceno stanovanja.

Razsevni diagram in metoda paralelnih koordinat sta se pri vseh nalogah izkazali zelo dobro. Paralelnim koordinatam edine težave povzroča prekrivanje črt (posledica večjega števila primerov), kar otežuje induciranje pravil za ločevanje med razredoma.

Odlično so se odrezale tudi linearne projekcije in metoda radviz. V linearni projekciji na sliki 2.19 je vidna tudi potencialna slabost te metode. O pomembnosti posameznega atributa namreč običajno sklepamo na podlagi dolžine vektorja v prikazu, ki ustreza temu atributu – daljši kot je vektor, pomembnejši je atribut za diskriminacijo. V tem prikazu je daleč najdaljši vektor atributa *NOX*, zaradi česar bi sklepali, da je to najpomembnejši atribut. Podrobnejša analiza na žalost pokaže, da atribut ni posebno informativen pri ločevanju med razredi; njegov edini vpliv je, da razprši majhno skupino rdečih primerov. Sklepanje o pomembnosti atributov zgolj na podlagi dolžine vektorja je tako lahko v nekaterih primerih zavajajoče in privede do napačnih zaključkov.

Z mozaičnim diagramom lahko definiramo dokaj uspešno pravilo za ločevanje med razredi, popolnoma neuspešna pa je metoda pri iskanju osamelcev. V nasprotju s tem so v preglednem diagramu osamelci lepo vidni, manj uspešni pa smo pri definiranju napovednega modela.

Domena imports-85

V tej zbirki so navedeni podatki o avtomobilih, uvoženih leta 1985. Domena vsebuje 15 zveznih in deset diskretnih atributov. Razred, ki predstavlja ceno avtomobila, je v originalni zbirki zvezen, vendar smo ga za potrebe analize binarizirali.

Razsevni diagram in metoda paralelnih koordinat sta se pri tej domeni izkazala odlično. Malce slabše od njiju se je odrezal radviz, s katerim je bilo teže opaziti gruče primerov. Še bistveno slabše pa so se izkazale linearne projekcije, s katerimi so bile ponovno težave pri ocenjevanju pomembnosti atributov ter pri gradnji interpretabilnega modela. Atributi, ki so v projekcijah s svojo dolžino vektorjev nakazovali pomembnost, so se pri podrobnejši analizi pogosto izkazali kot neuporabni za ločevanje med razredi.

Še bolj presenetljivo je, da bi iz smeri vektorja *num-of-cylinders* na sliki 2.20 sklepali, da imajo avtomobili z večjim številom cilindrov nižjo ceno, čeprav je iz razsevnega diagrama razvidno, da je resnična odvisnost ravno obratna.

Mozaični in pregledni diagram sta bila relativno uspešna. Mozaični diagram je bil manj primeren za iskanje gruč in osamelcev, pregledni diagram pa ima težave pri indukciji pravil z uporabo več kot enega atributa.

Domena monks 3

Monks 3 je sintetična domena s šestimi diskretnimi atributi *a1* do *a6*. Binaren razred predstavlja koncept $(a5 = 3 \wedge a4 = 1) \vee (a5 \neq 4 \wedge a2 \neq 3)$.

Z osnovno različico razsevnega diagrama, ki hkrati prikaže dva atributa, bi podobno kot pri domeni zoo, definirali samo del ciljnega koncepta – ostala bi majhna skupina primerov, med katerimi ne bi uspeli ločiti. V prikazu na sliki 2.21 smo dodatno vizualizirali še atribut *a4* tako, da smo njegove vrednosti preslikali v različne oblike prikazanega simbola. Čeprav je percepcija različnih oblik simbola bistveno slabša od percepcije položaja (glej [12, 13]), nam to v tem enostavnem primeru zadostuje za indukcijo celotnega koncepta. Težavo s prikazom je imela tudi metoda paralelnih koordinat, saj ne glede na vrstni red vizualiziranih atributov ni mogoče najti takega zaporedja, s katerim bi bilo mogoče definirati celotno pravilo za ločevanje med razredoma.

Uspešnosti metode radviz in linearnih projekcij se na tej domeni bistveno razlikujeta. Linearne projekcije najdejo rotirano različico razsevnega diagrama, ki z majhno pomočjo atributa *a4* odlično loči med razredoma. V nasprotju s tem je prikaz radviz popolnoma neinterpretabilen – čeprav vsaka od gruč vsebuje zgolj primere enega razreda, ni mogoče zaradi velikega števila gruč razbrati niti dela koncepta.

Velike razlike so bile tudi med mozaičnim in preglednim diagramom. Mozaični diagram zelo jasno prikaže celoten koncept, medtem ko lahko v preglednem diagramu le s težavo odkrijemo posamezne dele koncepta.

2.5.2 Zaključki

Test metod, ki smo ga opravili, nikakor ni izčrpen – uporabili smo majhno število vizualizacijskih metod kot tudi majhno število testnih domen. Ne glede na to lahko iz rezultatov vseeno sklepamo o določenih prednostih in slabostih uporabljenih metod. V tabeli 2.4 so prikazane končne ocene metod za posamezne naloge. Ocene so izračunane na podlagi tabele 2.3 kot vsota simbolov '+' na vseh testnih problemskih domenah.

Razsevni diagram je od vseh metod najuspešnejši pri odkrivanju tako linearnih kot tudi nelinearnih relacij med pari atributov. Zelo uspešno ga lahko uporabimo za odkrivanje pravil za ločevanje med razredi, vendar le, če nam za to zadostuje uporaba dveh atributov. Več težav ima metoda pri domenah z večjim številom razredov, kjer je za ločevanje med njimi običajno potrebno hkrati opazovati večje število atributov.

Z metodo paralelnih koordinat lahko enostavno in uspešno odkrivamo širok spekter

Diagrami	Gruče	Gruče razredov	Osamelci	Pomembnost atributov	Pravilo/ model
wine					
razsevni	++	++	++	++	++
paralelne k.	++	++	++	++	++
linearne p.	++	++	++	+	+
radviz	++	++	++	+	+
mozaični	+	++		++	++
pregledni	+	+	+	++	+
zoo					
razsevni		+	++	++	+
paralelne k.		++	++	++	++
linearne p.		++	++	+	+
radviz		++	++	+	+
mozaični		++	++	++	++
pregledni		++	++	++	+
housing					
razsevni	++	+	++	++	++
paralelne k.	++	+	++	++	+
linearne p.	++	+	++	+	++
radviz	++	+	++	++	++
mozaični	+	+		++	++
pregledni	++	+	++	++	+
imports-85					
razsevni	++	++	++	++	++
paralelne k.	++	++	++	++	++
linearne p.	++	+	++		
radviz	+	+	++	++	++
mozaični		+	+	++	++
pregledni	+	+	++	++	+
monks 3					
razsevni		+		++	+
paralelne k.		+		+	+
linearne p.		++		++	++
radviz					
mozaični		++		++	++
pregledni				+	+

Tabela 2.3: Ocene vizualizacijskih metod na petih problemskih domenah.

zakonitosti v podatkih – gruče, osamelce, pravila, korelacije med atributi, itd. Težave nastopijo v primeru, ko imamo večje število primerov (npr. > 500), pri čemer postane prikaz nasičen s črtami. Če je potek črt poleg tega zelo neregularen, to še dodatno oteži

Diagrami	Gruče	Gruče razredov	Osamelci	Pomembnost atributov	Pravilo/ model
razsevni	6	7	8	10	8
paralelne k.	6	8	8	9	8
linearne p.	6	8	8	5	6
radviz	5	6	8	6	6
mozaični	2	8	3	10	10
pregledni	4	5	7	9	5

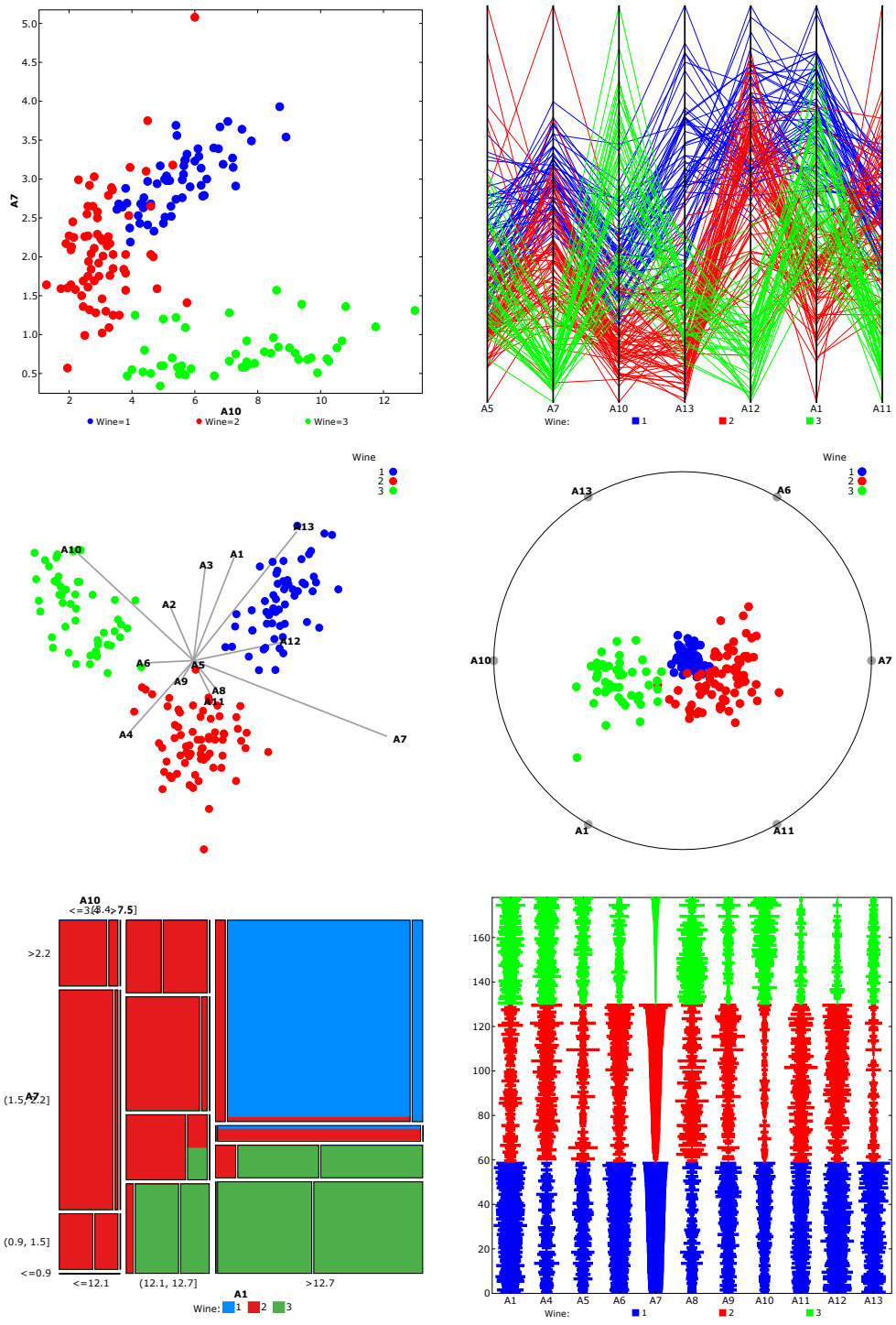
Tabela 2.4: Končne ocene vizualizacijskih metod, izračunane na podlagi tabele 2.3 kot vsota simbolov '+' na vseh problemskih domenah.

percepcijo zakonitosti. Uspešnost prikaza je močno odvisna tudi od izbora in vrstnega reda prikazanih atributov – zakonitost med dvema atributoma je opazna samo v primeru, da atributa postavimo enega zraven drugega.

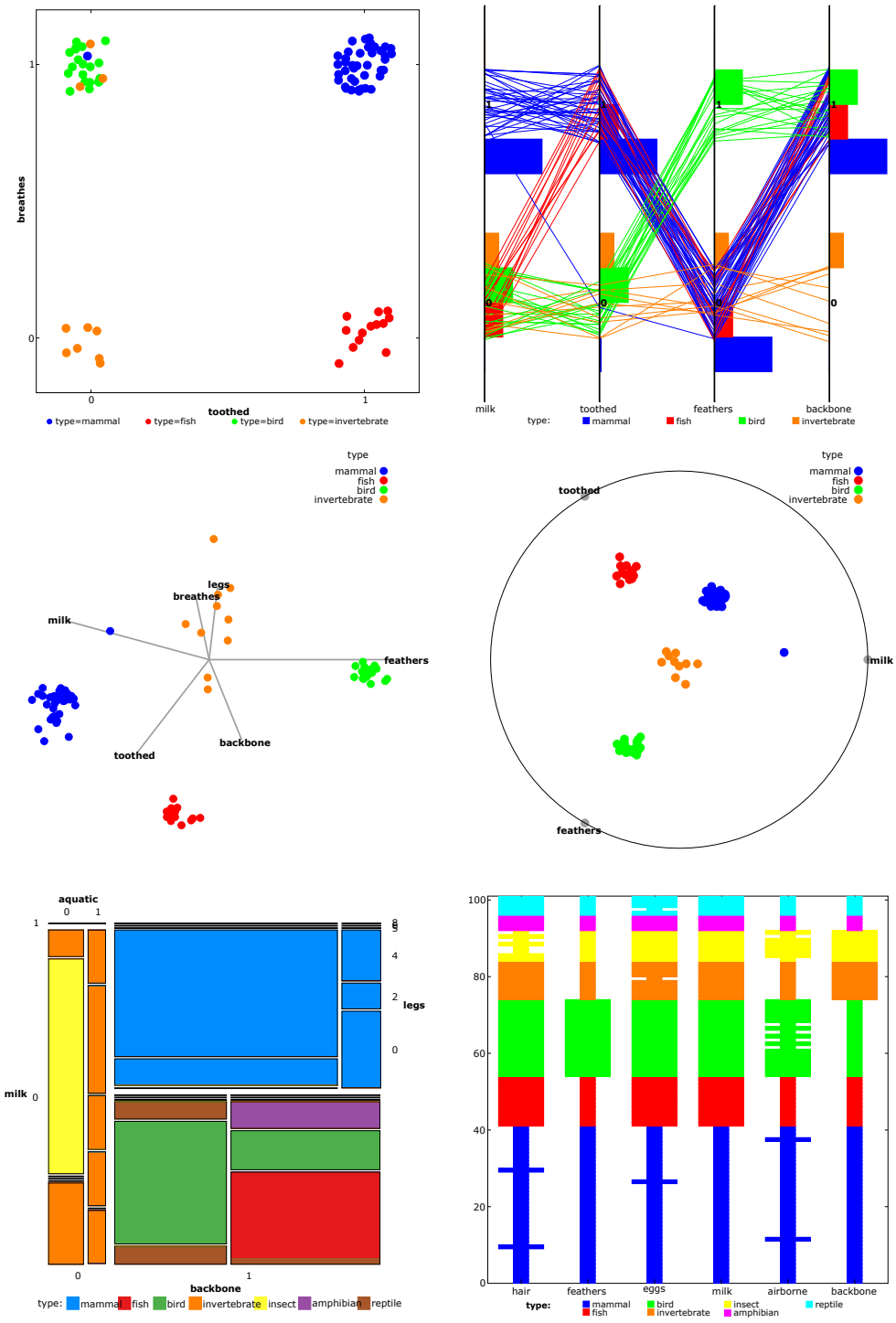
Linearne projekcije in metoda radviz so zelo sorodne, zato imajo podobne prednosti in slabosti. Ker sta metodi projekcijski in hkrati vizualizirata več atributov, lahko pogosto najdemo zanimive projekcije z dobro ločenimi razredi. Težave se včasih pojavijo pri interpretaciji takih projekcij, zato je ti metodi smiselno kombinirati z metodo paralelnih koordinat. Manj primerni sta metodi pri vizualizaciji diskretnih atributov, saj nastanejo v projekciji gruče, znotraj katerih ležijo primeri z različnimi vrednostmi, kar še dodatno otežuje sklepanje.

Mozaični diagram se je izkazal kot metoda, s katero na najrazumljivejši način prikažemo modele, ki vključujejo večje število atributov. Diagrama za domeni zoo in monks 3 sta naprimer idealna zgleda za pregovor, da slika pove več kot tisoč besed. Ker je potrebno zvezne atribute predhodno diskretizirati, je metoda manj uspešna pri odkrivanju osamelcev ter gruč v podatkih. Težave nastanejo tudi pri vizualizaciji diskretnih atributov z velikim številom vrednosti – pri vizualizaciji skupine takih atributov postanejo namreč posamezni pravokotniki v prikazu nerazpoznavno majhni.

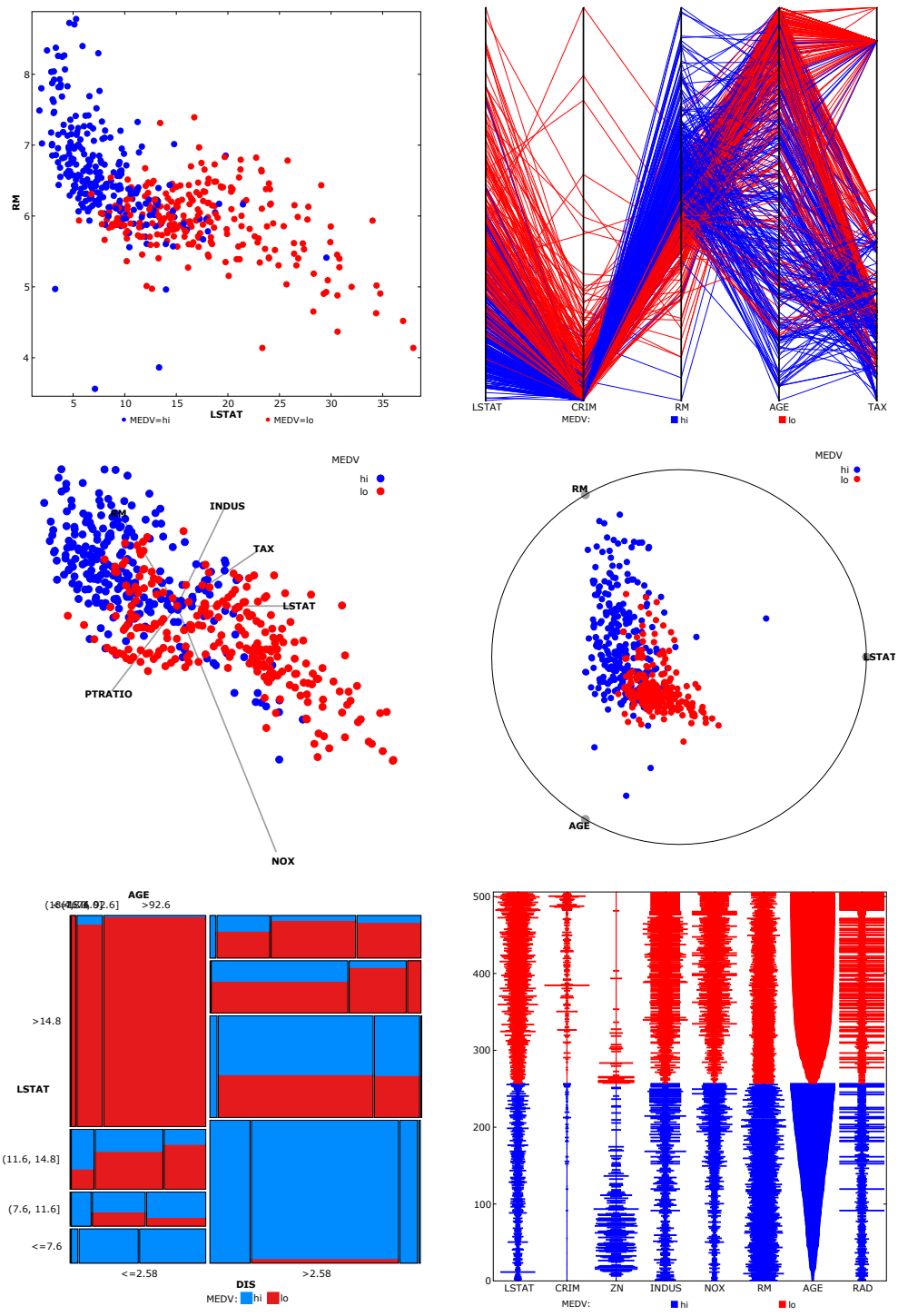
S preglednim diagramom je mogoče uspešno odkriti pomembne atribute ter linearne odvisnosti med atributi. Več težav ima metoda vsekakor s prikazom modela, saj je z izjemo enostavnih pravil, ki vključujejo samo en atribut, indukcija modelov netrivialna.



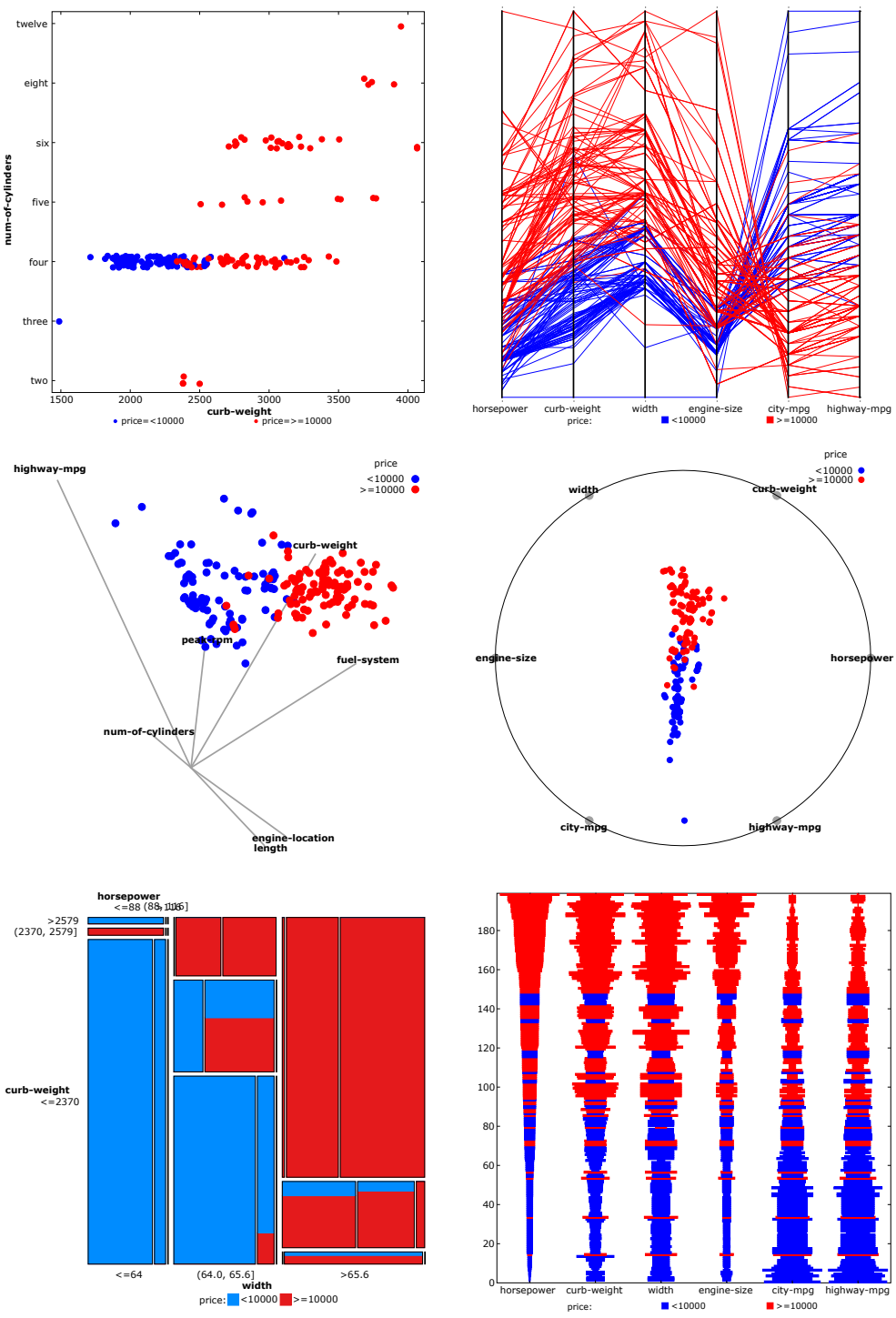
Slika 2.17: Izbrani prikazi za domeno wine.



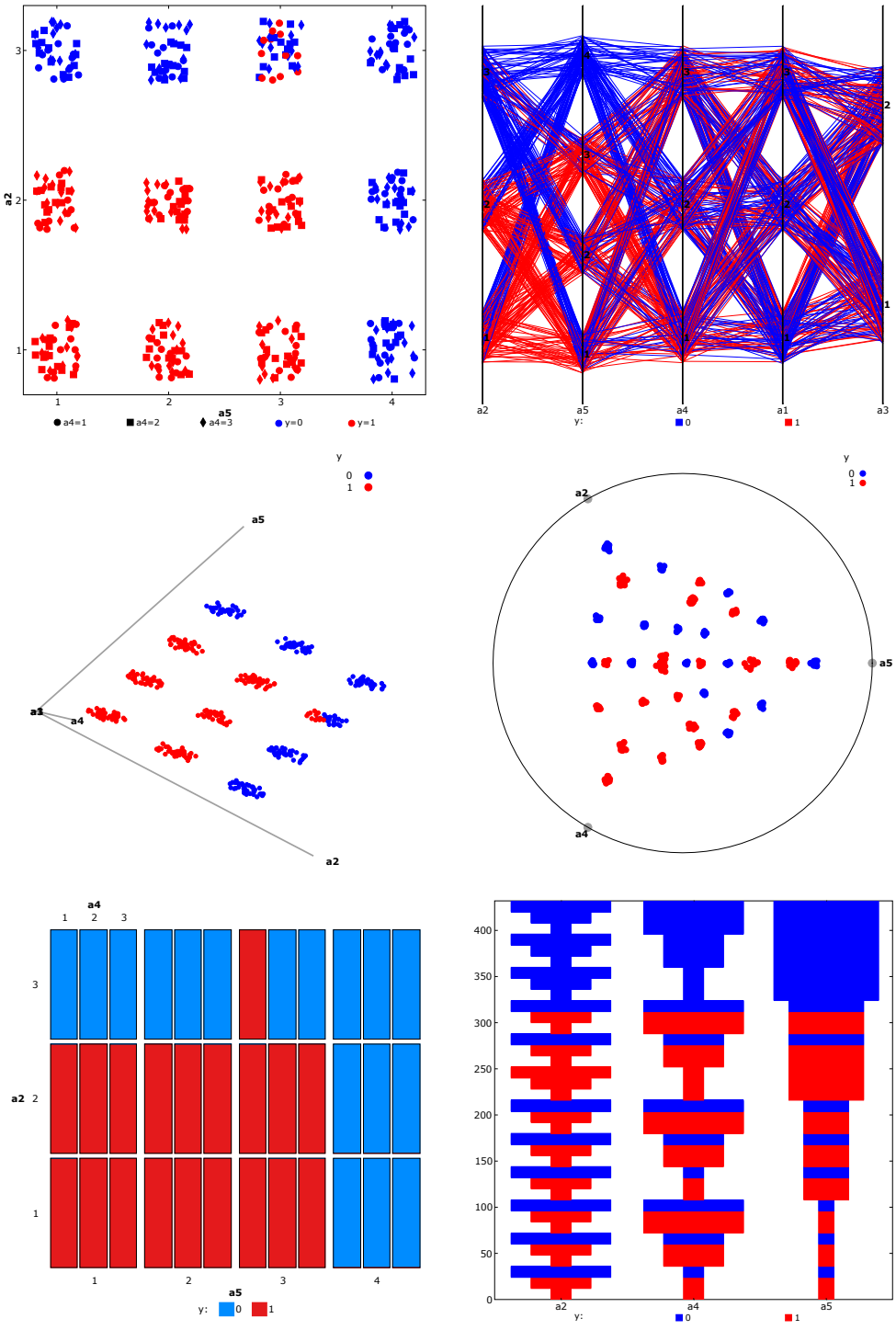
Slika 2.18: Izbrani prikazi za domeno zoo.



Slika 2.19: Izbrani prikazi za domeno housing.



Slika 2.20: Izbrani prikazi za domeno imports-85.



Slika 2.21: Izbrani prikazi za domeno monks 3.

Poglavje 3

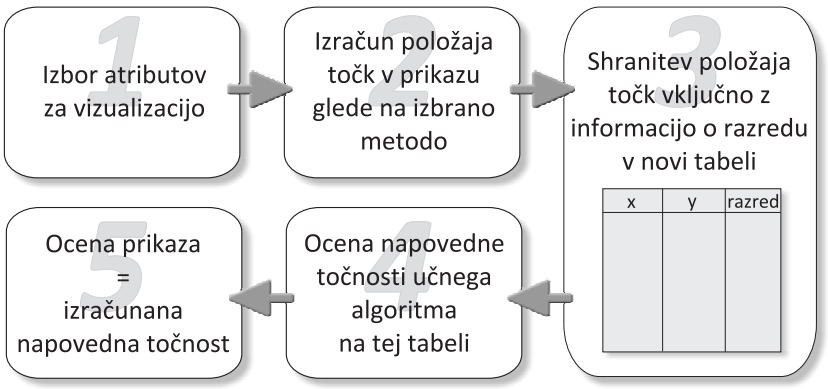
Ocenjevanje in rangiranje točkovnih prikazov

V razdelku 2.4 smo prikazali skupino metod, ki jih je mogoče uporabiti za iskanje zanimivih linearnih projekcij. Del teh metod je primeren za uporabo v primeru nadzorovanega, del pa v primeru nenadzorovanega učenja. V tem poglavju bomo predstavili našo novo metodo za iskanje zanimivih prikazov klasificiranih podatkov. Za razliko od metod, omenjenih v razdelku 2.4, je ta primerna za iskanje zanimivih prikazov, generiranih s poljubno vizualizacijsko metodo, ki posamezne primere prikaže kot točke v prostoru, pri čemer je položaj točk določen z vrednostmi vizualiziranih atributov. Primeri takih metod so razsevni diagram, metodi radviz in polyviz ter splošne linearne projekcije. Za prikaze, generirane z uporabo takih vizualizacijskih metod, bomo odslej uporabljali izraz projekcije.

V nadaljevanju bomo najprej predstavili podrobnosti metode. Opisali bomo, kakšen vpliv ima izbira učnega algoritma in cenilne funkcije na ocenjevanje ter kako lahko z ustrezno hevristiko bistveno pohitrimo iskanje zanimivih projekcij. Kot demonstracijo primernosti metode bomo prikazali rezultate analize, kjer smo primerjali, kako podobno rangirata projekcije algoritem in človek. Rezultat metode ni ena sama projekcija (kot pri ostalih postopkih), ampak seznam rangiranih projekcij. Opisali bomo tri načine, kako lahko ta seznam uporabimo za ocenjevanje pomembnosti atributov, odkrivanje interakcij med atributi ter za detekcijo osamelcev. Pokazali bomo tudi rezultate, ki jih lahko dosežemo, če najbolje rangirane projekcije uporabimo kot napovedne modele.

3.1 Metoda VizRank

Metoda VizRank [71, 72, 73] omogoča rangiranje različnih točkovnih prikazov glede na njihovo zanimivost. Uporabimo jo lahko v primeru nadzorovanega učenja, t.j. učenja, pri



Slika 3.1: Shematski diagram metode VizRank.

katerem želimo najti pravilo, s katerim bi na podlagi vrednosti atributov čim natančneje napovedali vrednost posebnega atributa, ki mu pravimo razred. Pri analizi takih podatkov so za nas najzanimivejše tiste projekcije, v katerih ležijo primeri istega razreda čim bolj skupaj in so kar se da ločeni od primerov, ki pripadajo drugim razredom. S pomočjo takih projekcij lahko enostavno poiščemo skupne lastnosti, ki veljajo za primere z istim razredom ter razlike, na osnovi katerih ločimo razrede med sabo.

Pri danih podatkih in izbrani vizualizacijski metodi VizRank oceni možne projekcije s ponavljanjem naslednjega postopka (glej sliko 3.1). Najprej izbere podmnožico atributov namenjenih vizualizaciji (postopek za izbor je natančno opisan v razdelku 3.2). Za te atribute nato glede na izbrano vizualizacijsko metodo izračuna x in y koordinate točk v projekciji ter jih shrani v novi tabeli, kjer poleg tega doda še informacijo o vrednosti razreda pri posameznih primerih. Z uporabo izbranega učnega algoritma nato oceni, kako dobro lahko le-ta napove vrednost razreda zgolj na podlagi podatkov v tej tabeli. Podatki, iz katerih učni algoritem zgradi model, torej ne vsebujejo vrednosti originalnih atributov, temveč samo vrednosti tistih "vizualnih" atributov, ki so uporabniku na voljo pri ogledu projekcije. Če je ločenost razredov v projekciji dobra, lahko pričakujemo, da bo točnost algoritma visoka, v primeru slabše ločenosti pa temu primerno slabša. Točnost, ki jo algoritem doseže, lahko torej uporabimo kot numerično oceno zanimivosti projekcije. Projekcije lahko glede na to oceno rangiramo in uporabnik se lahko namesto pregledovanja projekcij v naključnem vrstnem redu osredotoči zgolj na ogled manjše skupine najbolje rangiranih projekcij, ki mu bodo nudile najboljši vpogled v pomembne zakonitosti.

3.1.1 Učni algoritem

Izbira učnega algoritma je eden ključnih elementov, ki vplivajo na uspešnost ocenjevanja projekcij. Čeprav obstaja veliko število algoritmov, niso vsi enako primerni za naš namen. Kot rezultat učenja si lahko predstavljamo, da vsak učni algoritem definira odločitvene meje, ki ločujejo med področji, znotraj katerih naj bi bili samo primeri, ki pripadajo istemu razredu. Različni algoritmi imajo različne omejitve glede oblik teh mej in vse oblike

niso enako primerne za namen vizualizacije. Vzemimo primer odločitvenih dreves. Meje so v tem primeru lahko zgolj vertikalne in horizontalne črte, ki bi projekcijo razdelile na skupino pravokotnikov. Čeprav se odločitvena drevesa pogosto dobro obnesejo kot klasifikatorji, si lahko zamislimo enostavno projekcijo z lepo ločenimi razredi (naprimer tako z dvema poševno podolgovatima gručama točk), kjer bi bila dosežena napovedna točnost slaba samo zaradi omejitve v obliki mej. Na drugi strani bi lahko bila točnost na kakšni drugi projekciji, kjer je prekrivanje med razredi večje, boljša samo zaradi tega, ker bi bile gruče rotirane na algoritmu bolj prijazen način. Če želimo ocenjevati projekcije podobno kot jih človek, potem se moramo takim nerodnostim izogniti. Zato pa potrebujemo tak učni algoritem, ki poišče odločitvene meje, ki se čim bolj skladajo z "vizualnimi" mejami, kot jih v projekcijah nezavedno opazimo ljudje.

Algoritem, ki nima omejitev glede oblik odločitvenih mej in za katerega na podlagi empiričnih rezultatov in primerjav (glej razdelek 3.3) verjamemo, da je najprimernejši za namen ocenjevanja projekcij, je metoda k-najbližjih sosedov (k-NN). k-NN je algoritem, ki napove vrednost razreda za nov primer na osnovi informacije o tem, kakšna je porazdelitev razredov pri k najbližjih primerih. Vsak od k sosedov glasuje za svoj razred, njihov glas pa je običajno utežen glede na razdaljo od primera. V naši implementaciji smo glasove utežili s funkcijo e^{-t^2/s^2}, kjer je t razdalja do primera, parameter s pa je izbran tako, da je vpliv najbolj oddaljenega primera enak 0,001. Rezultat glasovanja je verjetnostna porazdelitev razredov, primer pa klasificiramo v razred z največjo verjetnostjo.

Da bi definirali soseščino primerov, moramo najprej izbrati metriko za merjenje razdalj med primeri. V naših poskusih smo uporabljali evklidsko metriko, ki ima številne želene matematične lastnosti, kot je naprimer invariantnost glede na rotacijo projekcije. Čeprav se ne sklada popolnoma s tem, kako ljudje ocenjujemo različnost med objekti [92, 93], je dobra aproksimacija za to, kako ocenjujemo razdalje med njimi [22].

Ker pri predikciji razreda glasuje k najbližjih sosedov, je potrebno definirati pravilo za določanje vrednosti parametra k. Na eni strani želimo izbrati dovolj velik k, da bo dobljena predikcija zanesljiva, po drugi strani pa dovolj majhen, da ne bomo upoštevali prevelikega deleža primerov. Pomembnost tega parametra smo zmanjšali s tem, da smo izbrali uteževanje primerov glede na razdaljo, s čimer imajo bližnji primeri večji vpliv na napoved kot bolj oddaljeni. Eno od pogosto uporabljenih pravil za določanje vrednosti parametra k je definiral Dasarathy [23], ki je predlagal uporabo formule $k = \sqrt{N}$, kjer je N število primerov v zbirki podatkov. V eksperimentu, ki smo ga opravili in je opisan v razdelku 3.3, smo ugotovili, da med VizRankovim in človeškim rangiranjem projekcij dosežemo še boljše ujemanje, če za k izberemo še večje vrednosti – naprimer po enačbi $k = N/c$, kjer je c število različnih razredov v podatkih.

Glede na to, da je iskanje k najbližjih sosedov za en primer potreben čas reda $O(N)$, pri čemer je N število primerov v zbirki podatkov, je za enostavno različico algoritma k-NN časovna zahtevnost $O(N^2)$. Za iskanje najbližjih sosedov obstajajo različne pohitritve in ena najhitrejših je z uporabo k-D (k-dimenzionalnih) dreves [4, 94]. k-D drevesa so posplošitev binarnih iskalnih dreves, kjer namesto enega ključa uporabimo k ključev

(dimenzij). Vsako od notranjih vozlišč ima dva naslednika, s pomočjo katerih rekurzivno razdelimo primere v dve skupini glede na enega od k ključev. Rekurzivno deljenje se ustavi takrat, ko je v vozlišču manj kot določeno število primerov. Drevo je mogoče zgraditi v času $O(N \log N)$, iskanje k najbližjih sosedov enega primera pa nato s pomočjo drevesa izvedemo v času $O(\log N)$ [34].

3.1.2 Cenilna funkcija

V strojnem učenju obstaja veliko število cenilnih funkcij namenjenih ocenjevanju uspešnosti klasifikatorjev. Ena od mer, ki se zelo pogosto uporablja, je klasifikacijska točnost. Definirana je kot delež primerov, pri katerih je klasifikator pravilno napovedal razred. Klasifikacijska točnost ima na žalost funkcijo napake 0/1, zaradi česar je zelo neobčutljiva v primerih verjetnostne klasifikacije; za klasifikacijsko točnost je popolnoma vseeno, če je klasifikator napovedal pravilni razred z verjetnostjo 1 ali pa naprimer z verjetnostjo 0,51. Ker v našem primeru glavni cilj ni ocenjevanje klasifikatorja, ampak ocenjevanje projekcije, je bolj smiselno uporabiti mero, ki upošteva dejanske napovedane verjetnosti. Za verjetnostni klasifikator, kar k-NN je, lahko naprimer izračunamo povprečno verjetnost $\overline{P}$, ki jo je klasifikator določil pravilnemu razredu:

$$\overline{P} = E(P_f(\mathbf{y}|\mathbf{x})) = \frac{1}{N}\sum_{i=1}^{N} P_f(y_i|x_i), \tag{3.1}$$

kjer je N število primerov v zbirki podatkov, y_i je pravilni razred za primer x_i, $P_f(y_i|x_i)$ pa je verjetnost, ki jo je za pravilni razred napovedal klasifikator f. To je mera, ki se je izkazala kot najprimernejša za namen ocenjevanja projekcij in smo jo uporabili v vseh naših eksperimentih. Sorodna mera, kjer imajo napovedane verjetnosti še večji vpliv, je Brierjeva ocena [7]. Pri dveh danih verjetnostnih porazdelitvah, izračunani porazdelitvi razredov p' ter dejanski porazdelitvi razredov p, kjer imamo c razredov, izračunamo Brierjevo oceno kot

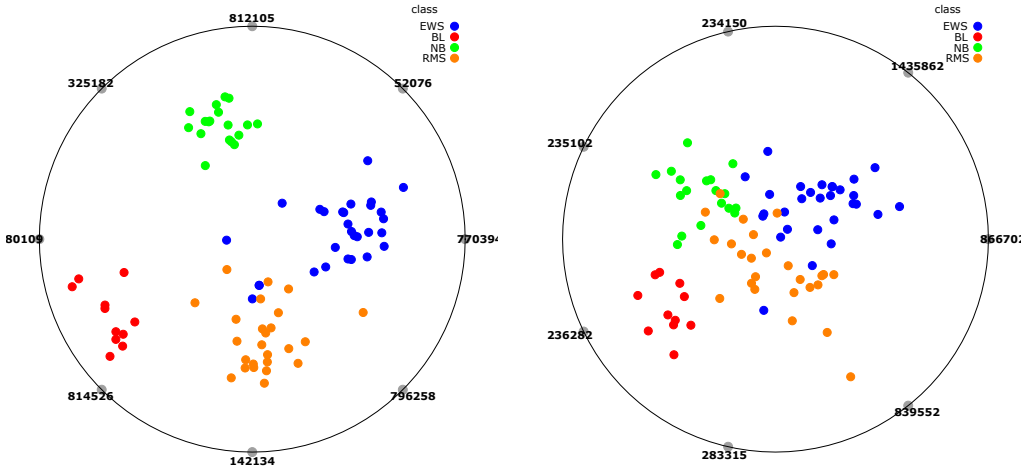
$$b(p;p') = \frac{1}{c}\sum_{i=1}^{c}(p'_i - p_i)^2. \tag{3.2}$$

Večja kot je Brierjeva ocena, slabša je napoved klasifikatorja. Pri izračunu Brierjeve ocene za dani primer določimo dejansko porazdelitev razredov p tako, da ima le-ta vrednost 1 pri pravem razredu in 0 pri ostalih. Brierjevo oceno klasifikatorja na celotni zbirki podatkov nato izračunamo tako, da uporabimo enačbo 3.2 na vsakem primeru in izračunamo povprečno vrednost teh ocen.

Za primerjavo, kako izbira cenilne funkcije vpliva na rangiranje projekcij v praksi, smo naredili eksperiment. Na podatkih SRBCT (za opis glej dodatek A) smo rangirali 10.000 projekcij radviz z uporabo klasifikacijske točnosti, povprečne verjetnosti pravilne klasifikacije $\overline{P}$ ter Brierjeve ocene. Korelacije med različnimi merami so prikazane v tabeli 3.1

	$\overline{P}$	Brierjeva ocena
Klas. točnost	0,938	-0,952
$\overline{P}$		-0,987

Tabela 3.1: Pearsonove korelacije med tremi različnimi cenilnimi funkcijami pri ocenjevanju 10.000 projekcij radviz na podatkih SRBCT. Vse korelacije so statistično pomembne ($p < 0,01$).



(a)
K.T.: 89,16%
$\overline{P}$: 91,71%
Brier: 0,093

(b)
K.T.: 91,57%
$\overline{P}$: 78,58%
Brier: 0,207

Slika 3.2: Različnost ocenjevanja projekcij z različnimi cenilnimi funkcijami. Klasifikacijska točnost (K.T.) je kot boljšo ocenila projekcijo (b), medtem ko je za meri $\overline{P}$ in Brierjevo oceno projekcija (a) bistveno boljša.

in so statistično pomembne ($p < 0,01$). Največje razlike opazimo med klasifikacijsko točnostjo in mero $\overline{P}$, medtem, ko obstaja med $\overline{P}$ in Brierjevo oceno skoraj popolno ujemanje v rangiranju. Po pričakovanju so največje razlike v rangiranju pri projekcijah, kjer veliko primerov leži na robu svoje gruče, blizu primerom iz drugega razreda. Taki primeri imajo nižjo verjetnost pravilne klasifikacije, vendar še vedno dovolj veliko, da jih klasifikator uvrsti v pravilni razred. Brierjeva ocena in mera $\overline{P}$ sta pri ocenjevanju takih projekcij bolj konzervativni in upoštevata to negotovost tako, da zmanjšata oceno projekcije. Ta pojav lepo ilustrirata dve projekciji podatkov SRBCT na sliki 3.2. Na levi projekciji so gruče z vsemi štirimi razredi (z izjemo nekaj osamelcev) lepo ločene med sabo, v desni projekciji pa gruče ležijo zelo blizu ena drugi in se deloma prekrivajo. Kljub bistveno slabši ločenosti gruč je klasifikacijska točnost določila desni projekciji boljšo oceno. Povprečna verjetnost pravilne klasifikacije $\overline{P}$ ter Brierjeva ocena sta manjšo razdaljo med gručami upoštevali in sta zato levi projekciji dodelili bistveno boljšo oceno.

Zaključimo lahko, da je od opisanih mer klasifikacijska točnost najmanj primerna za ocenjevanje zanimivosti projekcij, saj zavrže pomembno informacijo o napovedni negotovosti. Ker sta povprečna verjetnost pravilne klasifikacije $\overline{P}$ ter Brierjeva ocena izredno korelirani, je vseeno, katero od njiju uporabimo. Vseeno bolj priporočamo uporabo mere $\overline{P}$, saj je tako oceno projekcije lažje interpretirati kot oceno, ki jo določi Brierjeva ocena. Mera $\overline{P}$ je tista, ki smo jo uporabili pri ocenjevanju projekcij v vseh naših eksperimentih.

Za izračun končne ocene klasifikatorja (projekcije) smo v vseh poskusih uporabljali prečno preverjanje "izloči enega" (ang. *leave-one-out cross validation*). Algoritem k-NN smo torej testirali na vseh primerih v zbirki podatkov, pri čemer primer, ki smo ga v posameznem koraku želeli klasificirati, ni sodeloval pri napovedi.

3.2 Računska zahtevnost in hevristično preiskovanje prostora projekcij

Za ocenjevanje projekcij smo uporabili enostavno implementacijo algoritma k-NN s časovno zahtevnostjo $O(N^2)$, kjer je N število primerov v zbirki podatkov. Kljub relativno visoki zahtevnosti se je ta implementacija izkazala kot dovolj hitra za potrebe iskanja zanimivih projekcij na izbranih zbirkah podatkov. Za ocenitev 10.000 projekcij na podatkih SRBCT, ki vsebujejo 83 primerov, je naprimer VizRank potreboval 3 minute na računalniku Pentium 4 PC z 2,4 GHz procesorjem. V primeru analize podatkov z bistveno večjim številom primerov (npr. > 2.000) bi bila vseeno primernejša katera od učinkovitejših implementacij algoritma k-NN, kot naprimer že omenjena različica, ki uporablja k-D drevesa.

Računski čas, potreben za ocenitev projekcije, je pomemben, saj je število možnih projekcij pogosto zelo veliko. Vzemimo naprimer metodo radviz, ki je primerna za vizualizacijo poljubnega števila atributov. Recimo, da bi radi ocenili vse projekcije z m atributi pri podatkih, ki vsebujejo n atributov. Možnih izborov m atributov je $\binom{n}{m}$, z vsakim izborom pa lahko s preurejanjem vrstnega reda atributov generiramo $(m-1)!/2$ različnih projekcij. Glede na to, da število možnih projekcij narašča eksponentno s številom vizualiziranih atributov, se je pogosto smiselno omejiti zgolj na ocenjevanje projekcij z majhnim številom atributov. Naše izkušnje kažejo, da to ne predstavlja resne omejitve glede uporabnosti, saj so projekcije z večjim številom atributov (npr. > 10) težko interpretabilne.

Primer, kako število možnih projekcij pri metodi radviz hitro narašča v odvisnosti od števila atributov v podatkih n ter števila vizualiziranih atributov m, je prikazan v tabeli 3.2. Iz tabele je hitro razvidno, da število možnih projekcij že pri zmerno velikih zbirkah podatkov postane ogromno, zaradi česar izčrpno preiskovanje celotnega prostora možnih projekcij ne pride v poštev. Izkaže se, da lahko VizRank celo v primeru obvladljivega števila možnih projekcij več ur ocenjuje projekcije, preden naleti na kakšno zanimivo. Da bi se izognili omenjenima težavama, smo razvili dve hevristiki, ki določata vrstni red, v katerem VizRank ocenjuje projekcije. Namesto naključnega preiskovanja

Število atributov	Število vizualiziranih atributov				
	3	5	7	10	15
10	120	3.024	43.200	181.440	0
20	1.140	$1,8 \cdot 10^5$	$2,7 \cdot 10^7$	$3,3 \cdot 10^{10}$	$6,7 \cdot 10^{14}$
50	19.600	$2,5 \cdot 10^7$	$3,5 \cdot 10^7$	$1,8 \cdot 10^{15}$	$9,8 \cdot 10^{22}$
1000	$1,7 \cdot 10^8$	$9,9 \cdot 10^{13}$	$7,0 \cdot 10^{19}$	$4,8 \cdot 10^{28}$	$3,0 \cdot 10^{43}$

Tabela 3.2: Demonstracija hitrega naraščanja števila možnih projekcij radviz v odvisnosti od števila atributov v podatkih n ter števila vizualiziranih atributov m.

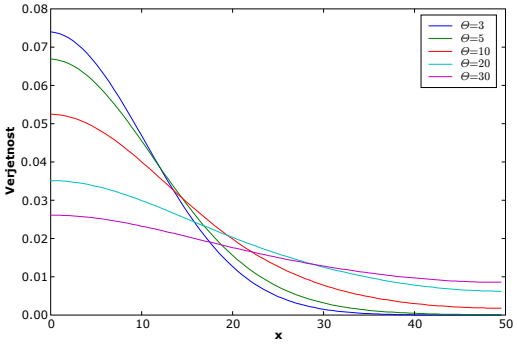
prostora možnih projekcij hevristiki omogočata, da najprej preiščemo tiste dele prostora, ki bolj verjetno vsebujejo zanimive projekcije. Z uporabo teh hevristik postane velikost prostora možnih projekcij relativno nepomembna, saj se z njuno uporabo izognemo ocenjevanju velike večine projekcij in kljub temu z veliko verjetnostjo najdemo najzanimivejše.

V nadaljevanju sledi podrobnejši opis obeh hevristik. Prva je splošna hevristika, ki je namenjena vsem točkovnim vizualizacijskim metodam, druga pa je namenjena še dodatni izboljšavi v primeru metode radviz.

3.2.1 Splošna hevristika

Ocenjevanje projekcij v naključnem vrstnem redu lahko smatramo kot neinformirano, saj ne upošteva nobenega znanja, ki ga imamo ali pa lahko pridobimo v zvezi z atributi v podatkih. V primeru klasificiranih podatkov je naprimer pomembna informacija, ki jo lahko upoštevamo, kako informativen je posamezen atribut glede ločevanja med različnimi razredi. Bolj kot je atribut informativen, bolj lahko pričakujemo, da bo v kombinaciji z drugimi atributi generiral zanimiv prikaz. Podobno lahko od neinformativnega atributa pričakujemo, da v projekciji ne bo nič pripomogel k boljši ločenosti razredov, zaradi česar je bolj verjetno, da bo taka projekcija nezanimiva. Hevristika, ki jo lahko definiramo na podlagi takega sklepanja, deluje na naslednji način. Najprej uporabimo eno od mer za ocenjevanje pomembnosti atributov in izračunamo numerično oceno kvalitete za vsak atribut. Vsaki projekciji lahko nato določimo grobo oceno njene zanimivosti kot vsoto ocen atributov, ki v njej nastopajo. Višja kot je ocena, bolj verjetno je, da je projekcija zanimiva. VizRank lahko te grobe ocene uporabi in projekcije ocenjuje v vrstnem redu od najbolje do najslabše ocenjenih.

Za tako definirano hevristiko lahko trdimo, da je deterministična, saj je vrstni red, v katerem bo VizRank ocenjeval projekcije, natančno določen glede na izračunane ocene atributov. V primerih z velikim številom atributov, kjer vemo, da bomo zmožni preiskati le majhen delček celotnega prostora projekcij, ta deterističnost ni zaželena. Povzroči namreč, da so projekcije, ki jih z uporabo hevristike ocenimo, med sabo zelo korelirane – v njih je prisotno le majhno število različnih atributov. Oglejmo si enostaven primer. Vzemimo, da bi radi poiskali zanimive projekcije radviz s šestimi atributi pri podatkih SRBCT, ki vsebujejo 2.308 atributov. Ko bi z uporabo hevristike izbrali 10.000 različnih



Slika 3.3: Verjetnostna porazdelitev gama pri $k = 1$ in različnih vrednostih parametra Θ.

šesteric atributov (kar pomeni, da bi VizRank moral zaradi 60 različnih možnih postavitev šestih atributov oceniti 600.000 različnih projekcij), bi pri tem uporabili zgolj 15 različnih atributov. Število različnih atributov, ki se pojavijo v teh 600.000 projekcijah, je torej zelo nizko, zaradi česar so si različne projekcije med sabo zelo podobne.

Da bi se izognili tej podobnosti med projekcijami, lahko v postopek izbiranja atributov, ki bodo prisotni v projekciji, vpeljemo naključnost. To lahko storimo na naslednji način. Atribute najprej ocenimo z izbrano mero za ocenjevanje informativnosti atributov in jih uredimo glede na to oceno od najbolj do najmanj informativnega. Podmnožico atributov za vizualizacijo nato določimo tako, da atribute izberemo naključno glede na določeno verjetnostno porazdelitev. Če bi uporabili uniformno verjetnostno porazdelitev bi na ta način dosegli naključno preiskovanje prostora projekcij. Ker je naš namen dati prednost izbire bolje ocenjenim atributom je ena od verjetnostnih porazdelitev, ki so primerne za naš namen, porazdelitev gama. Definirana je kot

$$f(x; k, \Theta) = x^{k-1} \frac{e^{-x/\Theta}}{\Theta^k \Gamma(k)} \qquad \text{pri } x > 0,$$

kjer je Γ funkcija gama. Parameter k predstavlja obliko porazdelitve, Θ pa sploščenost porazdelitve. Verjetnostna porazdelitev pri $k = 1$ in različnih vrednostih parametra Θ je prikazana na sliki 3.3. Če za parameter Θ izberemo primerno vrednost, lahko porazdelitev uspešno uporabimo pri izbiranju podmnožice atributov za vizualizacijo. Čeprav bodo imeli z uporabo te porazdelitve bolje ocenjeni atributi večjo verjetnost izbora, bodo kljub temu pogosto izbrani tudi slabše ocenjeni atributi. Na ta način poskrbimo za večjo raznolikost vizualiziranih atributov, še vedno pa je večji poudarek na tistih delih prostora možnih projekcij, ki bolj verjetno vsebujejo zanimive projekcije.

Za ocenjevanje informativnosti posameznih atributov je mogoče uporabiti poljubno mero. Mere, ki se v strojnem učenju pogosto uporabljajo, so ReliefF, Gini index, informacijski dobitek ter Quinlanov relativni dobitek. Za namen iskanja zanimivih projekcij smo kot najprimernejši ocenili meri ReliefF [66] ter kvocient signala proti šumu (ang. *Signal-to-noise ratio, S2N*) [41]. Kvocient signala proti šumu je definiran kot

$$S2N(a_m) = \sum_{i=1}^{c} \sum_{j=i+1}^{c} \frac{|\mu_i(a_m) - \mu_j(a_m)|}{\sigma_i(a_m) + \sigma_j(a_m)},$$

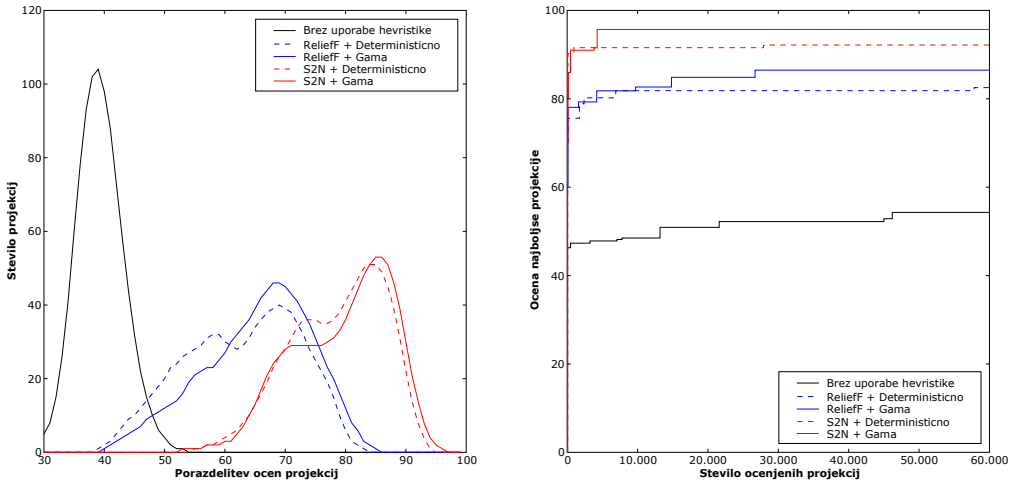
kjer je μ_k srednja vrednost primerov, ki pripadajo k-temu razredu pri atributu a_m, σ_k pa standardna deviacija vrednosti teh primerov.

Ocena uspešnosti hevristike

Uspešnost opisane hevristike bomo demonstrirali na podatkih SRBCT*. Kot vizualizacijsko metodo smo izbrali metodo radviz, saj bi bil v primeru razsevnega diagrama prostor možnih projekcij dovolj obvladljiv, da vpliv hevristike ne bi bil tako izrazit. Z VizRankom smo tako ocenjevali projekcije radviz s šestimi atributi, pri čemer smo kot cenilno funkcijo uporabili povprečno verjetnost pravilne klasifikacije $\overline{P}$. Zanimalo nas je, kako dobre projekcije lahko najdemo, če hevristike ne uporabimo (prostor projekcij preiskujemo naključno), če uporabimo deterministično hevristiko ter če uporabimo hevristiko, ki uporablja porazdelitev gama. Za ocenjevanje atributov smo uporabili meri ReliefF ter S2N. Pri vsakem naboru parametrov smo ocenili 1.000 različnih izborov šestih atributov, oziroma 60.000 različnih projekcij radviz (z uporabo vsakega izbora šestih atributov je mogoče generirati 60 različnih projekcij).

Rezultati primerjave so prikazani na sliki 3.4. Slika 3.4.a prikazuje porazdelitev dobljenih ocen projekcij, kjer smo za vsakega od izborov atributov upoštevali zgolj projekcijo, ki je dosegla najvišjo oceno. Slika 3.4.b pa prikazuje, kako z naraščanjem števila ocenjenih projekcij narašča ocena najboljše najdene projekcije. Iz obeh slik je lepo razvidno, da je hevristika močno potrebna, če želimo najti zanimive projekcije. Brez uporabe hevristike je povprečna ocena projekcij zelo nizka, ocena najboljše najdene projekcije pa je še vedno krepko nižja od prvih ocenjenih projekcij, najdenih z uporabo hevristike. Če primerjamo med sabo uporabljeni meri za ocenjevanje atributov, lahko razberemo, da je mera S2N bistveno uspešneje ocenila kvaliteto atributov, kot pa ReliefF. Prednost mere S2N je prisotna tako pri porazdelitvi ocen projekcij, kot tudi pri najboljši najdeni projekciji. Najverjetnejši razlog, da se je mera ReliefF slabo odrezala je, da imajo uporabljeni podatki zelo veliko število atributov. ReliefF namreč ni kratkovidna mera, ampak pri ocenjevanju upošteva tudi vrednosti drugih atributov. Glede na to, da je v teh zelo visokodimenzionalnih podatkih veliko nekoristnih atributov, ki predstavljajo šum, so razdalje med različnimi primeri, ki jih ReliefF upošteva pri ocenjevanju atributov, zelo neinformativne in neprimerne za uporabo [90]. Iz slik je opazna tudi razlika med determinističnim izbiranjem atributov ter izbiranjem z uporabo porazdelitve gama. Obe različici imata podobno porazdelitev ocen projekcij, le da je deterministična različica pomaknjena za nekaj odstotkov v levo. Na desnem grafu je tudi vidno, da ne glede na izbrano mero za ocenjevanje atributov različica, ki uporablja verjetnostno izbiranje atributov, uspe najti signifikantno boljše projekcije.

*Rezultati eksperimenta na dodatnih petih zbirkah podatkov so opisani v dodatku B. Uspešnost hevristike na teh zbirkah je zelo podobna uspešnosti hevristike na podatkih SRBCT.



(a) (b)

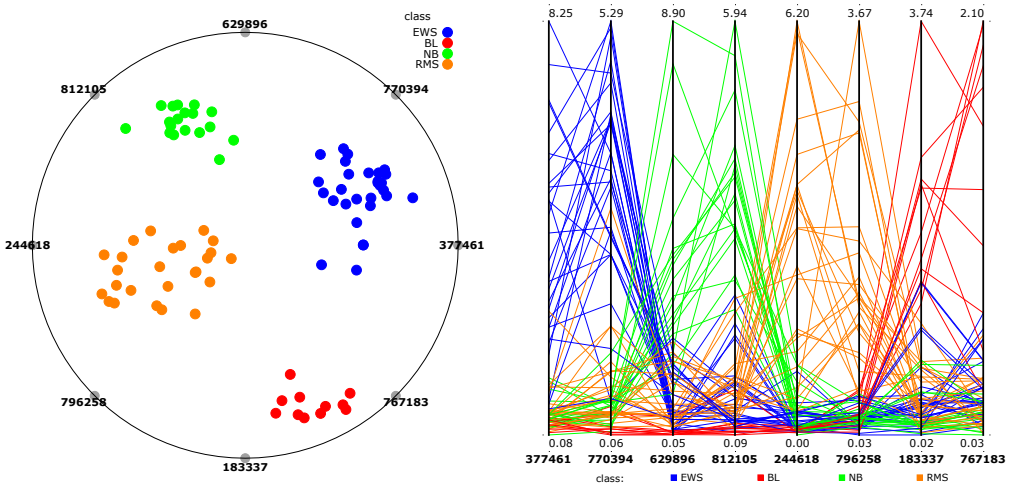
Slika 3.4: Uspešnost splošne hevristike pri izbiri 1.000 šesteric atributov. (a) Prikaz porazdelitve ocen projekcij glede na različne parametre hevristike. (b) Prikaz, kako z naraščanjem števila ocenjenih projekcij narašča ocena najboljše trenutno najdene projekcije.

Glede na dobljene rezultate lahko sklepamo, da je hevristika zelo uspešen in nujen člen pri iskanju zanimivih prikazov. Za ocenjevanje atributov predlagamo uporabo kvocienta signala proti šumu, sam izbor atributov pa je uspešnejši, če uporabimo verjetnostno porazdelitev gama.

3.2.2 Posebna hevristika za metodo radviz

Pravkar opisana hevristika je primerna za uporabo pri vseh točkovnih vizualizacijskih metodah in temelji na tem, da lahko z bolje ocenjenimi atributi bolj verjetno generiramo zanimive projekcije, kot pa s slabše ocenjenimi atributi. V primeru, da nas zanimajo zgolj projekcije, generirane z metodo radviz, pa lahko proces iskanja zanimivih projekcij še bistveno pohitrimo. V nadaljevanju bomo predstavili hevristiko, primerno samo za metodo radviz, s katero lahko izboljšamo izbiranje atributov ter hkrati zmanjšamo število različnih postavitev atributov, ki jih je potrebno oceniti.

Vzemimo primer zelo zanimive projekcije radviz z osmimi atributi, ki jo je VizRank z uporabo splošne hevristike našel pri podatkih SRBCT. Projekcija je prikazana na sliki 3.5.a, slika 3.5.b pa prikazuje istih osem atributov v istem vrstnem redu v prikazu s paralelnimi koordinatami. Iz prikaza s paralelnimi koordinatami je nemudoma vidna zelo pomembna regularnost v zvezi z vrednostmi primerov, ki pripadajo posameznim razredom. Pri prvih dveh atributih imajo primeri iz razreda *EWS* visoke vrednosti, medtem ko imajo primeri iz ostalih razredov nizke vrednosti. Pri drugih dveh atributih velja ista zakonitost za razred *NB*, pri tretjih dveh za razred *RMS*, pri zadnjih dveh atributih pa za razred *BL*. Ta zakonitost je pravzaprav eden glavnih razlogov, zaradi katerih so različni



(a) (b)

Slika 3.5: (a) Primer odlične projekcije radviz pri podatkih SRBCT. (b) Prikaz istih atributov s paralelnimi koordinatami.

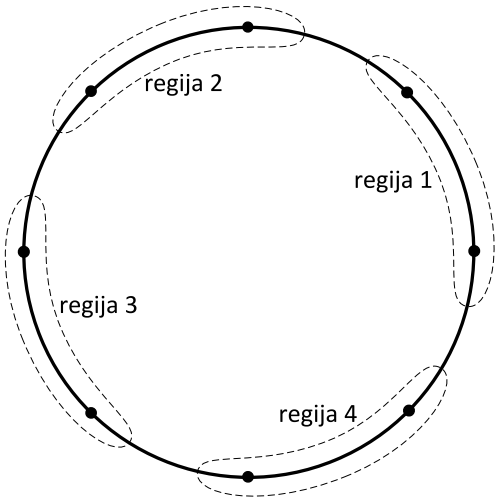
razredi v projekciji tako lepo ločeni. Vsak od omenjenih parov atributov namreč privlači k sebi primere iz enega razreda, s čimer se doseže dobra medsebojna ločenost posameznih razredov.

Dobro ločenost razredov v projekcijah radviz torej najenostavneje dosežemo tako, da izpolnimo dva pogoja. Prvi je, da atribute izberemo tako, da je v izboru za vsakega od razredov prisoten vsaj en atribut, pri katerem imajo primeri iz tega razreda v povprečju večje vrednosti kot pri ostalih razredih. S tem dosežemo, da za vsak razred obstaja atribut, ki bo v projekciji k sebi močneje privlačil primere tega razreda, kot pa primere ostalih razredov (temu bomo odslej rekli, da atribut glasuje za razred). Drugi pogoj pa je, da te atribute na krožnici smiselno uredimo. Če imamo naprimer dva atributa, ki glasujeta za isti razred, potem ju je na krožnici smiselno postaviti kot sosednja, kajti če ju postavimo nasproti, se bo njuna sposobnost ločevanja enega razreda od ostalih izničila.

Glede na omenjena pogoja je mogoče sestaviti posebno hevristiko, ki jo bomo imenovali *S2N-One*. Prvi del hevristike vključuje izbiranje atributov, ki bodo nastopili v projekciji. Kot rečeno je pri tem pomembno, da atribute izberemo tako, da je število atributov, ki glasujejo za posamezen razred, čim bolj uravnoteženo. Tak izbor lahko dosežemo z naslednjim postopkom. Za vsak razred k najprej ocenimo vsakega od atributov a_m z modificirano različico kvocienta signala proti šumu

$$S2N\text{-}One_k(a_m) = \sum_{i \neq k} \frac{\mu_k(a_m) - \mu_i(a_m)}{\sigma_k(a_m) + \sigma_i(a_m)}.$$

S to mero lahko za posamezen razred k izračunamo, kako močno vsak od atributov glasuje za ta razred – večja kot je izračunana vrednost, močneje atribut glasuje za dani razred. Pri izračunu upoštevamo predznak razlike $\mu_k - \mu_i$, zaradi česar nam pozitivni



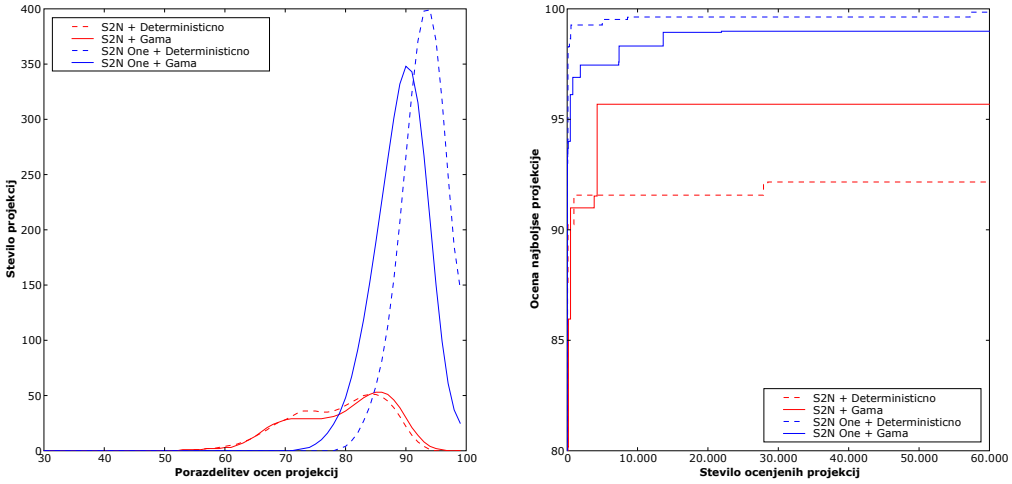
Slika 3.6: Razdelitev delov krožnice na regije. Vsakemu razredu pripada ena regija, znotraj katere je smiselno postaviti atribute, ki glasujejo za isti razred.

predznak ocene $v_k(a_m)$ pove, da atribut a_m glasuje za razred k, negativni predznak pa da glasuje proti njemu (oziroma, da glasuje za enega od drugih razredov). Za vsako vrednost razreda k lahko sestavimo seznam L_k, v katerem atribute uredimo glede na padajoče vrednosti ocen. Kot pri splošni hevristiki nato atribute izbiramo tako, da imajo bolje ocenjeni atributi večjo verjetnost izbire. Za razliko od splošne hevristike, pa v tem primeru nimamo enega, ampak c seznamov rangiranih atributov. Pri izbiranju m atributov zato iz vsakega seznama L_k izberemo približno m/c atributov. Če želimo naprimer izbrati osem atributov pri podatkih s štirimi razredi, potem za vsak razred izberemo po dva atributa, ki glasujeta zanj. V primeru da m/c ni celo število, izberemo atribute tako, da je število atributov, izbranih iz posameznega seznama, čim bolj uravnoteženo. Pri izbiranju atributov iz posameznih seznamov L_k lahko, kot pri splošni hevristiki, uporabimo deterministično izbiranje ali pa verjetnostno izbiranje z uporabo porazdelitve gama.

Drugi del hevristike *S2N-One* se nanaša na postavitev izbranih atributov po krožnici. Ker je atribute, ki glasujejo za isti razred, smiselno postaviti kot sosednje, lahko z uporabo tega znanja bistveno zmanjšamo število različnih permutacij atributov, ki jih je potrebno oceniti. Posamezna mesta, kjer so pritrjena dimenzijska sidra, lahko združimo v c različnih regij, kot je to prikazano na sliki 3.6. Vsakemu od razredov pripada ena regija, znotraj katere so vsi atributi, ki glasujejo zanj. Različne permutacije atributov, ki jih mora VizRank oceniti, dobimo tako, da preizkusimo različna prirejanja razredov posameznim regijam ter različne postavitve atributov znotraj posameznih regij. Število različnih permutacij (projekcij), ki jih je potrebno oceniti, če atribute razporejamo na ta način, je prikazano v tabeli 3.3. Številke v oklepajih predstavljajo število permutacij, ki jih je potrebno oceniti, če tega znanja ne upoštevamo. Iz primerjave je hitro razvidno, da hevristika signifikantno pripomore k hitrejšemu iskanju zanimivih projekcij, saj je z njeno uporabo v nekaterih primerih potrebno oceniti manj kot eno stotino vseh permutacij, ki

Število razredov	Število vizualiziranih atributov				
	5	6	7	8	9
2	6 (12)	18 (60)	72 (360)	288 (2.520)	1440 (20.160)
3	4 (12)	8 (60)	24 (360)	72 (2.520)	216 (20.160)
4	6 (12)	12 (60)	24 (360)	48 (2.520)	144 (20.160)
5	12 (12)	24 (60)	48 (360)	96 (2.520)	192 (20.160)

Tabela 3.3: Število različnih permutacij, ki jih je potrebno oceniti, če upoštevamo, da so zanimive zgolj tiste permutacije, v katerih so atributi, ki glasujejo za isti razred, znotraj iste regije. Številke v oklepaju predstavljajo število permutacij, ki jih je potrebno oceniti, če tega ne upoštevamo.



(a) (b)

Slika 3.7: Primerjava med splošno hevristiko in posebno hevristiko za metodo radviz. (a) Prikaz porazdelitve ocen projekcij glede na različne parametre hevristike. (b) Prikaz, kako z naraščanjem števila ocenjenih projekcij narašča ocena najboljše najdene projekcije.

bi jih morali sicer oceniti.

Ocena uspešnosti hevristike

Da bi ocenili uspešnost hevristike *S2N-One*, smo ponovno uporabili podatke SRBCT ter iskali zanimive projekcije radviz z uporabo šestih atributov[†]. Cenilna funkcija, ki smo jo uporabili za ocenjevanje projekcij, je bila povprečna verjetnost pravilne klasifikacije. Poleg uspešnosti hevristike nas je zanimalo tudi, kakšen vpliv imata deterministično ter verjetnostno izbiranje atributov na uspešnost iskanja zanimivih projekcij. Uspešnost hevristike *S2N-One* smo primerjali z uspešnostjo splošne hevristike.

[†]Rezultati eksperimenta na dodatnih petih zbirkah podatkov so opisani v dodatku B in so zelo podobni rezultatom, dobljenim na podatkih SRBCT.

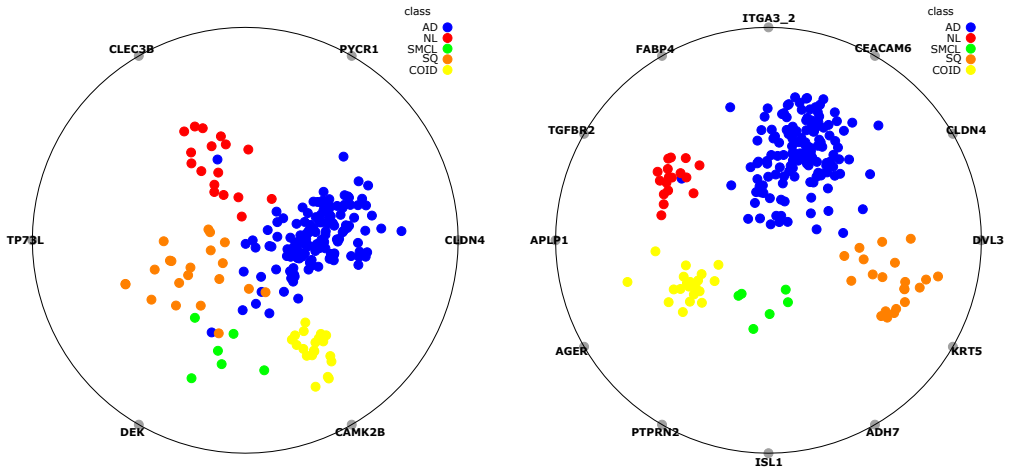
Rezultati primerjave so prikazani na sliki 3.7. Iz obeh grafov je razvidno, da je hevristika *S2N-One* izrazito boljša od splošne hevristike. Uspešnejša je tako pri hitrejšem iskanju zanimivih projekcij kot tudi glede porazdelitve ocen ocenjenih projekcij. Bistvena prednost hevristike *S2N-One* je, da z njo ocenjujemo samo podmnožico vseh možnih permutacij atributov, zaradi česar pri enakem številu ocenjenih projekcij dejansko ocenimo večje število izborov atributov – na sliki 3.7.a je to razvidno iz razlike med ploščinama pod krivuljama obeh hevristik. Iz rezultatov je razvidno tudi, da se pri hevristiki *S2N-One* deterministično izbiranje atributov izkaže bolje kot verjetnostno. Razlog za to je v tem, da z modificirano različico kvocienta signala proti šumu zelo uspešno ocenimo primernost atributov za generiranje zanimivih projekcij radviz. Če iz vseh urejenih seznamov L_k za vsak razred deterministično izbiramo atribute, bomo z večjo verjetnostjo generirali zanimivo projekcijo, kot če to storimo glede na izbrano verjetnostno porazdelitev. Kljub temu, da z determinističnim izbiranjem atributov v povprečju odkrijemo boljše projekcije, pa še vedno velja, da so te projekcije med sabo bolj korelirane, kot če atribute izbiramo verjetnostno.

3.2.3 Lokalna optimizacija projekcij

Preiskovanje prostora možnih projekcij, ki ga dosežemo z uporabo splošne ter posebne hevristike, lahko smatramo za usmerjeno preiskovanje, ki pa prostora ne preiskuje lokalno. Vsak naslednji izbor atributov je neodvisen od zanimivosti prejšnjega, zaradi česar se nezvezno premikamo po različnih delih prostora.

Poleg takega načina iskanja zanimivih projekcij je mogoče uporabiti tudi postopek lokalne optimizacije. Primeren je za vse točkovne metode, tukaj pa ga bomo ponovno opisali na primeru metode radviz. Vzemimo, da imamo neko projekcijo, v kateri bi radi dodatno izboljšali ločenost med posameznimi razredi. Vsakega od atributov, ki trenutno nastopajo v projekciji, lahko poskusimo nadomestiti z enim od preostalih atributov iz zbirke podatkov ter ocenimo zanimivost nove projekcije. V primeru, da je ocena nove projekcija boljša kot ocena stare projekcije, ponovimo postopek optimizacije, pri čemer poskušamo tokrat dodatno izboljšati novo projekcijo. Kadar ima zbirka podatkov preveliko število atributov, da bi preizkušali vsakega od atributov v projekciji zamenjati z vsakim od njih, poskušamo projekcijo izboljšati z uporabo zgolj manjše podmnožice najbolje ocenjenih atributov. Ko naletimo na projekcijo, ki je ni mogoče dodatno izboljšati z zamenjavo enega od atributov, lahko pri metodah kot je radviz, kjer lahko vizualiziramo poljubno število atributov, poskusimo tudi s povečanjem števila vizualiziranih atributov. V trenutno projekcijo poskusimo med posamezne sosednje atribute vstaviti dodaten atribut in s tem še dodatno izboljšati oceno projekcije. Postopek dodajanja novih atributov lahko ponavljamo, dokler je mogoče doseči izboljšanje projekcije ter dokler ne postane projekcija zaradi prevelikega števila prikazanih atributov slabo interpretabilna.

Čeprav je opisani postopek lokalne optimizacije mogoče uporabiti za optimizacijo naključnih projekcij, se je v naših poskusih najuspešneje izkazala takrat, ko smo jo uporabili za dodatno izboljšanje najzanimivejših projekcij najdenih z uporabo splošne ali



(a) 89,98% (b) 97,41%

Slika 3.8: Primer uporabe lokalne optimizacije. Iz začetne projekcije (a) je VizRank z zamenjavo in dodajanjem atributov našel projekcijo z višjo oceno (b).

posebne hevristike. Kot posebej koristna se je lokalna optimizacija izkazala v primerih, ko imamo večje število razredov, med katerimi lahko uspešno ločimo samo z uporabo večjega števila atributov. V takih primerih je smiselno z uporabo običajnega preiskovanja poiskati čim boljšo začetno projekcijo z manjšim številom atributov ter jo nato z lokalno optimizacijo dodatno izboljšati z dodajanjem novih atributov. Primer tako dobljene projekcije je na sliki 3.8.b. Le-ta vsebuje projekcijo radviz z dvanajstimi atributi pri podatkih o pljučnem raku (glej dodatek A) in je bila dobljena z lokalno optimizacijo projekcije s šestimi atributi (slika 3.8.a), najdene z uporabo posebne hevristike.

3.3 Empirična ocena primernosti metode VizRank

Glavni namen metode VizRank je, da omogoča avtomatsko ocenjevanje zanimivosti projekcij in s tem človeku olajša analizo podatkov. Da bi bil ta cilj izpolnjen, pa je bistvenega pomena, da je ujemanje med rangiranjem projekcij, ki ga določi VizRank, čim bolj podobno rangiranju, ki bi ga določil človek. Le v tem primeru namreč človeku ne bo potrebno zanimivih projekcij iskati ročno, ampak bo lahko zaupal ocenam projekcij, ki jih bo določil VizRank.

Da bi preverili, kakšno je dejansko ujemanje med človekovim in VizRankovim rangiranjem projekcij ter ugotovili pri katerih parametrih algoritma je to ujemanje največje, smo izvedli naslednji eksperiment. Izbrali smo pet zbirk podatkov dobljenih z uporabo mikromrež – SRBCT, MLL, DLBCL, levkemija ter pljučni rak (glej dodatek A). Zbirke podatkov so bile izbrane tako, da smo dosegli čim večjo variabilnost v številu razredov ter številu primerov – te zbirke imajo od dva do pet razredov ter od 72 do 203 primerov. Na vsaki od zbirk smo z uporabo posebne hevristike ocenili 30.000 projekcij radviz, ki so

Učni algoritem	Klasifikacijska točnost	$\overline{P}$	Brierjeva ocena
k-NN ($k = N/c$)	0,711	0,782	-0,746
k-NN ($k = \sqrt{N}$)	0,587	0,642	-0,597
SVM	0,677	0,586	-0,677
Odločitvena drevesa	0,252	0,278	-0,210

Tabela 3.4: Korelacije med človeškim in VizRankovim primerjanjem parov projekcij pri uporabi različnih učnih algoritmov in cenilnih funkcij. Vse korelacije so statistično pomembne ($p < 0,01$).

vsebovale od tri do osem atributov. Projekcije smo ocenili z uporabo različnih cenilnih funkcij – klasifikacijsko točnostjo, povprečno verjetnostjo pravilne klasifikacije ter Brierjevo oceno. Ker nas je zanimala tudi primernost različnih učnih algoritmov, smo poleg metode k-najbližjih sosedov uporabili še metodo podpornih vektorjev (SVM) z uporabo RBF jeder (s standardnimi parametri $\gamma = 0,05$, $C = 1,0$, $p = 0,5$ ter $\epsilon = 0,001$) ter odločitvena drevesa, pri katerih smo globino dreves omejili na maksimalno štiri nivoje. Pri metodi k-NN smo projekcije ocenili z uporabo dveh formul za določanje parametra k: $k = \sqrt{N}$ ter $k = N/c$.

Pri vsaki zbirki podatkov smo za vsak par učnih algoritmov (a, b) izmed 1.000 najbolje ocenjenih projekcij izbrali 50 takih parov projekcij (p_1, p_2), za katere velja, da je vrednost izraza $|(o_a(p_1) - o_a(p_2)) - (o_b(p_1) - o_b(p_2))|$ čim večja, pri čemer je $o_k(p_i)$ ocena zanimivosti projekcije p_i ob uporabi algoritma k. Izbrali smo torej tiste pare projekcij, pri katerih je bila razlika v ocenah obeh projekcij ob uporabi različnih algoritmov čim večja. Če želimo ugotoviti, kateri učni algoritem rangira projekcije najbolj podobno kot človek, je namreč smiselno uporabiti tiste pare projekcij, pri katerih je neskladje med algoritmi največje; iz projekcij, na katerih se različni algoritmi strinjajo glede ocen, je nemogoče sklepati o tem, kateri algoritem je *bolj* primeren.

Na internetu smo nato postavili spletno stran, kjer se je za vsakega obiskovalca iz celotne množice izbranih parov projekcij prikazalo 20 naključno določenih parov projekcij. Za vsak par projekcij je moral obiskovalec na devet-stopenjski lestvici določiti, če je ena od projekcij bolj zanimiva kot druga in če to je, koliko bolj zanimiva je. Za ocenjevanje zanimivosti projekcij nismo postavili nobenih časovnih omejitev.

V eksperimentu je sodelovalo 30 ljudi iz poklicnega področja računalništva in kognitivne psihologije, kar pomeni, da je bilo skupno ocenjenih 600 parov projekcij. Ujemanje med VizRankovim in človeškim rangiranjem smo določili tako, da smo izračunali korelacijo med človeškim rangiranjem parov projekcij (p_1, p_2) ter razlikami $o_k(p_1) - o_k(p_2)$ za vsak učni algoritem k ter uporabljeno cenilno funkcijo. Za ocenjevanje korelacije smo uporabili Spearmanov korelacijski koeficient, ki je v primerjavi s Pearsonovim koeficientom primeren za uporabo tudi v primeru ordinalnih spremenljivk.

Korelacije med človekovim rangiranjem ter VizRankovimi rangiranji pri različnih učnih algoritmih in cenilnih funkcijah so prikazane v tabeli 3.4. Med tremi cenilnimi

funkcijami se je kot najprimernejša izkazala povprečna verjetnost pravilne klasifikacije $\overline{P}$. Poleg tega je iz rezultatov razvidno, da je najprimernejši učni algoritem za rangiranje projekcij ravno metoda k-najbližjih sosedov, pri kateri število sosedov določimo po enačbi $k = N/c$. Če parametru k določimo manjšo vrednost (naprimer po enačbi $k = \sqrt{N}$), postane ujemanje manjše, kar najverjetneje kaže na to, da ljudje raje vidimo vse primere enega razreda v eni večji gruči, kot pa v več manjših, medsebojno ločenih gručah. Poleg tega so pri uporabi večje vrednosti parametra k kot boljše ocenjene tiste projekcije, v katerih je razdalja med gručami različnih razredov večja, kar je vsekakor vizualno zelo zaželena lastnost. Malenkost manj kot metoda k-NN je primerna metoda podpornih vektorjev, še bistveno manj pa je primerna uporaba odločitvenih dreves.

Na podlagi rezultatov opisanega eksperimenta lahko tudi empirično potrdimo uspešnost metode VizRank. Visoke korelacije v tabeli 3.4 kažejo, da VizRank, ob pravilno izbranem učnem algoritmu in cenilni funkciji, ocenjuje projekcije zelo podobno, kot bi jih ocenjeval človek. Oba omenjena parametra imata pri ocenjevanju pomembno vlogo in odločilno vplivata na uspešnost postopka. Eksperiment je še dodatno potrdil teoretična sklepanja, da je kot učni algoritem najprimernejša metoda k-najbližjih sosedov (z veliko vrednostjo parametra k), kot cenilna funkcija pa povprečna verjetnost pravilne klasifikacije $\overline{P}$.

3.4 Uporaba seznama ocenjenih projekcij

Sposobnost VizRanka je, da za vsako izbrano projekcijo oceni njeno potencialno zanimivost. Glede na te ocene je mogoče projekcije urediti od potencialno najbolj do najmanj zanimivih, pri čemer si uporabnik nato običajno ogleda zgolj zelo majhno podmnožico najbolje ocenjenih projekcij. Težave nastanejo pri analizi visokodimenzionalnih podatkov, kjer se pogosto zgodi, da ima zaradi ogromnega števila možnih projekcij veliko projekcij zelo podobne ocene. Lep zgled tega je bila uporaba posebne hevristike na podatkih SRBCT (glej sliko 3.7.a), kjer je bila razlika v oceni od prve do dvestote najbolje ocenjene projekcije le 2%. Ker je zelo malo verjetno, da si bomo natančno ogledali dvesto ali več projekcij, je koristno, če lahko informacijo, ki jo vsebujejo te projekcije, vseeno nekako povzamemo in predstavimo uporabniku.

V nadaljevanju bomo predstavili tri možne načine, kako lahko seznam najbolje ocenjenih projekcij koristno uporabimo za odkrivanje dodatnega znanja v podatkih. Prikazali bomo, kako je mogoče te projekcije uporabiti za ocenjevanje pomembnosti atributov, odkrivanje interakcij med atributi ter za iskanje osamelcev.

3.4.1 Mera za ocenjevanje pomembnosti atributov

Ocenjevanje pomembnosti atributov je v strojnem učenju ena zelo pomembnih nalog. Za številne postopke strojnega učenja, kot so naprimer gradnja odločitvenih dreves, konstruktivna indukcija ter izbor podmnožice atributov, je namreč ocenjevanje pomembnosti atributov eden ključnih elementov.

Kot smo že omenili, obstaja večje število mer, ki na različne načine ocenjujejo, kako dobro posamezni atributi ločujejo med različnimi razredi. Večina mer je takih, da pri ocenjevanju upoštevajo vsak atribut posebej, neodvisno od vrednosti drugih atributov. Težava pri takem pristopu je, da so lahko nekateri atributi med sabo odvisni in postanejo informativni šele v kombinaciji z drugimi atributi. Enostaven primer take odvisnosti (čemur pravimo tudi interakcija) je problem paritete ($y = x_1 \otimes x_2$), pri katerem je vsak atribut posebej popolnoma neinformativen pri določanju vrednosti razreda y.

Primer nekratkovidne mere, ki lahko uspešno upošteva interakcije med atributi, je mera Relief [63] ter njena kasnejša razširitev ReliefF [64, 67, 90]. Osnovna različica mere ocenjuje atribute z iterativnim ponavljanjem naslednjega postopka. V i-ti iteraciji poišče za naključno izbran primer R_i najbližji primer H iz istega razreda ter najbližji primer M iz drugega razreda. Za vsak atribut a nato popravi njegovo oceno $W(a)$ z uporabo enačbe

$$W(a) := W(a) - \text{diff}(a, R_i, H) + \text{diff}(a, R_i, M),$$

pri čemer je $\text{diff}(a, X, Y)$ razlika v vrednosti med primeroma X in Y pri atributu a. Iz enačbe lahko sklepamo, da je za informativen atribut zaželeno, da je razlika med primeroma iz različnih razredov čim večja, med primeroma iz enakih razredov pa čim manjša. Opisan postopek ponovimo m krat, pri čemer je m uporabniško nastavljiv parameter.

Razlog, zaradi katerega Relief ni kratkovidna mera, je, da so najbližji sosedje H in M določeni glede na vrednosti vseh atributov. Iskanje najbližjih sosedov na osnovi razdalj med primeri v originalnem prostoru se odlično obnese v primerih, ko zbirka podatkov vsebuje manjše število atributov, v primeru večjega števila atributov pa pogosto nastanejo težave. Z večanjem števila atributov namreč postajajo razdalje med posameznimi primeri v prostoru čedalje večje, kar pomeni, da se tudi najbližji primeri medsebojno zelo razlikujejo. Za omenjeni pojav se je uveljavilo ime "prekletstvo dimenzionalnosti" (ang. *curse of dimensionality*) [3] in je glavni razlog, zaradi katerega se metode, ki temeljijo na lokalnosti primerov (k-NN, parzenova okna, Relief), slabo obnesejo pri velikem številu atributov. Primer, ki demonstrira, kako večanje števila atributov vpliva na Reliefovo ocenjevanje atributov, je prikazan v naslednjem razdelku.

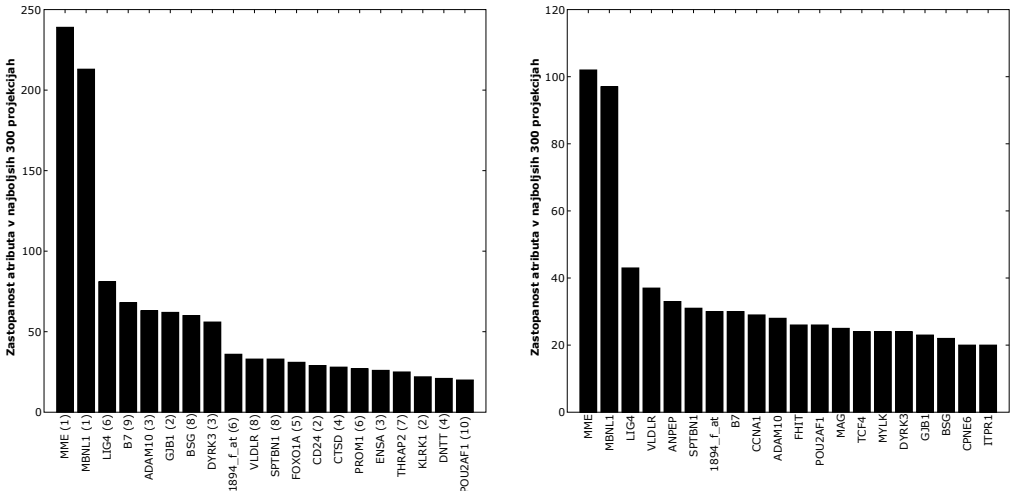
Enostaven način, kako lahko nekratkovidno in uspešno ocenimo pomembnost atributov tudi v primeru velikega števila atributov, je z uporabo seznama ocenjenih projekcij. Pri izbrani podmnožici m najboljših najdenih projekcij enostavno preštejemo kako pogosto je vsak od atributov zastopan v teh projekcijah. Parameter m določi uporabnik in je običajno med 50 in 300, odvisno od tega, koliko najboljših projekcij ima zelo podobne ocene. Pogostost zastopanosti posameznega atributa v izbrani množici projekcij smatramo kot indikator njegove pomembnosti – večje kot je, bolj je atribut pomemben za uspešno ločevanje med razredi. Zastopanost posameznih atributov lahko prikažemo v obliki histograma, ki je prikazan na sliki 3.9.a. Graf prikazuje 20 najpogosteje zastopanih atributov v najboljših 300 projekcijah radviz (od 60.000 ocenjenih projekcij) pri podatkih

MLL. Iz grafa je lepo razvidno, da sta atributa *MME* in *MBNL1* številčno bistveno bolj zastopana kot ostali atributi in se pojavljata v več kot polovici upoštevanih projekcij.

Pri ocenjevanju atributov na podlagi njihove zastopanosti v skupini najboljših projekcij ne uporabljamo nobenega postopka, ki bi dejansko ocenjeval, kako dobro posamezen atribut loči med razredi. Namesto tega se zanašamo na to, da bodo najboljše projekcije tiste, v katerih bo vsak od atributov maksimalno prispeval k dobri ločenosti razredov. Če projekcija vsebuje polovico neuporabnih atributov, bo slabo ocenjena, zaradi česar ne bo upoštevana pri ocenjevanju atributov. Gotovo se seveda zgodi, da se v kakšni odlični projekciji znajde tudi kateri od neuporabnih atributov, zelo malo verjetno pa je, da bo tak atribut zastopan v večjem številu takih projekcij.

Za rangiranje atributov lahko uporabimo različne vizualizacijske metode, vendar se je pri tem potrebno zavedati razlik in omejitev. Če uporabimo razsevne diagrame, potem ocenjujemo atribute na podlagi zelo omejenega konteksta, saj upoštevamo le interakcije med pari atributov. V primeru metode radviz je kontekst večji, saj hkrati vizualiziramo več atributov. Pri tem je seveda potrebno paziti, da so v ocenjenih projekcijah čim bolj enakomerno zastopani vsi atributi, da s tem ne dajemo prevelike prednosti določenim atributom. Smiselno se je tudi omejiti na projekcije z relativno majhnim številom atributov (npr. 3–6), saj dobro ocenjene projekcije z večjim številom atributov pogosteje vsebujejo tudi neuporabne atribute. Rangiranje atributov na sliki 3.9.a je bilo naprimer narejeno z uporabo projekcij radviz z največ petimi atributi. Za ocenjevanje atributov so primerne tudi splošne linearne projekcije, le da pri tem prisotnosti atributov ni smiselno upoštevati binarno, ampak je bolj primerno upoševati dolžino vektorja, ki pripada atributu – atributi s krajšimi vektorji so namreč manj pomembni kot atributi z daljšimi vektorji.

Uporaba opisanega načina rangiranja atributov na osnovi seznama najboljših projekcij je smiselna zgolj v primerih, ko imamo na voljo dovolj atributov (vsaj 10), da je z njimi mogoče generirati večje število projekcij. Če je število možnih projekcij nizko, je namreč dobljena ocena atributov nezanesljiva. Slabost, ki je iz tega pogoja razvidna, je počasnost postopka, zaradi česar je tovrstno ocenjevanje atributov manj primerno pri nalogah, kot je naprimer gradnja odločitvenih dreves, kjer je potrebno ocenjevanje atributov izvesti velikokrat. Težava opisanega postopka ocenjevanja je tudi ta, da je rezultat odvisen od števila upoštevanih projekcij m. Iz naših opravljenih eksperimentov lahko sklepamo, da je pri domenah z večjim številom atributov, pri katerih običajno zaradi podobnih ocen najboljših projekcij upoštevamo sto ali več projekcij, rangiranje atributov stabilno in neobčutljivo na spreminjanje vrednosti parametra m. Ne glede na to lahko vpliv parametra uspešno zmanjšamo tako, da pri ocenjevanju atributov upoštevamo tudi rang projekcije. Posamezne uporabljene projekcije tako niso več enakovredne pri njihovem glasovanju, ampak je njihov glas odvisen od tega, kako visoko so rangirane. Za uteževanje lahko uporabimo poljubno padajočo funkcijo – v naših eksperimentih smo naprimer uporabljali funkcijo e^{-t^2/s^2}, kjer je t rang projekcije, s pa je določen tako, da je utež zadnje uporabljene (m-te) projekcije enaka 0,001.



(a) (b)

Slika 3.9: Dva primera uporabe seznama projekcij za ocenjevanje pomembnosti atributov. Histograma prikazujeta kako pogosto so posamezni atributi prisotni v 300 najbolje ocenjenih projekcijah radviz pri podatkih MLL. Histogram (a) je dobljen iz najboljših projekcij najdenih z uporabo posebne hevristike za metodo radviz. Pri histogramu (b) smo z uporabo mere *S2N-One* najprej izbrali 99 najbolje ocenjenih atributov ter z njimi generirali različne projekcije brez uporabe hevristike (atribute smo izbirali naključno).

Vpliv izbrane hevristike na ocenjevanje pomembnosti atributov

Kot je bilo pokazano v razdelku 3.2 je potrebno pri zbirkah podatkov z večjim številom atributov uporabljati hevristiko, ki nekatere atribute izbere pogosteje kot druge in na ta način hitreje odkrije boljše projekcije. Ob takem "pristranskem" izbiranju atributov se postavlja vprašanje, kako to vpliva na dobljene ocene atributov.

Oglejmo si naprimer histogram atributov za podatke MLL, ki je prikazan na sliki 3.9.a. Narejen je bil na osnovi projekcij, dobljenih z uporabo posebne hevristike za metodo radviz (glej razdelek 3.2.2), ki za vsako vrednost razreda k sestavi seznam L_k, v katerem so atributi rangirani z uporabo mere *S2N-One*. Pri podatkih MLL, kjer imamo tri razrede – *MLL*, *ALL* ter *AML* – so bili torej seznami trije: L_{MLL}, L_{ALL} ter L_{AML}. Pri izbiranju atributov je bilo nato uporabljeno verjetnostno izbiranje z uporabo porazdelitve gama, tako da so bili najpogosteje izbrani atributi iz začetka seznamov L_k. V histogramu 3.9.a so mesta (indeksi), kjer se posamezni atributi nahajajo v teh seznamih, predstavljeni z vrednostmi v oklepajih. Takoj lahko opazimo, da sta oba najbolje ocenjena atributa na prvem mestu v enem od seznamov – atribut *MME* je prvi v seznamu L_{ALL}, atribut *MBNL1* pa prvi v seznamu L_{MLL}. Glede na to, da zasedajo tudi ostali prikazani atributi visoka mesta v seznamih L_k, bi bil lahko eden od možnih sklepov ta, da predstavljena metoda za ocenjevanje pomembnosti atributov močno korelira z mero, ki jo pri hevristiki uporabimo za rangiranje atributov. Če bi se to izkazalo kot resnično, bi bilo predstavljeno mero nesmiselno uporabljati, saj bi bilo mogoče zelo podobno rangiranje atributov doseči

direktno z uporabo mere.

Da bi stvar natančneje preučili, smo izvedli naslednji eksperiment. Sestavili smo novo zbirko podatkov, ki je vsebovala zgolj prvih (najboljših) 33 atributov iz vsakega seznama L_k, torej skupno 99 atributov. Z VizRankom smo nato ocenili različne projekcije radviz z največ petimi atributi, pri čemer smo atribute izbirali brez uporabe hevristike – vsak atribut je torej imel enako verjetnost izbora. Da bi bili dobljeni rezultati kljub večjemu številu atributov zanesljivi, smo ocenili kar 500.000 različnih projekcij. Dobljeni histogram atributov za najboljših 300 najdenih projekcij je prikazan na sliki 3.9.b. Iz histograma je jasno razvidno, da atributa *MME* ter *MBNL1* ponovno močno odstopata v številu pojavitev. Glede na to, da obstaja tudi v splošnem dobro ujemanje med skupinama atributov na slikah 3.9.a in 3.9.b lahko zaključimo, da je dobljeno rangiranje atributov stabilno in ni zgolj odraz tega, katero hevristiko smo uporabili pri ocenjevanju projekcij.

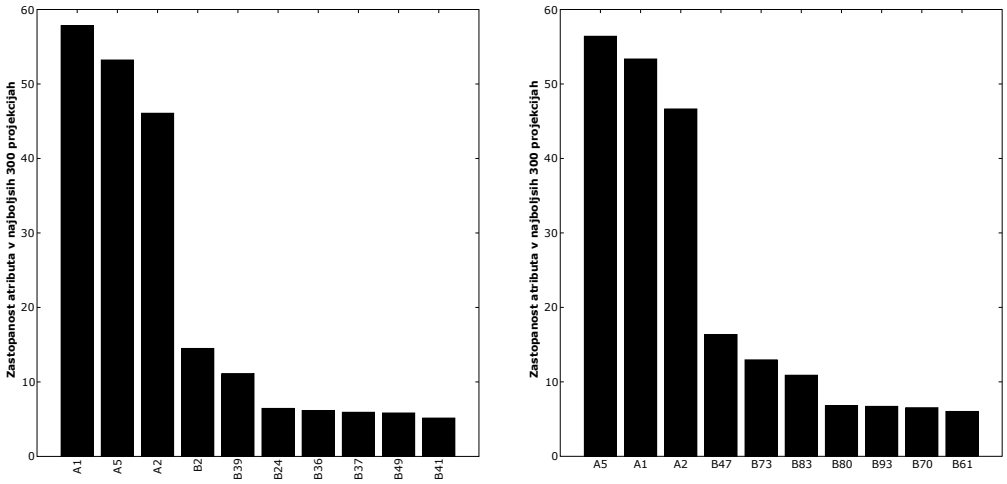
Izbira hevristike pa vseeno igra pomembno vlogo pri uspešnem ocenjevanju atributov pri visokodimenzionalnih zbirkah podatkov. Že v zgornjem eksperimentu, kjer so podatki vsebovali "le" 99 atributov, je bilo potrebno zaradi naključnega izbiranja atributov oceniti veliko število različnih projekcij – posamezni atributi so namreč morali biti v projekcijah prisotni dovolj pogosto, da je bila njihova ocena pomembnosti zanesljiva. Z uporabo hevristike se število projekcij, ki jih je potrebno oceniti, zmanjša, saj nam hevristika služi kot groba ocena pomembnosti posameznih atributov. Z njeno pomočjo se namreč pogosteje izberejo tisti atributi, ki bodo bolj verjetno ocenjeni kot dobri in manjkrat atributi, ki bi bili tudi sicer ocenjeni kot slabši.

Primerjava z mero ReliefF na spremenjeni domeni monks 1

V tem delu bomo pokazali enostaven primer, pri katerem mera ReliefF zaradi velikega števila atributov neuspešno oceni pomembnost atributov, mera na osnovi seznama projekcij pa pri tem nima težav. Za testno domeno smo izbrali sintetično zbirko podatkov monks 1, ki vsebujejo 124 primerov in 6 atributov (A_1 do A_6), razred Y pa ima vrednost 1, kadar velja $(A_1 = A_2) \vee (A_5 = 1)$. Atributi A_3, A_4 in A_6 so nepomembni in ne vplivajo na razred.

Zanimalo nas je, kako dodajanje naključnih atributov vpliva na uspešnost ocenjevanja pomembnosti atributov. Atribute smo zato z obema merama ocenili na originalnih podatkih (monks 1) ter na podatkih z 10 (monks 1 (+10)), 50 (monks 1 (+50)) ter 100 (monks 1 (+100)) dodatnimi atributi, katerih vrednosti so bile določene naključno. Z VizRankom smo na vsaki od teh domen ocenili do 20.000 projekcij radviz z največ štirimi atributi in za ocenjevanje uporabili 300 najbolje ocenjenih projekcij[‡]. Ker podatki vsebujejo diskretne atribute, smo mero ReliefF uporabili tudi pri hevristiki za rangiranje atributov. V tabeli 3.5 je prikazano, kako so z uporabo najboljših projekcij ter mere ReliefF na vseh štirih zbirkah podatkov rangirani atributi A_1, A_2 in A_5. Prva vrednost v vsaki celici tabele predstavlja rang atributa z uporabo seznama projekcij, druga vrednost pa

[‡]Pri originalnih podatkih smo zaradi zelo majhnega števila različnih projekcij atribute ocenili na podlagi zgolj desetih najboljših projekcij.



(a) (b)

Slika 3.10: Histograma atributov za domeni monks 1 (+50) (a) ter monks 1 (+100) (b). Grafa prikazujeta kako pogosto so posamezni atributi prisotni v 300 najbolje ocenjenih projekcijah radviz.

	monks 1	monks 1 (+10)	monks 1 (+50)	monks 1 (+100)
A_1	2 / 2	2 / 3	1 / 2	2 / 16
A_2	3 / 3	3 / 4	3 / 24	3 / 5
A_5	1 / 1	1 / 1	2 /1	1 / 2

Tabela 3.5: Rang pomembnih atributov na štirih različicah podatkov monks 1, dobljen z uporabo seznama najboljših 300 projekcij (prva vrednost) ter z uporabo mere ReliefF (druga vrednost).

rang atributa z uporabo mere ReliefF. Iz vrednosti v tabeli je razvidno, da prične ReliefF z večanjem števila atributov slabše ocenjevati koristne atribute, medtem ko opisana mera pri vseh štirih zbirkah podatkov primerno oceni edine tri pomembne atribute. Primera ocen najboljših desetih atributov pri podatkih monks 1 (+50) ter monks 1 (+100) prikazujeta histograma na sliki 3.10. Iz obeh diagramov je razvidno, da so atributi A_1, A_2 in A_5 v najboljših projekcijah zastopani bistveno pogosteje kot ostali atributi. Pomembno pri interpretaciji teh grafov je, da se zavedamo, da zaradi uteževanja projekcij glede na njihov rang višine stolpcev ne predstavljajo dejanskega števila pojavitev teh atributov v najboljših projekcijah.

Glede na to, da ima predstavljena mera dva glavna parametra, ki vplivata na dobljene ocene atributov – to sta število ocenjenih projekcij in število najboljših upoštevanih projekcij – smo želeli preveriti, kakšen vpliv imajo te vrednosti na dobljene ocene atributov. V ta namen smo izvedli različne eksperimente, kjer smo ocenili 10.000, 20.000, 50.000 ter 100.000 različnih projekcij radviz na domeni monks 1 (+100). Pri vsakem od eksperimentov smo nato atribute ocenili z uporabo 50, 100, 300 ter 1.000 najboljših projekcij. Ne

glede na izbrane vrednosti parametrov so bili atributi A_1, A_2 in A_5 vedno rangirani kot prvi trije atributi.

Namen predstavljenega primera uporabe nikakor ni bil dokazovati, da lahko z uporabo seznama ocenjenih projekcij bolje ocenjujemo pomembnost atributov kot z uporabo mere ReliefF. Cilj je bil zgolj demonstracija, kako lahko v posebnih primerih, kjer ReliefF zaradi prevelikega števila atributov odpove, vseeno ocenimo atribute tako, da upoštevamo potencialne odvisnosti med njimi.

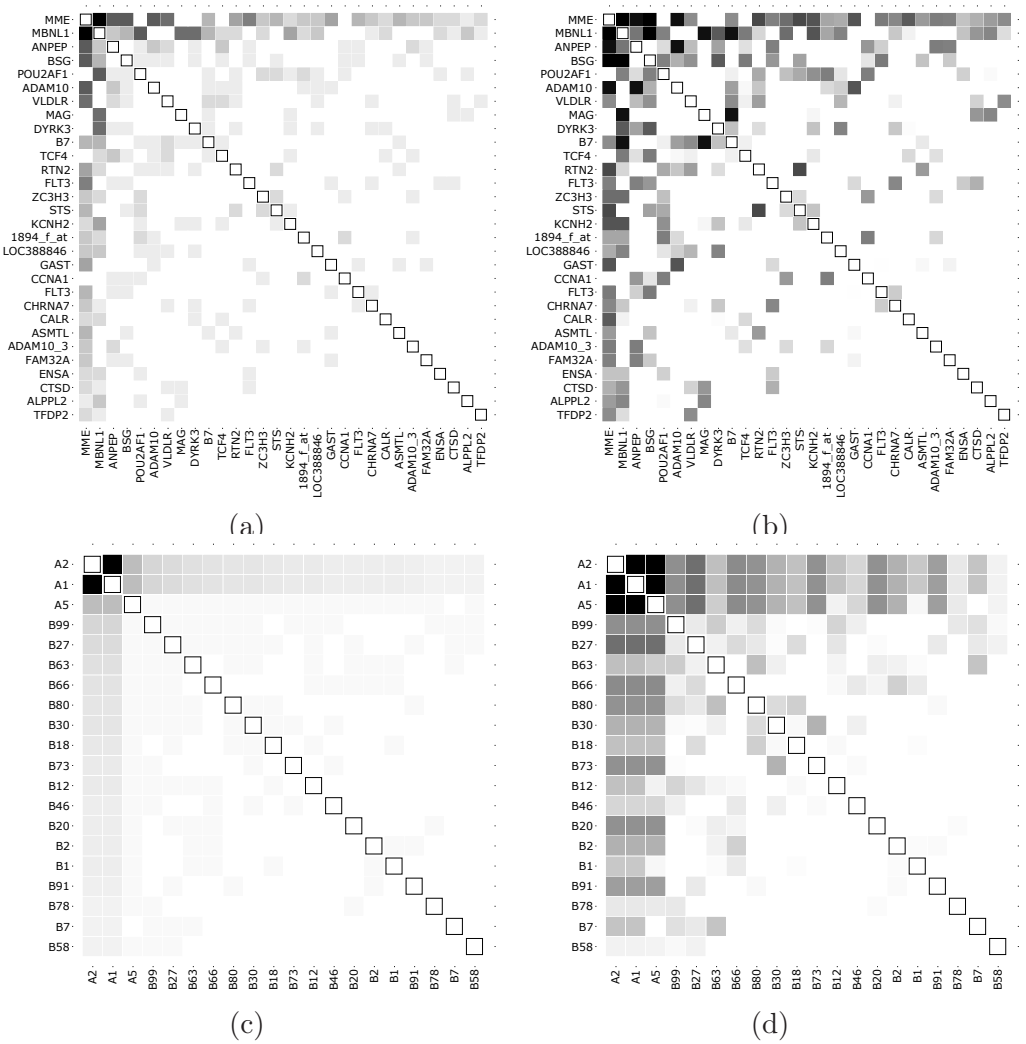
3.4.2 Odkrivanje interakcij med atributi

Kot smo že omenili, je težava kratkovidnih mer za ocenjevanje pomembnosti atributov to, da ne upoštevajo interakcij med atributi. Pri meri, ki smo jo predstavili v prejšnjem razdelku, se interakcije med atributi upoštevajo implicitno – atribut, ki je pomemben šele v interakciji z drugimi atributi, je dobro ocenjen, čeprav iz samega ranga atributa ni nikjer razvidno, da je to zgolj zaradi interakcij z drugimi atributi.

Interakcije med atributi pa je mogoče s pomočjo seznama ocenjenih projekcij odkrivati tudi neposredno. V majhni podmnožici najbolje ocenjenih projekcij si lahko ogledamo, kako pogosto se posamezni pari atributov hkrati pojavijo v isti projekciji. Če se nek par atributov pojavlja zelo pogosto, lahko to smatramo kot indikator, da je med atributoma določena interakcija.

Interakcije med atributi lahko odkrivamo z uporabo grafov, kot so prikazani na sliki 3.11. V njih na x in y osi prikažemo tiste atribute, ki se pojavljajo v najbolje ocenjenih projekcijah. Za vsak par različnih atributov, ki se hkrati pojavita v eni od najboljših m projekcij, narišemo kvadrat, katerega sivino lahko uporabimo za prikaz neke dodatne pomembne informacije. V grafih na sliki 3.11 smo sivino določili na dva načina. Pri grafih (a) in (c) sivina predstavlja pogostost pojavljanja danega para atributov v najboljših projekcijah – večja kot je pogostost, temnejši je kvadrat. V grafih (b) in (d) pa sivina predstavlja oceno najboljše najdene projekcije, ki vsebuje dani par atributov – višje ocene so predstavljene s temnejšim kvadratom. Pri iskanju interakcij je smiselno uporabiti obe vrsti grafov. Prvi nam namreč pove, kateri pari atributov so najverjetneje v interakciji, drugi pa pove, katere od interakcij so bolj pomembne za uspešno ločevanje razredov. Vzemimo naprimer, da ocenjujemo interakcije na osnovi 300 najboljših projekcij, kjer odkrijemo, da se nek par atributov pojavi v vsaki od zadnjih 50 projekcij (temen kvadrat v prvem grafu). Ta pogostost vsekakor nakazuje veliko verjetnost interakcije med atributoma. Glede na to, da se ta par ne pojavi tudi med najboljšimi projekcijami (svetel kvadrat v drugem grafu), pa je njuna interakcija verjetno manj pomembna kot naprimer interakcija med atributoma, ki se pojavita v vsaki od najboljših 20 projekcij.

Grafa na sliki 3.11 sta bila dobljena iz projekcij radviz pri podatkih MLL ter monks 1 (+100), pri čemer smo upoštevali tristo ter sto najboljših projekcij. Pri podatkih MLL (grafa (a) in (b)) lahko odkrijemo, da je med atributoma *MME* in *MBNL1* zelo verjetno pomembna interakcija, saj se v projekcijah zelo pogosto pojavljata hkrati, poleg tega pa je tudi najboljša projekcija z njima zelo dobro ocenjena. Oba atributa se pogosto



Slika 3.11: Odkrivanje interakcij med atributi na podatkih MLL (a,b) ter monks 1 (+100) (c,d). Za vsak par atributov sivina kvadrata ponazarjaja pogostost njune skupne pojavitve v najboljših projekcijah (a,c) ter oceno najboljše projekcije, ki vsebuje dani par atributov (b,d).

pojavita tudi z drugimi atributi, kar nakazuje, da sta tudi vsak zase zelo pomembna za ločevanje med razredi. Za podatke monks 1 (+100) so rezultati prikazani v grafih (c) in (d). Iz njih je razvidno, da je najbolje ocenjena projekcija, ki vsebuje atribute A_1, A_2 in A_5. Tudi druge projekcije, ki vsebujejo samo nekatere od teh treh atributov, so ocenjene bistveno bolje od ostalih projekcij. V najboljših 100 projekcijah sta najpogosteje skupaj zastopana atributa A_1 in A_2, kar glede na ciljni koncept domene dokazuje, da lahko z uporabo prikazanih grafov uspešno odkrivamo interakcije med pari atributov.

Kot za ocenjevanje pomembnosti atributov lahko tudi za odkrivanje interakcij uporabimo poljubno točkovno vizualizacijsko metodo. Vseeno velja pripomniti, da je uspešnost odkrivanja interakcij zelo odvisna od tipa vsebovane interakcije. Če je vsebovana inter-

akcija taka, da je z izbrano vizualizacijsko metodo ne moremo uspešno prikazati, potem jo z uporabo opisanega postopka ne bo mogoče odkriti.

3.4.3 Iskanje osamelcev

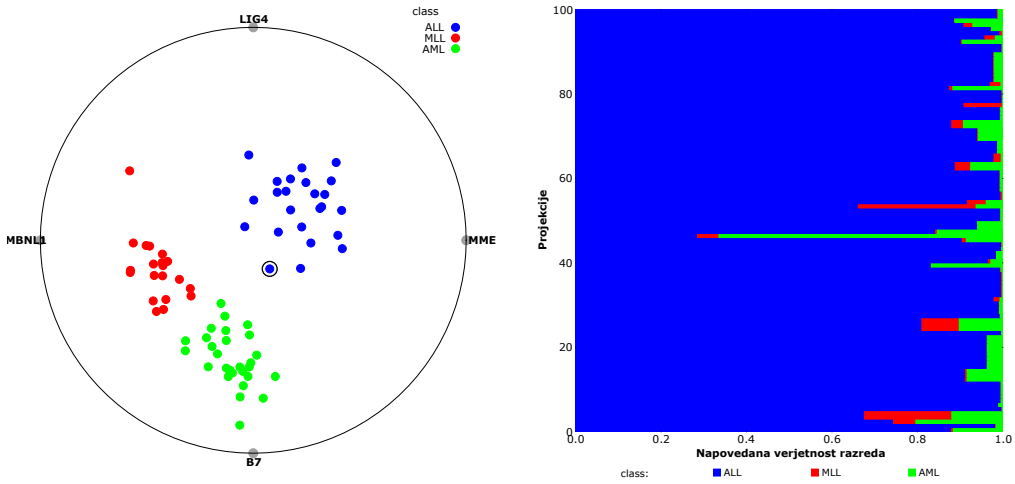
Osamelci so primeri, pri katerih vrednosti atributov bistveno odstopajo od vrednosti atributov ostalih primerov. V primeru klasificiranih podatkov med osamelce štejemo tudi tiste primere, ki ne ležijo med primeri svojega razreda, ampak med primeri nekega drugega razreda. Taki primeri so lahko napačno klasificirani ali pa vsebujejo zgolj določene karakteristike, ki so tipične za primere drugega razreda. Identifikacija teh primerov ni enostavna. Če kot kriterij podobnosti primerov uporabimo razdalje med primeri, pridemo namreč v primeru večjega števila atributov ponovno do problema zaradi "prekletstva dimenzionalnosti". Ne glede na podobnost primerov so namreč razdalje med njimi v visokodimenzionalnih prostorih velike.

Iskanje primerov, ki niso tipični predstavniki svojega razreda, je mogoče uspešno izvesti z uporabo seznama najboljših najdenih projekcij. Za vsakega od primerov lahko analiziramo, kakšni primeri ležijo v njegovi okolici v vsaki od izbranih projekcij. Če se naprimer v okolici nekega primera v večini projekcij nahajajo samo primeri istega razreda, potem lahko ta primer smatramo kot tipičen predstavnik tega razreda. Med netipične ali celo napačno klasificirane pa je smiselno uvrstiti tiste primere, ki v projekcijah pogosto ležijo blizu ali celo med primeri nekega drugega razreda. Čeprav se pri tovrstnem ocenjevanju podobnosti primerov še vedno zanašamo na razdalje med primeri, je prednost takega načina ta, da se omejimo zgolj na upoštevanje tistih atributov, ki so (glede na to, da so projekcije zanimive) bistveni za ločevanje med različnimi razredi.

Oglejmo si primer na podatkih MLL. Ocenili smo 20.000 projekcij radviz z največ štirimi atributi, najboljša najdena projekcija pa je prikazana na sliki 3.12.a. Za obkroženi primer iz razreda *ALL* bi radi ocenili ali pogosto leži na robu svoje gruče primerov ali pa je to le izjemoma. V vsaki od sto najboljših projekcij smo zato z uporabo algoritma k-NN izračunali, kakšna je napovedana verjetnost posameznih razredov. Te verjetnosti so prikazane na sliki 3.12.b. Vsaki projekciji ustreza na sliki ena vrstica, kjer je z barvo ponazorjena napovedana verjetnost posameznega razreda. Glede na to, da na sliki močno prevladuje modra barva (razred *ALL*), lahko sklepamo, da primer ni napačno klasificiran. Ker posamezne vrstice vseeno vsebujejo kar nekaj zelene in rdeče barve, pa je to indikator, da primer pogosto leži na robu svoje gruče točk.

3.5 Veljavnost prikazanih zakonitosti

VizRank išče zanimive projekcije z izbiranjem različnih podmnožic atributov in njihovo vizualizacijo v različnem vrstnem redu. Ker je mogoče pri številnih zbirkah podatkov najti projekcije z dobro ločenostjo razredov, se pri tem poraja vprašanje o dejanski veljavnosti prikazanih zakonitosti. Glede na to, da običajno ocenimo veliko število projekcij, bi bila



(a) (b)

Slika 3.12: Za izbrani primer na sliki (a) lahko ugotovimo, kakšna je njegova okolica v stotih najboljših najdenih projekcijah. Vsaka vrstica slike (b) predstavlja napovedno točnost algoritma k-NN za izbran primer na eni od stotih najboljših projekcij.

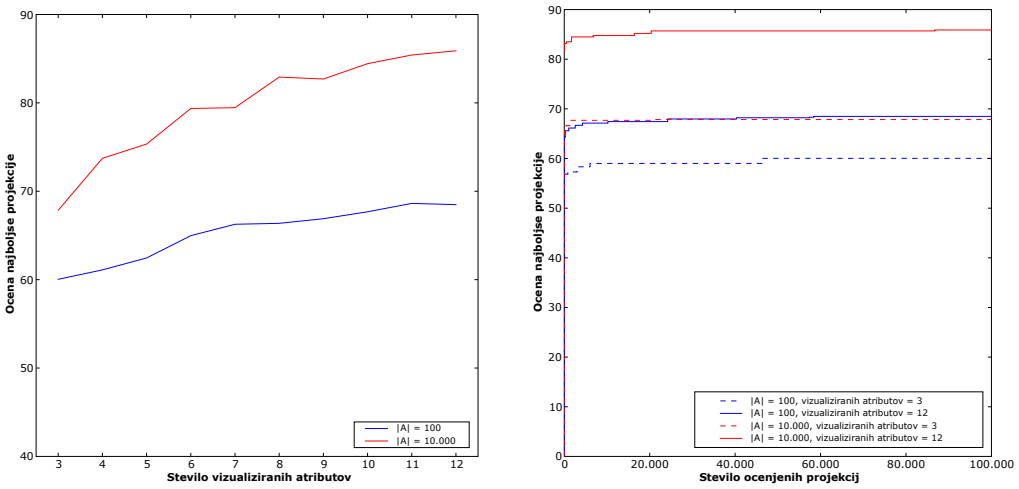
dobra ločenost razredov v najboljših projekcijah lahko tudi zgolj rezultat pretiranega prilagajanja podatkom.

Da bi preverili veljavnost prikazanih zakonitosti smo izvedli dva eksperimenta. Pri prvem smo preverili, ali lahko z dovolj vztrajnim preiskovanjem prostora projekcij odkrijemo regularnosti tudi v naključno generiranih podatkih, v drugem eksperimentu pa smo preverili kakšno napovedno točnost lahko dosežemo, če najboljšo najdeno projekcijo uporabimo za klasifikacijo novih primerov.

3.5.1 Poskus z naključno generiranimi podatki

Zanimalo nas je, kako dobre projekcije lahko najdemo, če analiziramo naključno generirane podatke. V ta namen smo sestavili dve zbirki podatkov s stotimi učnimi primeri. Prva zbirka je vsebovala 100, druga pa 10.000 zveznih atributov. Obe zbirki sta vsebovali binaren razred. Vrednosti atributov in razreda smo določili naključno. Z uporabo posebne hevristike (glej razdelek 3.2.2) smo ocenili 100.000 projekcij radviz za vsako zbirko podatkov ter za različno število vizualiziranih atributov (od 3 do 12 atributov).

Dobljeni rezultati so prikazani na sliki 3.13. Graf 3.13.a prikazuje kakšen vpliv ima število hkrati vizualiziranih atributov na najboljšo najdeno projekcijo. Po pričakovanjih je dobljena krivulja naraščajoča, saj lahko z večjim številom uporabljenih atributov uspešneje ločimo med razredoma. Opazna je tudi razlika med obema zbirkama podatkov. Zbirka z 10.000 atributi ima bistveno višje ocene projekcij, kot zbirka s stotimi atributi. Tudi to je pričakovano, saj je pri velikem številu naključno generiranih atributov večja verjetnost, da bodo kateri od atributov relativno uspešno ločevali med razredoma. Na sliki 3.13.b je prikazana odvisnost med številom ocenjenih projekcij ter oceno najboljše



(a) (b)

Slika 3.13: Graf (a) prikazuje, kako število vizualiziranih atributov vpliva na oceno najboljše najdene projekcije. Krivulje na sliki (b) prikazujejo, kako se med ocenjevanjem projekcij spreminja ocena najboljše najdene projekcije pri obeh zbirkah podatkov pri poskusu s tremi ter dvanajstimi vizualiziranimi atributi.

najdene projekcije. Iz grafa je razvidno, da se z večanjem števila ocenjenih projekcij ocena najboljše projekcije spreminja le malenkostno.

Na osnovi opisanih rezultatov lahko sklepamo naslednje: Verjetnost, da bomo v podatkih, ki ne vsebujejo pomembnih zakonitosti, odkrili zanimivo projekcijo, se veča z večanjem števila vizualiziranih atributov ter z večanjem števila atributov v zbirki podatkov. V nasprotju s temi parametri pa ima samo število ocenjenih projekcij majhen vpliv na zanimivost dobljenih projekcij, saj je z uporabo hevrističnega preiskovanja prostora projekcij očitno mogoče hitro identificirati najzanimivejše projekcije.

3.5.2 Napovedna točnost projekcij

Eden od načinov za preverjanje veljavnosti prikazane zakonitosti je, da najboljše projekcije uporabimo kot napovedne modele in ocenimo njihovo napovedno točnost. Kot pri običajnem testiranju učnih algoritmov podatke najprej razdelimo na učno in testno množico, nato pa z uporabo učne množice poiščemo zanimive projekcije. Primerom iz testne množice nato izračunamo njihov položaj v teh projekcijah. Vsakemu od testnih primerov lahko nato na podlagi njegovega položaja v projekciji napovemo vrednost razreda z uporabo algoritma k-NN. Testni primer bo klasificiran pravilno zgolj takrat, kadar bodo v njegovi okolici prevladovali učni primeri iz pravega razreda. Celotni postopek ponovimo večkrat z različno delitvijo podatkov na učno in testno množico, nato pa izračunamo napovedno uspešnost. Če projekcije z dobro ločenostjo razredov prikazujejo neko resnično zakonitost, bo to razvidno tako, da bo dobljena napovedna točnost visoka oziroma, kolikor je mogoče, primerljiva z ostalimi učnimi algoritmi.

Napovedno točnost, ki jo je mogoče doseči z uporabo projekcij, smo želeli izmeriti z uporabo projekcij radviz ter z uporabo linearnih projekcij, dobljenih z uteženo linearno diskriminantno analizo (glej razdelek 2.4.4). Pri tem smo uporabili dve skupini podatkov. Prva skupina vsebuje podatke, dobljene z uporabo mikromrež – to so zbirke levkemija, DLBCL, MLL, SRBCT, rak na prostati ter pljučni rak. Osnovne informacije o njih so opisane v dodatku A. Izbrane zbirke so še posebej primerne za preverjanje, ali pride do pretiranega prilagajanja, saj vsebujejo relativno majhno število primerov, ogromno število atributov, vrednosti primerov pa poleg tega vsebujejo veliko šuma. V drugo skupino podatkov smo uvrstili zbirke iz repozitorija UCI [80], ki se pogosto uporabljajo pri testiranju učnih algoritmov. Za ocenjevanje klasifikacijske točnosti smo uporabili 10-kratno prečno preverjanje. Na vsaki od omenjenih zbirk podatkov smo z obema vizualizacijskima metodama (metodo radviz in linearnimi projekcijami) na učni množici ocenili do 50.000 projekcij z največ desetimi atributi ter nato testne primere klasificirali z uporabo algoritma k-NN ($k = \sqrt{N}$) glede na njihov položaj v najboljši najdeni projekciji. Doseženo točnost smo primerjali s točnostjo, ki jo dosežejo metoda podpornih vektorjev (SVM) z uporabo RBF jeder (s standardnimi parametri $\gamma = 0,05$, $C = 1,0$, $p = 0,5$ ter $\epsilon = 0,001$), metoda k-NN ($k = 10$), naivni Bayes ter odločitvena drevesa (algoritem C4.5 z vrednostmi $m = 2$ in $cf = 0,25$). Poleg klasifikacijske točnosti smo za vsako metodo izračunali še povprečni rang metode na uporabljenih zbirkah podatkov.

Zgornji del tabele 3.6 vsebuje dobljene klasifikacijske točnosti na zbirkah podatkov, dobljenih z uporabo mikromrež. Zaradi visoke dimenzionalnosti podatkov smo točnosti algoritmov SVM, k-NN, naivnega Bayesa ter odločitvenih dreves izračunali na dva načina – pri prvem načinu so imeli ti algoritmi za učenje na voljo vse atribute, pri drugem pa smo iz podatkov izbrali zgolj sto najbolje ocenjenih atributov. Za ocenjevanje pomembnosti atributov smo uporabili kvocient signala proti šumu, ki se pri analizi tovrstnih podatkov najpogosteje uporablja.

Rezultati so pokazali, da je dosežena napovedna točnost projekcij zelo odvisna od izbire vizualizacijske metode. Pri tej skupini podatkov se je naprimer kot bistveno uspešnejša izkazala metoda radviz. Z njo smo pri eksperimentu brez izbora podmnožice atributov dosegli najboljši povprečni rang od vseh učnih algoritmov, pri izboru podmnožice atributov pa je bil njen povprečni rang primerljiv z metodo podpornih vektorjev ter metodo k-NN. Enakovrednost metod (oziroma celo rahla prednost metode k-NN) je zelo zanimiva, saj ne potrjuje bistvene superiornosti metode podpornih vektorjev, o kateri poročajo nekateri raziskovalci [100]. Rezultat je zanimiv tudi zato, ker dokazuje, da lahko z uporabo ene same najboljše ocenjene projekcije podatkov dosežemo točnost, ki je primerljiva s točnostjo sodobnih algoritmov strojnega učenja. V primerjavi z ostalimi uporabljenimi učnimi algoritmi (z izjemo odločitvenih dreves) je velika prednost pri uporabi projekcije kot modela njena enostavna interpretabilnost. Iz projekcije zelo enostavno ugotovimo vpliv posameznih atributov, vidimo podobnosti in razlike med različnimi razredi ter razumemo razloge, zaradi katerih primere klasificiramo v določen razred.

Rezultati na zbirkah podatkov iz repozitorija UCI so prikazani v spodnjem delu

Zbirke podatkov	VizRank (radviz)	VizRank (lin. proj.)	SVM	k-NN	Naivni Bayes	C4.5
podatki o raku	Z uporabo vseh atributov					
levkemija	97,14%	94,29%	87,86%	89,11%	97,14%	79,11%
SRBCT	97,78%	95,28%	92,92%	84,58%	96,39%	86,94%
MLL	97,14%	87,50%	93,04%	91,61%	93,21%	90,36%
DLBCL	93,39%	90,89%	87,14%	91,07%	83,21%	83,21%
prostata	95,18%	88,45%	92,36%	88,36%	81,36%	76,64%
pljučni rak	89,14%	89,67%	92,19%	96,10%	87,24%	89,10%
Povprečni rang	**1,58**	**3,50**	**3,33**	**3,50**	**3,66**	**5,41**
podatki o raku	Z uporabo stotih najpomembnejših atributov					
levkemija	97,14%	94,29%	98,57%	97,14%	97,14%	80,71%
SRBCT	97,78%	95,28%	98,89%	100,00%	98,75%	84,44%
MLL	97,14%	87,50%	92,86%	93,04%	87,50%	93,21%
DLBCL	93,39%	90,89%	91,96%	94,82%	89,46%	86,96%
prostata	95,18%	88,45%	93,27%	91,36%	91,27%	82,45%
pljučni rak	89,14%	89,67%	94,14%	94,14%	88,71%	82,21%
Povprečni rang	**2,50**	**4,58**	**2,25**	**2,08**	**4,25**	**5,33**
domene UCI						
adult	81,37%	82,80%	83,41%	79,73%	79,84%	76,96%
housing	82,82%	83,40%	87,76%	84,39%	79,26%	83,80%
imports-85	90,07%	89,05%	88,05%	71,21%	90,57%	92,55%
ionosphere	86,30%	90,02%	94,29%	86,61%	88,89%	91,44%
mushroom	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
shuttle	90,88%	97,23%	96,82%	97,23%	93,28%	98,02%
titanic	79,01%	78,74%	79,06%	78,51%	77,87%	78,92%
voting	94,24%	95,17%	96,31%	92,62%	90,32%	96,09%
wine	92,65%	97,75%	98,33%	95,46%	98,33%	94,38%
zoo	95,09%	94,18%	96,09%	96,09%	91,09%	96,09%
Povprečni rang	**4,25**	**3,40**	**2,10**	**4,00**	**4,40**	**2,85**

Tabela 3.6: Klasifikacijske točnosti različnih algoritmov na podatkih, dobljenih z uporabo mikromrež, ter na podatkih iz repozitorija UCI. Uspešnost štirih učnih algoritmov smo primerjali s točnostjo najboljših projekcij radviz ter linearnih projekcij, dobljenih z uteženo diskriminantno analizo. Rezultati so bili dobljeni z 10-kratnim prečnim preverjanjem.

tabele 3.6. Na teh domenah so se projekcije radviz izkazale slabše, njihova točnost pa je primerljiva s točnostjo, ki jo dosegata metodi k-NN ter naivni Bayes. V nasprotju s prejšnjo skupino podatkov pa smo v tem primeru dosegli relativno dobro klasifikacijsko točnost na osnovi najboljših linearnih projekcij, dobljenih z uteženo diskriminantno analizo.

Pomembno vprašanje, ki si ga lahko na osnovi prikazanih rezultatov zastavimo, je, kaj je razlog, zaradi katerega se napovedne točnosti obeh vizualizacijskih metod tako razlikujeta na različnih skupinah zbirk podatkov. Ker imamo pri uteženi diskriminantni analizi več fleksibilnosti pri natančnem nastavljanju parametrov projekcije (naklon in

dolžina osi posameznih atributov), bi bila ena od možnih razlag za to razliko ta, da pride v primeru diskriminantne analize pri podatkih iz mikromrež do pretiranega prilagajanja učnim podatkom. Če bi bilo to res, potem bi lahko upravičeno pričakovali, da so najboljše projekcije, dobljene z uporabo utežene diskriminantne analize bolje ocenjene kot najboljše projekcije radviz. Kot bo to razvidno iz tabele 3.7 v razdelku 3.6.2, temu ni tako – pri vseh podatkih iz mikromrež smo namreč boljše projekcije našli prav z metodo radviz. Verjetnejši razlog za razliko je ta, da podatki iz mikromrež enostavno vsebujejo tip zakonitosti, ki jo je mogoče uspešneje prikazati s tipom transformacije (projekcije), ki jo uporablja metoda radviz, medtem ko so domene iz repozitorija UCI uspešneje vizualizirane z uporabo linearnih projekcij.

Na osnovi vseh opravljenih eksperimentov lahko sklepamo, da so lastnosti, vidne v projekcijah z dobro ločenimi razredi, resnične in niso zgolj rezultat pretiranega prilagajanja podatkom. Z uporabo projekcij kot napovednih modelov je mogoče doseči zelo visoko napovedno točnost, ki je primerljiva z ostalimi učnimi metodami. Prednost pri uporabi projekcij v primerjavi z ostalimi metodami je v interpretabilnosti – projekcije, uporabljene kot modeli, so enostavno razumljive, zaradi česar jim je veliko lažje zaupati kot modelom, ki delujejo na principu črne škatle. Klasifikacija primerov z uporabo projekcij pa vsekakor ni primerna za vse učne probleme – projekcija je konec koncev transformacija, ki preslika atribute iz večdimenzionalnega prostora v zgolj dve dimenziji. Kadar take transformacije ni mogoče narediti na način, da bi se pri tem kolikor je mogoče ohranila ločenost razredov, so dobljene projekcije in klasifikacija na njihovi osnovi slabe.

3.6 Primeri uporabe

Da bi ugotovili, kako uspešen je VizRank pri iskanju zanimivih projekcij, smo izvedli eksperimente na različnih zbirkah podatkov. Baze smo izbrali tako, da vsebujejo čim večje število atributov, saj je v tem primeru postopek iskanja dodatno otežen zaradi ogromnega števila možnih projekcij. Za vsako od uporabljenih zbirk podatkov se podrobnejše informacije o podatkih nahajajo v dodatku A.

3.6.1 Podatki o kvasovki

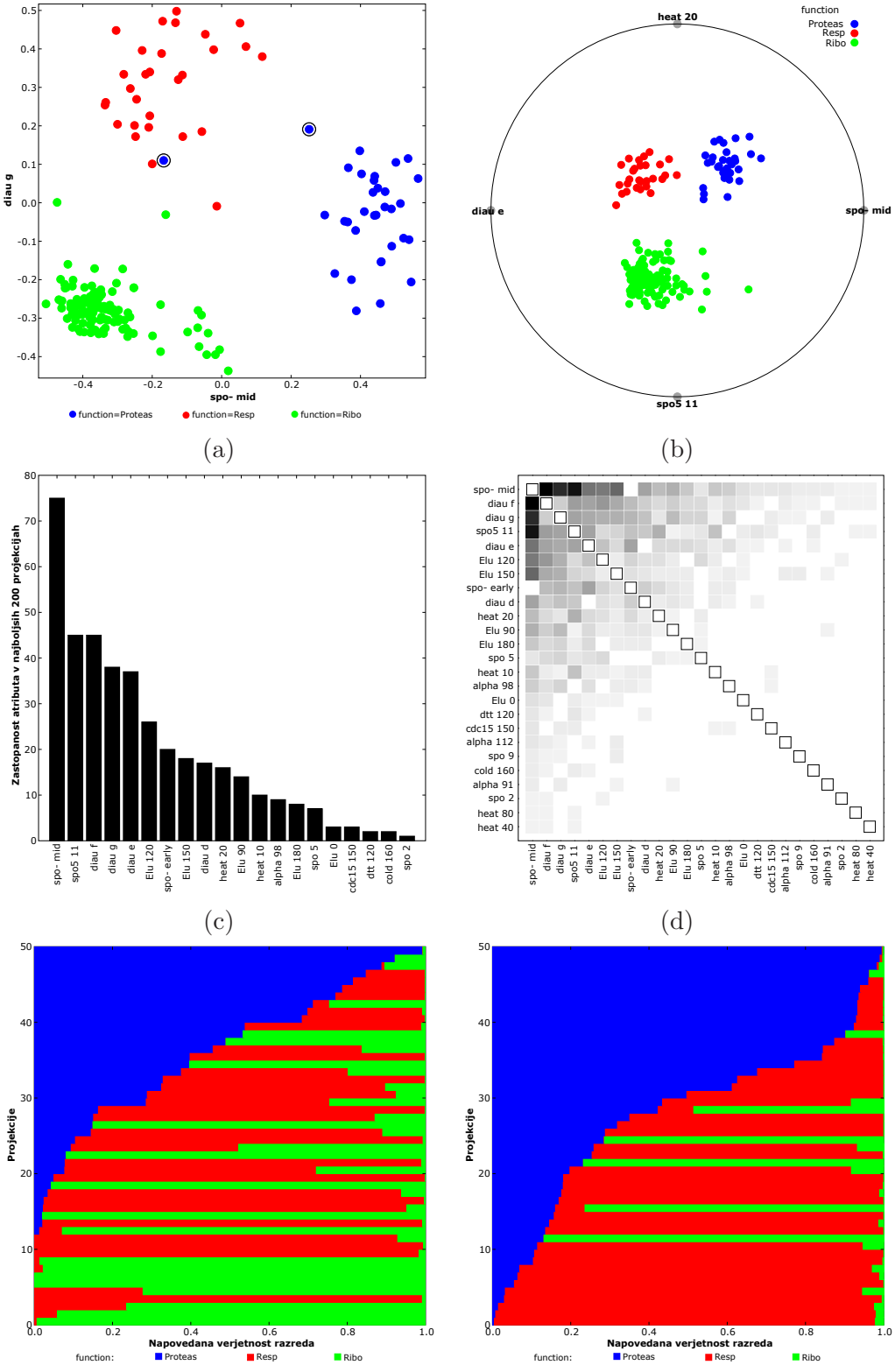
Zbirka podatkov o kvasovki *Saccharomyces cerevisiae* je s področja funkcijske genomike in vsebuje podatke o tem, kako so v različnih časovnih točkah pri osmih različnih eksperimentih (kot naprimer toplotni šok, izpostavljenost nizki temperaturi, regulacija metabolizma itd.) izraženi posamezni geni v organizmu [28]. Podatki vsebujejo 79 atributov (meritev v različnih časovnih točkah pri različnih eksperimentih) ter 186 primerov (genov), ki so razdeljeni v tri razrede, glede na funkcijo genov: *Ribo* (citoplazemski ribosomi, 121 primerov), *Proteas* (proteasomi, 35 primerov) ter *Resp* (celično dihanje, 30 primerov). To zbirko podatkov (vključno s še dvema razredoma, ki smo ju zaradi zelo majhnega števila primerov odstranili) je pri svoji analizi uporabil tudi Brown [9] in z njo pokazal superi-

ornost metode podpornih vektorjev pred ostalimi postopki strojnega učenja pri analizi genskih podatkov.

Z metodo VizRank smo poiskali zanimive projekcije z uporabo razsevnih diagramov ter metode radviz. Ocenili smo vseh 3.081 različnih razsevnih diagramov, katerih ocene so bile med 99,45 (najboljša projekcija) ter 50,86 (najslabša projekcija). Zanimivo je, da je vseh najboljših deset projekcij vsebovalo en atribut iz ene od časovnih točk v procesu sporulacije, drugi atribut pa je bil iz eksperimenta s toplotnim šokom ali pa regulacijo metabolizma. Najboljši razsevni diagram je prikazan na sliki 3.14.a. Iz njega je razvidno, da lahko z eno samo meritvijo izraženosti genov med sporulacijo jasno ločimo gene iz funkcijske skupine proteasomov od ostalih genov. Da bi ločili še med ostalima dvema funkcijskima skupinama, pa v tem primeru potrebujemo še atribut iz eksperimenta z regulacijo metabolizma. Uporabnost meritev pri regulaciji metabolizma za ločevanje med dvema skupinama – citoplazemskih ribosomov ter genov, ki uravnavajo celično dihanje – so potrdile tudi predhodne študije [26].

V svoji raziskavi je Brown podrobno analiziral tudi manjšo skupino genov, ki so jih različne metode podpornih vektorjev konsistentno napačno klasificirale. Ti geni so najverjetneje osamelci ali pa napačno klasificirani primeri. S postopkom, ki smo ga opisali v razdelku 3.4.3, smo netipičnost posameznih primerov analizirali tudi mi in sicer na osnovi najboljših 50 razsevnih diagramov. Dva gena (*YDR069C* ter *YDL020C*), ki sta v teh projekcijah najpogosteje ležala izven svoje skupine primerov, sta označena v razsevnem diagramu na sliki 3.14.a, njuna grafa verjetnosti pa sta prikazana na slikah 3.14.e in 3.14.f. Posamezne vrstice so v teh grafih tokrat urejene glede na padajočo verjetnost pravilnega razreda, saj je tako lažje opaziti, kakšno je povprečje preko uporabljenih projekcij. Za oba gena je takoj očitno, da pogosto ležita med geni drugih razredov – gen *YDR069C* oscilira med vsemi tremi razredi, gen *YDL020C* pa predvsem med razredoma *Proteas* ter *Resp.* Zanimivo je, da sta oba gena med tistimi, ki so bili tudi v Brownovi raziskavi konsistentno napačno klasificirani in za katera je dobro znano, da sta zaradi drugačne regulacije šibkeje povezana s svojo funkcijsko skupino.

Pri uporabi metode radviz smo se zaradi dobre ločenosti razredov že v razsevnih diagramih omejili na ocenjevanje projekcij z največ štirimi atributi. Tudi pri projekcijah radviz se je izkazalo, da so za dobro ločenost razredov potrebne meritve iz vsaj dveh različnih eksperimentov, naprimer sporulacije in regulacije metabolizma. Nobena projekcija z dobro ločenimi razredi namreč ni vsebovala samo atributov iz enega samega eksperimenta. Ta rezultat je pomemben tudi iz biološkega vidika, saj priča o tem, kakšno je minimalno število eksperimentov, ki jih moramo v tej domeni napraviti za uspešno določitev funkcije gena. Najboljša izmed 20.000 ocenjenih projekcij je prikazana na sliki 3.14.b. Omogoča odlično ločevanje med razredi ter enostavno interpretacijo: atributa *diau e* in *spo-mid* ločujeta med proteasomi in geni, ki uravnavajo celično dihanje, atributa *spo5 11* in *heat 20* pa ločujeta ribosome od genov iz ostalih dveh funkcijskih skupin. Seznam stotih najboljših projekcij radviz smo uporabili tudi za analizo pomembnosti atributov (slika 3.14.c) ter interakcij med pari atributov (slika 3.14.d). Zanimivo je, da se atribut



Slika 3.14: Najboljši razsevni diagram (a) ter projekcija radviz s štirimi atributi (b) pri podatkih o kvasovki. Najpogostejših 15 atributov (c) ter interakcije med njimi (d) pri analizi stotih najboljših radviz projekcij. Grafa (e) in (f) prikazujeta napovedane verjetnosti razredov za gena, označena v razsevnem diagramu (a).

spo-mid pojavi kar v 75 projekcijah, medtem ko se ostali atributi pojavljajo bistveno redkeje. Glede na prikazan histogram bi lahko zaključili, da je za napovedovanje razreda smiselno uporabljati največ prvih 15 atributov. V grafu na sliki 3.14.d je prikazana pogostost pojavljanja posameznih parov atributov v najboljših sto projekcijah. Izkaže se, da atribut *spo-mid* zelo pogosto nastopa v paru z atributi *diau f* (35-krat), *diau g* (28-krat) ter *spo5 11* (36-krat). Ostali atributi se v parih pojavljajo bistveno redkeje.

Brown je v svoji raziskavi pokazal, da lahko z metodo podpornih vektorjev na teh podatkih dosežemo zanesljivo klasifikacijo. Nikjer v raziskavi pa avtor ne poroča o tem, da bi lahko za ločevanje med razredi definirali enostavna in razumljiva pravila. Z opisanim eksperimentom smo pokazali, da taka pravila obstajajo in lahko celo signifikantno izboljšajo razumevanje problemskega področja. Namesto samoumevnega zanašanja na "najboljši" algoritem strojnega učenja se je zato včasih (vsaj v začetnih fazah analize podatkov) bolj smiselno obrniti na enostavne postopke, ki prikažejo rezultate na človeku razumljiv način.

3.6.2 Podatki o različnih vrstah rakastih obolenj

V drugi primer uporabe sodijo podatki, ki smo jih za demonstriranje različnih hevristik in metod uporabljali že skozi celotno poglavje ter vsebujejo informacije o izraženosti posameznih genov pri pacientih z različnimi vrstami rakastih obolenj. Rakasta obolenja so posledica progresivnih genetskih sprememb, ki vodijo pretvorbo normalnih celic v njihove maligne derivate. S tehnologijo DNA mikromrež (ang. *DNA microarrays*) lahko simultano merimo izražanje več tisoč genov v biološkem vzorcu, kar je v zadnjem desetletju omogočilo velik napredek pri raziskavah raka. Številne novejše raziskave so pokazale superiorne diagnostične zmožnosti DNA mikromrež za klasifikacijo rakastih obolenj v primerjavi s standardnimi morfološkimi kriteriji [41, 96, 82]. Cilji uporabe mikromrež v raziskavah raka so vpogled v proces karcinogeneze, identifikacija biomarkerjev za različne tipe raka, natančnejša klasifikacija ter izboljšanje in individualizacija zdravljenja z razvojem novih, usmerjenih terapevtikov.

Poleg velike količine šuma je največji problem podatkov o genski izraženosti njihova visoka dimenzionalnost. Podatki namreč tipično vsebujejo več tisoč atributov (genov) in samo majhno število primerov (pacientov). Analitiki se pri analizi tovrstnih podatkov običajno poslužujejo številnih postopkov za modeliranje ter izbiranje in konstrukcijo novih atributov. Kot primer si oglejmo študijo, ki jo je opravil Khan s sod. [62] na podatkih SRBCT. Avtorji so najprej odstranili gene z nizkim nivojem izraženosti, nato naučili 3.750 nevronskih mrež na različnih podmnožicah genov, izbranih z uporabo analize osnovnih komponent, analizirali pomembnost posameznih genov v dobljenih nevronskih mrežah in tako prišli do 96 genov, s katerimi so nato z uporabo večdimenzionalnega skaliranja dosegli uspešno ločevanje med različnimi razredi. Tudi druge študije so običajno podobne kompleksnosti, zaradi česar je rezultate takih raziskav zelo težko interpretirati.

Da bi preverili, kako težko je pri tovrstnih podatkih ločiti med razredi, smo VizRank uporabili na šestih javno dostopnih zbirkah podatkov: levkemija, SRBCT, MLL, DLBCL,

	Vizualizacijska metoda		
	Razsevni d.	Radviz	Lin. projekcije
levkemija	96,54%	99,76%	98,21%
SRBCT	83,52%	99,84%	99,28%
MLL	90,12%	99,83%	99,79%
DLBCL	89,34%	98,22%	98,08%
prostata	87,34%	95,14%	94,45%
pljučni rak	75,48%	93,48%	90,14%

Tabela 3.7: Ocene ($\overline{P}$, glej enačbo 3.1) najboljših najdenih projekcij na različnih zbirkah podatkov pri uporabi razsevnih diagramov, metode radviz ter linearnih projekcij, dobljenih z uteženo diskriminantno analizo. Z vsako vizualizacijsko metodo je bilo ocenjenih 50.000 projekcij.

rak na prostati ter pljučni rak. Nabori vsebujejo podatke o izraženosti 2.308 do 12.600 genov pri 72 do 203 bolnikih z rakom. Primeri so razvrščeni v dva do pet diagnostičnih skupin (različnih podvrst določenega raka). Za vsako od teh zbirk smo uporabili tri vizualizacijske metode – razsevne diagrame, metodo radviz in splošne linearne projekcije, dobljene z uteženo linearno diskriminantno analizo – ter z vsako od njih z VizRankom ocenili 50.000 projekcij. Pri metodi radviz in linearnih projekcijah smo se omejili na ocenjevanje projekcij z največ osmimi atributi. V tabeli 3.7 so zbrane ocene najboljših najdenih projekcij za vsako od vizualizacijskih metod na posamezni zbirki podatkov. Za nekatere od zbirk podatkov so najboljše projekcije prikazane na slikah 3.15.a do 3.15.d.

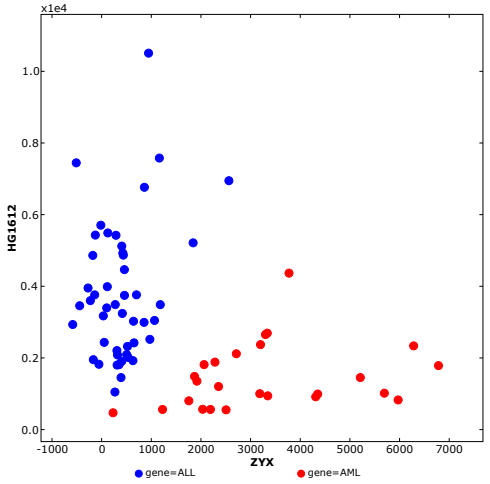
Najslabše od vizualizacijskih metod se je odrezal razsevni diagram, kar je pričakovano glede na to, da hkrati prikaže le dva atributa. Glede na to, da na uspešnost ločevanja med razredi vpliva tudi število razredov v podatkih, je VizRank najboljši razsevni diagram našel pri podatkih o levkemiji (2 razreda), najslabšega pa pri podatkih o pljučnem raku (5 razredov). V primeru metode radviz ter linearnih projekcij so dobljene ocene najboljših projekcij primerljive. Presenetljivo je, da so ocene linearnih projekcij, kljub njihovi večji izrazni moči, celo malenkost nižje od ocen projekcij radviz. Za to je možnih več razlogov. Eden od njih je, da sta kriterij, po katerem VizRank ocenjuje projekcije, ter kriterij, ki ga optimizira utežena linearna diskriminantna analiza, rahlo različna, zaradi česar dobljena projekcija ni popolnoma enaka optimalni linearni projekciji, kateri bi VizRank določil maksimalno oceno. Drugi zelo verjeten razlog je, da so zakonitosti v podatkih enostavno take, da jih je mogoče uspešneje vizualizirati z metodo radviz, kot pa z uporabo linearnih projekcij.

Za gene, ki nastopajo v najboljših projekcijah, smo ob pomoči domenskega eksperta analizirali njihovo biološko pomembnost. Glede na to, da pri večini zbirk podatkov poskušamo ločevati med različnimi tipi tumorja, smo predvidevali, da bodo najboljši geni najverjetneje označevalci tkivnega ali celičnega izvora vzorca in da ne bodo neposredno povezani z nastankom raka. Izkazalo pa se je, da so kljub temu številni geni, ki nastopajo v najboljših projekcijah, anotirani kot neposredno ali posredno povezani z rakom glede na

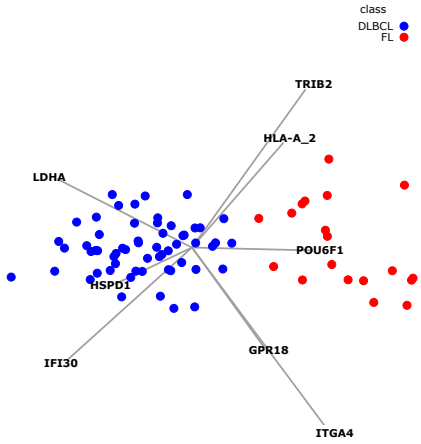
Atlas genetike in citogenetike v onkologiji in hematologiji. V primeru podatkov o raku na prostati, kjer poskušamo ločevati med zdravim in rakavim tkivom, pa bi na drugi strani pričakovali, da so pomembni geni predvsem tisti, ki so povezani z rakom. Hipoteza se izkaže kot pravilna, saj je glede na Atlas genetike v najboljši projekciji radviz kar šest od osmih genov (*LMO3*, *RBP1*, *HSPD1*, *HPN*, *MAF* in *TGFB3*) povezanih z rakom.

Na sliki 3.15.d je prikazana najboljša projekcija radviz za podatke o pljučnem raku. Ločenost različnih razredov je dobra, z izjemo nakaterih primerov iz razreda adenokarcinomov (AD), ki ležijo med primeri iz razreda ploščato celičnega karcinoma (SQ). Pri analizi dodatnih informacij o teh primerih smo ugotovili, da so bili nekateri od primerov razreda AD dejansko histološko diagnosticirani kot adenokarcinomi z lastnostmi ploščato celičnega karcinoma. Ti primeri so v projekciji prikazani kot prazni modri krožci. Kot je mogoče opaziti, jih večina leži prav v področju, kjer se nahajajo primeri razreda SQ. Za primer, ki je na sliki izbran, smo poleg tega analizirali tudi, kje se nahaja v najboljših stotih projekcijah. Graf verjetnosti je prikazan na sliki 3.15.f, vrstice v njem pa so urejene glede na padajočo verjetnost pravilnega razreda. Opazimo lahko, da primer zelo enakovredno leži v skupinah primerov AD ter SQ, kar dodatno potrjuje, da ima lastnosti obeh tipov tumorja.

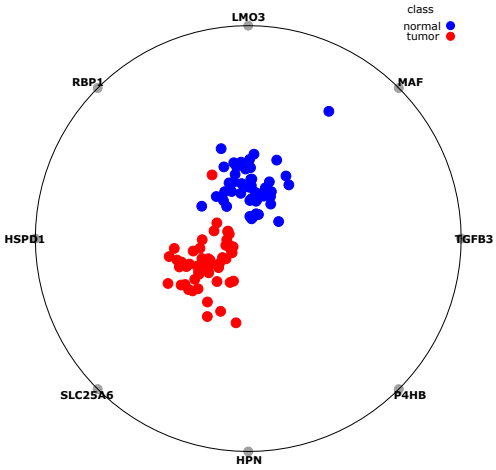
Za podatke o levkemiji smo ocenili tudi pogostost pojavljanja posameznih atributov v najboljših 50 razsevnih diagramih. Histogram dvajsetih najpogostejših atributov je prikazan na sliki 3.15.e. Iz grafa je takoj očitno, da v primerjavi z ostalimi geni gen ZYX (*zyxin*) bistveno izstopa v številu pojavitev. Ta gen je bil zelo pogosto prisoten tudi v najboljših projekcijah radviz ter splošnih linearnih projekcijah. Pri podrobnejši analizi gena *zyxin* smo odkrili, da je bila njegova uporabnost pri ločevanju obeh vrst levkemije potrjena tudi v drugih študijah. V originalni študiji teh podatkov, ki jo je opravil Golub s sod. [41], je bil prav *zyxin* izbran kot eden najpomembnejših genov za ločevanje med razredoma. Tudi v študiji, ki jo je opravil Wang s sod. [107], kjer so sistematično preučili in primerjali različne algoritme za izbiro podmnožic atributov, poročajo o tem, da je bil *zyxin* izbran kot najpomembnejši gen pri večini uporabljenih algoritmov. Ti rezultati dodatno potrjujejo, da je z uporabo najboljših projekcij mogoče uspešno identificirati pomembne atribute.



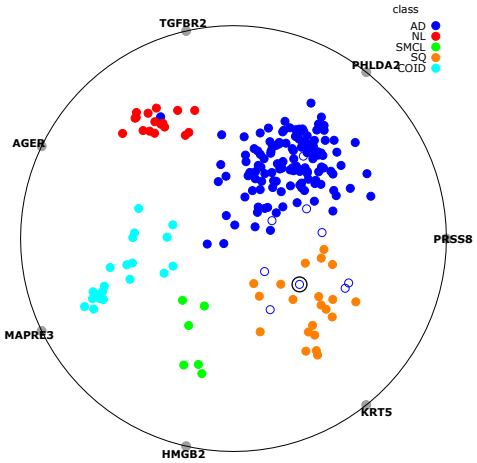
(a) levkemija



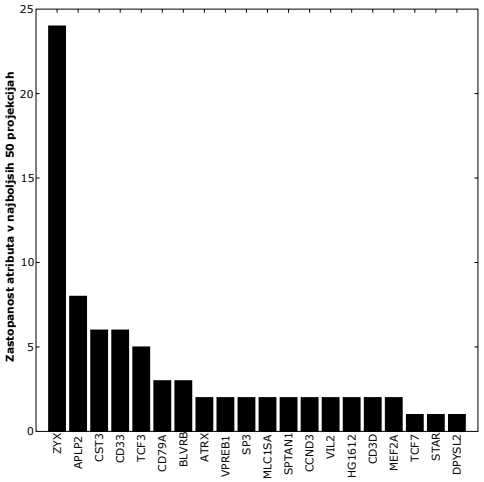
(b) DLBCL



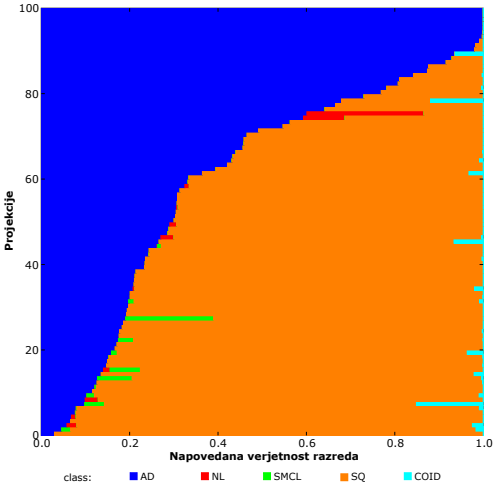
(c) rak na prostati



(d) pljučni rak



(e) levkemija / histogram atributov



(f) pljučni rak / iskanje osamelcev

Slika 3.15: Najboljše najdene projekcije pri podatkih o različnih vrstah raka (a–d), histogram pomembnih atributov pri levkemiji (e) ter primer iskanja osamelcev pri podatkih o pljučnem raku (f).

Poglavje 4

Ocenjevanje in rangiranje prikazov s paralelnimi koordinatami

V prejšnjem poglavju smo predstavili postopek VizRank, ki uspešno oceni in rangira prikaze, v katerih so primeri prikazani kot točke v 2D prostoru. V tem poglavju bomo definirali mero zanimivosti tudi za prikaze s paralelnimi koordinatami ter metodo VizRank razširili tako, da bo primerna tudi za ocenjevanje tovrstnih prikazov. Glede na to, da lahko smatramo vizualizacijski metodi radviz in metodo paralelnih koordinat kot komplementarni – vsaka od njih ima svoje prednosti – bomo predstavili še enostaven način, s katerim lahko atribute za prikaz v paralelnih koordinatah smiselno izberemo in uredimo na osnovi zanimivih projekcij radviz.

4.1 Uvod

V razdelku 2.5, kjer smo ocenili primernost različnih vizualizacijskih metod za odkrivanje znanja v podatkih, smo pokazali, da je metoda paralelnih koordinat zelo uspešna pri odkrivanju širokega spektra zakonitosti. Če si prikaz s paralelnimi koordinatami ogledujemo celostno, potem lahko na osnovi poteka črt v prikazu enostavno odkrivamo osamelce ter ločene gruče primerov. Z opazovanjem naklona črt med posameznimi sosednjimi osmi lahko odkrivamo interakcije med atributi ter pravila za ločevanje med razredi. Če pa analiziramo zgolj to, kje primeri sečejo posamezne koordinatne osi, pa je iz prikaza enostavno oceniti porazdelitev primerov pri posameznem atributu ter uspešnost atributa pri ločevanju med razredi.

Kot pri vizualizacijah s točkovnimi metodami tudi za prikaze s paralelnimi koordinatami vsekakor velja, da niso vsi enako zanimivi. V primeru, da ima analizirana zbirka podatkov večje število atributov, kot jih lahko prikažemo, se moramo najprej odločiti, katere atribute vizualizirati in katere ne. V primeru klasificiranih podatkov si lahko pri

izbiri pomagamo z različnimi merami za ocenjevanje pomembnosti atributov, s pomočjo katerih bomo izbrali atribute, ki bodo najinformativnejši za ločevanje med razredi. Druga naloga, ki prav tako bistveno odloča o zanimivosti prikaza, pa je določitev vrstnega reda prikazanih atributov. Zaradi načina vizualizacije so namreč relacije med atributi vidne zgolj med sosednjimi atributi. Čeprav lahko posameznemu primeru sledimo preko več koordinatnih osi, tega ne moremo izvesti za vse primere hkrati, zaradi česar je nemogoče odkriti, kakšna je naprimer odvisnost med prvim in zadnjim vizualiziranim atributom. Uspešnost odkrivanja interakcij med atributi ter induciranja pravil, ki vključujejo več atributov hkrati, je torej popolnoma odvisna od tega, v kakšnem vrstnem redu so atributi vizualizirani.

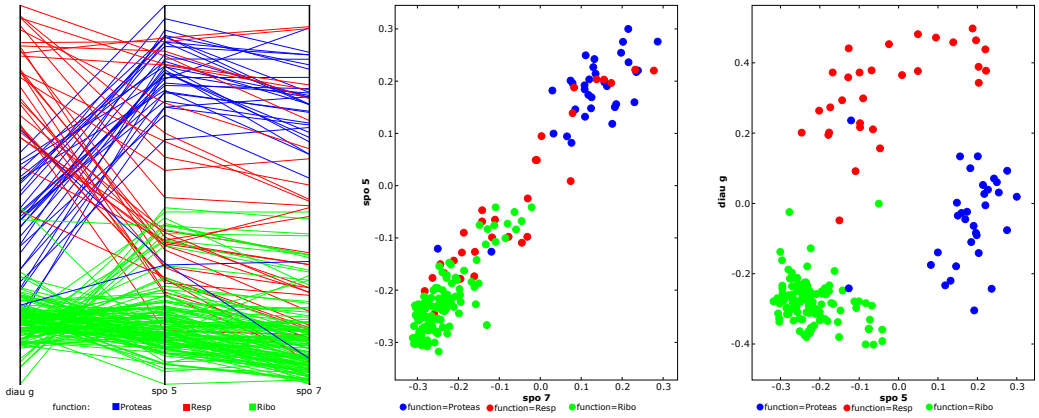
V tem poglavju bomo definirali dva pristopa k problemu izbire atributov in določitve njihovega vrstnega reda za prikaz s paralelnimi koordinatami. Prvi pristop bo namenjen iskanju prikazov, ki bodo zanimivi za opazovanje relacij med atributi – najdeni prikazi bodo vsebovali tak izbor in vrstni red atributov, da bodo prikazane glavne relacije med različnimi pari atributov. Drugi pristop bo povezan z vizualizacijsko metodo radviz in bo namenjen celostnemu pogledu na podatke. Vrstni red atributov bo tak, da bo v prikazu čim bolje opazna ločenost gruč, ki jih tvorijo primeri iz različnih razredov.

4.2 Postavitev atributov v prikazu

Preden lahko podatke vizualiziramo, je potrebno določiti vrstni red prikazanih atributov. Kot rečeno, lahko v prikazu uspešno detektiramo zgolj relacije med sosednjimi atributi, zato z določitvijo urejenosti atributov dejansko izberemo, katere relacije bomo analizirali. Dosedanje raziskave določanju urejenosti atributov niso posvečale nikakršne pozornosti – večina implementacij metode paralelnih koordinat namreč omogoča interaktivno spreminjanje položaja atributov, s čimer se naloga iskanja zanimivih relacij enostavno prenese na uporabnika. Naše mnenje je, da je ročno iskanje takih zakonitosti v podatkih počasno in nezanesljivo, zaradi česar je avtomatizacija tega procesa bistvenega pomena za njihovo uspešno identifikacijo.

4.2.1 Pomembne relacije med pari atributov

Kakšen je zanimiv vrstni red atributov in kako ga poiskati? Da bi lahko odgovorili na to vprašanje, je potrebno najprej definirati, kakšna vrsta relacij med atributi nas zanima. Ena od možnih zanimivih relacij je naprimer korelacija med atributi. Korelacijo smo že pri opisu metode omenili kot vrsto relacije, ki je v prikazu s paralelnimi koordinatami zelo dobro opazna. V primeru, da je med dvema sosednjima atributoma močna pozitivna korelacija, bodo črte med osema potekale približno paralelno; v primeru negativne korelacije pa se bodo črte križale v točki med osema. Primer dveh visoko pozitivno koreliranih atributov (*spo 5* in *spo 7*) pri podatkih o kvasovki je viden na sliki 4.1.a. Glede na to, da enako uspešno opazimo obe vrsti korelacij (pozitivno in negativno), lahko kot kriterij zanimivosti relacije med dvema atributoma uporabimo absolutno vrednost ko-



(a) (b) (c)

Slika 4.1: Korelacija med atributoma ter ločevanje med razredi kot primera dveh pomembnih relacij, ki jih je mogoče uspešno detektirati tudi v prikazu s paralelnimi koordinatami.

relacije med njima. Pri takem kriteriju bi kot zanimiv smatrali tak vrstni red atributov, v katerem so kot sosednji postavljeni atributi z visoko absolutno vrednostjo korelacije.

Opazovanje korelacij med atributi je pomembno, ker nam omogoča sklepati o povezanosti atributov in je pogosta naloga v primeru nenadzorovanega učenja, kjer želimo odkrivati nepričakovane relacije med atributi. Ureditev atributov glede na korelacije med njimi pa je pomembna tudi s stališča vizualne percepcije. Če je potek črt med različnimi osmi regularen, potem je namreč percepcija posameznih primerov lažja, posledično pa je lažja tudi detekcija gruč in osamelcev.

Kot mera zanimivosti je korelacija manj primerna v primeru nadzorovanega učenja. V tem primeru nas zanima relacija med atributi in razredom, zaradi česar korelirani atributi pravzaprav predstavljajo nepotrebno podvajanje infomacij. Če si korelirana atributa ogledamo v razsevnem diagramu na sliki 4.1.b, lahko vidimo, da oba atributa skupaj nista nič boljša pri ločevanju med razedi kot en sam atribut. V nasprotju s tem, sta naprimer atributa na sliki 4.1.c komplementarna – z opazovanjem obeh atributov hkrati bistveno uspešneje ločujemo med razredi kot z uporabo vsakega posameznega atributa. Zaradi omenjenih dualnih lastnosti med kartezičnim koordinatnim sistemom in paralelnimi koordinatami je zakonitost med takim parom atributov jasno vidna tudi v prikazu s paralelnimi koordinatami. V paralelnih koordinatah lahko informativnost vsakega posameznega atributa pri ločevanju med razredi razberemo iz položaja, kjer črte sečejo posamezno koordinatno os, dodatna informacija, ki je posledica interakcije med atributoma, pa je razvidna iz naklona črt med osema. Oglejmo si primer atributov iz razsevnega diagrama 4.1.c v paralelnih koordinatah na sliki 4.1.a. Opazimo lahko, da se naklona črt za rdeče in modre primere bistveno razlikujeta – naklon je močno pozitiven pri modrih primerih (z izjemo dveh osamelcev) in negativen pri rdečih primerih. Naklon črt torej vsebuje pomembno informacijo, s katero lahko med razredi ločujemo bolje kot z uporabo vsakega atributa posebej, vendar je to mogoče zgolj v primerih, kadar je med

atributoma pomembna interakcija.

Pari atributov, ki jih je v paralelnih koordinatah smiselno prikazati kot sosednje, so torej tisti, pri katerih je ločenost razredov, če atributa prikažemo v razsevnem diagramu, čim boljša. Za iskanje takih parov atributov lahko uporabimo VizRank, ki je bil opisan v prejšnjem poglavju. VizRank lahko oceni zanimivost vsakega od možnih razsevnih diagramov, atribute pa lahko nato v paralelnih koordinatah uredimo tako, da bodo atributi iz najbolje ocenjenih diagramov prikazani kot sosednji.

4.2.2 Algoritem za avtomatsko razvrstitev atributov

Naš cilj je torej poiskati tak vrstni red atributov, da bodo prikazane relacije med sosednjimi atributi čim bolj zanimive (glede na izbran kriterij zanimivosti). Da bi lahko trdili, da je ena urejenost boljša od druge, je potrebno definirati mero, s pomočjo katere je mogoče oceniti zanimivost posameznih urejenosti atributov. Zanimivost V_O vrstnega reda atributov O v ta namen definiramo kot povprečno zanimivost vseh sosednjih parov atributov:

$$V_O = \frac{1}{n-1} \sum_{i=1}^{n-1} V(a_i, a_{i+1}),$$

kjer je n število vizualiziranih atributov, $V(a_i, a_{i+1})$ pa zanimivost relacije med i-tim in $i+1$-im atributom (naprimer absolutna vrednost korelacije med sosednjima atributoma).

V primerih, ko je relacijo V mogoče definirati za vse pare atributov, lahko definiramo naslednji optimizacijski problem: za dani seznam atributov in relacijo V najdi tako urejenost atributov O, da bo njihova ocenjena zanimivost V_O maksimalna. Glede na to, da je predstavljeni problem ekvivalenten problemu trgovskega potnika, ki je eden najbolj znanih NP-težkih problemov, je smiselno, da zmanjšamo zahtevo po optimalnosti rešitve in se zadovoljimo s suboptimalnimi rešitvami, za katere obstajajo dobro znane in učinkovite hevristike.

Požrešni algoritem, ki smo ga implementirali za iskanje zanimivih urejenosti atributov (glej algoritem 4.1), uporablja enostavno hevristiko, ki jo je predlagal Flood [32]. Vhod v algoritem je seznam *attributePairs*, ki vsebuje informacijo o zanimivosti relacij med različnimi pari atributov, kot rezultat pa algoritem vrne urejenost atributov s čim višjo vrednostjo V_O. Algoritem deluje na naslednji način. Najprej se glede na izbrano relacijo poišče in izbere najinformativnejši par atributov. V ta delni vrstni red atributov *order* poskušamo nato na levo ali desno stran dodati enega od še neizbranih atributov. Od možnih atributov izberemo tistega, katerega relacija z enim od robnih atributov (prvim ali zadnjim) je najbolj pomembna. Izbrani atribut dodamo na ustrezno stran vrstnega reda *order* in na ta način, kolikor je mogoče, povečamo vrednost V_O. Izbor in dodajanje najprimernejšega atributa nato ponavljamo, dokler v seznam *order* ne dodamo vseh atributov. Preden algoritem vrne dobljeni vrstni red atributov, poskuša povečati oceno zanimivosti V_O s popravljanjem "prekrižanih" parov (ang. *intersecting pairs*), kot

Algoritem OrderAttributes
Vhod: seznam *attributePairs*, ki za različne pare atributov vsebuje informacijo o zanimivosti njihovih relacij
Izhod: vrstni red atributov *order*
    order = MostInformativeAttributePair(*attributePairs*)
    ponavljaj
        left je prvi atribut v seznamu *order*
        right je zadnji atribut v seznamu *order*
        bestL, valueL = FindBestNewAttributePair(*left, attributePairs, order*)
        bestR, valueR = FindBestNewAttributePair(*right, attributePairs, order*)
        if *valueL* > *valueR*:
            order = *bestL* + *order*
        else:
            order = *order* + *bestR*
    dokler niso postavljeni vsi atributi
    order = FixIntersectingPairs(*order*)

Tabela 4.1: Psevdokoda algoritma za iskanje zanimive urejenosti atributov

je to predlagal Croes [21]. Avtor je pokazal, da mora pri optimalnem vrstnem redu $(a_1, a_2, ..., a_n)$ za vsak $1 \leq p < q < n$ veljati naslednje:

$$V(a_{p-1}, a_p) + V(a_q, a_{q+1}) \geq V(a_{p-1}, a_q) + V(a_p, a_{q+1}). \tag{4.1}$$

Paru, za katerega omenjena enačba ne velja, pravimo prekrižani par. Namen funkcije FixIntersectingPairs je, da take pare (p, q) poišče in jih popravi z obračanjem podzaporedja $(a_p, ..., a_q)$; funkcija torej vrstni red spremeni v obliko:

$$(a_1, ..., a_{p-1}, a_q, a_{q-1}, ..., a_{p+1}, a_p, a_{q+1}, ..., a_n).$$

S popravkom vsakega prekrižanega para se ocena vrstnega reda poveča. Funkcija zaključi, ko so odpravljeni vsi prekrižani pari in algoritem takrat vrne končni vrstni red atributov *order*.

Čeprav bo vrstni red atributov, ki ga dobimo z uporabo algoritma 4.1, razkril številne pomembne relacije med atributi, pa bodo lahko nekatere med njimi še vedno ostale skrite. Da bi prikazali tudi te relacije, lahko nov zanimiv vrstni red atributov poiščemo tako, da iz seznama *attributePairs* odstranimo že videne pare atributov (sosednje pare v seznamu *order*) ter algoritem poženemo znova. Vrstni red, ki ga bomo v tem primeru dobili, bo (najverjetneje) imel nižjo oceno V_O in bo vseboval samo tiste zanimive relacije, ki še niso bile prikazane s prejšnjo urejenostjo atributov. Glede na to, da pri n atributih obstaja $n \cdot (n-1)/2$ različnih parov atributov in da lahko z enim prikazom prikažemo relacije med $(n-1)$ pari, lahko relacije med vsemi pari atributov prikažemo z uporabo $n/2$ različnih prikazov. Te prikaze lahko enostavno poiščemo tako, da algoritem poženemo $n/2$ krat, vsakič z manjšo množico *attributePairs*. Uporabnik si lahko nato ogleda vse te

prikaze, pri čemer največ pozornosti posveti najbolje ocenjenim prikazom, ki vsebujejo najzanimivejše relacije med atributi.

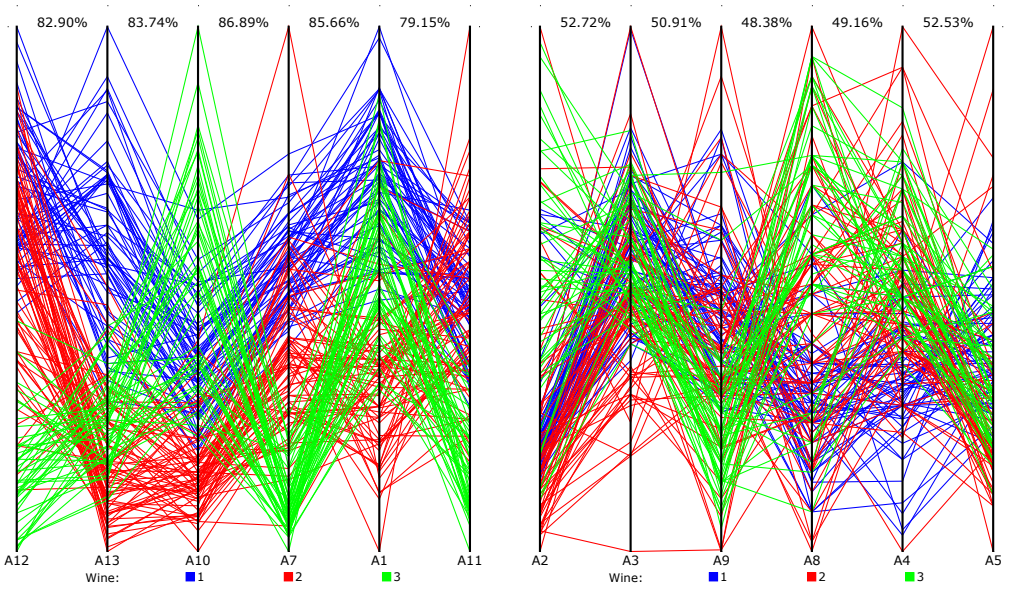
Kaj pa storiti takrat, ko je atributov preveč, da bi lahko vse vizualizirali v enem prikazu? Če bi uporabnik zanimive relacije iskal ročno, bi moral najprej izbrati različne podmnožice atributov, nato pa analizirati vsako podmnožico posebej. Taka analiza ne bi bila samo počasna, ampak tudi nepopolna, saj bi si pri tem gotovo ogledal zgolj majhno število izborov atributov in posledično zlahka spregledal številne pomembne relacije med njimi. V primeru, da bi želeli uporabiti opisani postopek, je potrebno samo malenkostno spremeniti pogoj v algoritmu 4.1 – pogoj "dokler niso postavljeni vsi atributi" enostavno nadomestimo z "dokler ni postavljenih m atributov". Kot rezultat bomo tako z algoritmom lahko poiskali $\frac{n\cdot(n-1)}{2m}$ različnih prikazov z m atributi. Prikazi bodo vsebovali različne podmnožice atributov, pri čemer bodo v vsakem od njih prikazane najpomembnejše relacije, ki še niso bile prikazane v bolje ocenjenih prikazih.

4.2.3 Primera uporabe

Kot ilustracijo uporabnosti opisanega postopka bomo predstavili dva primera uporabe; prvi primer bo na zbirki podatkov wine iz repozitorija UCI, drugi pa na podatkih o kvasovki (podrobnosti o zbirki so opisane v prejšnjem poglavju). Pri obeh zbirkah gre za klasificirane podatke, zato smo kot iskano zanimivo relacijo uporabili diskriminacijo med razredi, ki jo dosežemo z uporabo dveh atributov. Z VizRankom smo najprej ocenili razsevne diagrame z vsemi pari atributov (a_i, a_j), vsako dobljeno oceno pa smo nato uporabil v algoritmu OrderAttributes kot oceno za zanimivost relacije med atributoma.

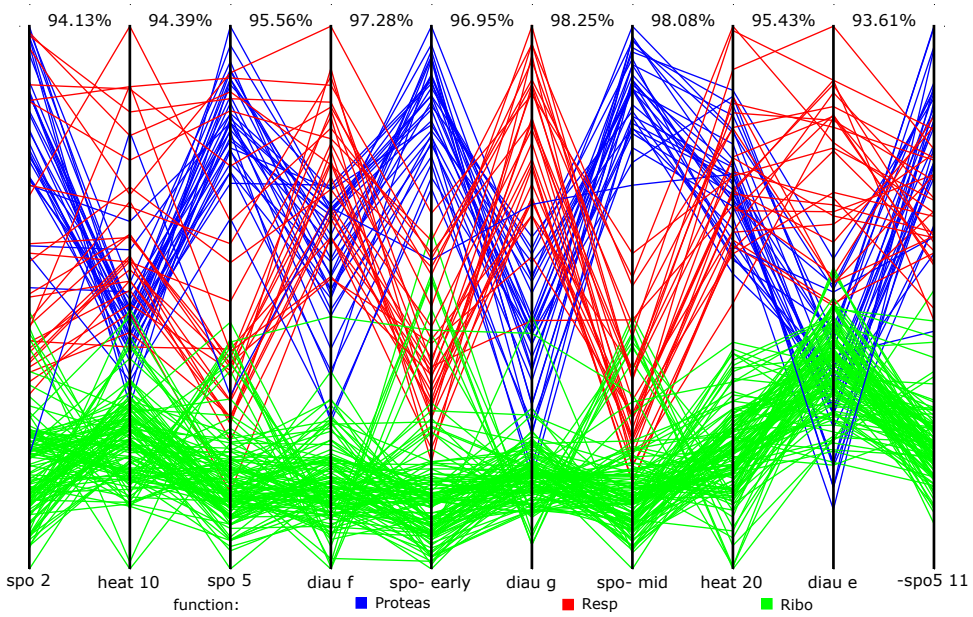
Na slikah 4.2.a ter 4.2.b sta prikazana najboljše in najslabše ocenjeni prikaz šestih atributov s paralelnimi koordinatami za podatke wine. Oceni prikazov sta 83,6% ter 50,7%. Nad vsakim parom osi je prikazana tudi VizRankova ocena za razsevni diagram z danim parom atributov. Čeprav ne moremo govoriti o odlični ločenosti razredov, je iz slike 4.2.a razvidno, da imajo primeri iz razreda *3* pri atributih *A13*, *A10* ter *A7* zelo specifično obnašanje. Primeri imajo najprej močno pozitiven, nato pa močno negativen naklon črt, s čimer lahko uspešno ločujemo med tem in ostalima dvema razredoma. Razlikovanje med razredoma *1* in *2* je iz samega naklona črt malenkost težje, ker se naklona ne razlikujeta bistveno. Vseeno lahko med razredoma opazimo razlike v naklonu med atributi *A7*, *A1* ter *A11*, samostojno pa je zelo informativen tudi atribut *A13*. V nasprotju s tem prikazom je prikaz na sliki 4.2.b popolnoma neuporaben za odkrivanje pravil za ločevanje med razredi. Črte med osmi potekajo zelo neregularno, zaradi česar je nemogoče iskati tudi osamelce ter gruče podobnih primerov.

Pri podatkih o kvasovki smo z opisanim algoritmom iskali zanimive prikaze z desetimi atributi. Ocene različnih prikazov so bile med 96,0% in 41,5%. Najboljše ocenjeni prikaz je prikazan na sliki 4.2.c. Iz njega je takoj vidno, da črte pri rdečih in modrih primerih potekajo v nasprotnih smereh, kar omogoča, da z uporabo različnih parov sosednjih atributov enostavno ločimo med obema razredoma. Lepo je vidna tudi lastnost, na osnovi katere lahko ločimo citoplazemske ribosome (zeleni primeri) od ostalih razredov



(a) 83,6%

(b) 50,7%



(c) 96,0%

Slika 4.2: Primer odkrivanja pomembnih relacij na dveh problemskih domenah. Najboljši in najslabši prikaz s paralelnimi koordinatami na podatkih wine (a,b) ter najboljši prikaz za podatke o kvasovki (c).

– ti primeri imajo namreč pri vseh prikazanih atributih majhne vrednosti. Pri tej skupini primerov je zanimivo tudi, da obstaja majhna podskupina primerov, katerih vrednosti se razlikujejo od običajnih vrednosti pri tem razredu. Ta odstopanja so še posebej izrazita pri procesu sporulacije, oziroma pri atributih s predpono “spo-”. Pomembna prednost

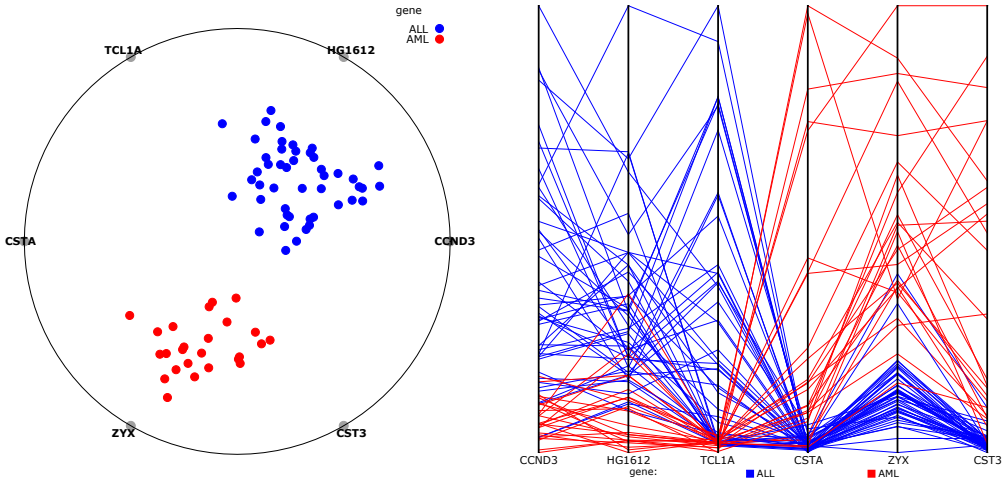
prikaza s paralelnimi koordinatami v primerjavi z razsevnimi diagrami je, da lahko pri paralelnih koordinatah zlahka ugotovimo, da gre za isto podskupino primerov, ki konsistentno odstopajo od drugih primerov. Če bi si podatke ogledovali v razsevnih diagramih, bi imeli namreč težave z ugotavljanjem, ali podskupina primerov v različnih diagramih vsebuje iste ali različne primere. V danem prikazu s paralelnimi koordinatami enostavno opazimo tudi nekatere osamelce. En izmed njih je naprimer gen *YDR069C* iz razreda proteasomov, ki smo ga omenili že pri analizi najboljšega razsevnega diagrama, za katerega vidimo, da ima popolnoma netipične vrednosti pri kar osmih prikazanih atributih.

4.3 Urejanje atributov z uporabo metode radviz

Pri urejanju atributov na osnovi korelacije med atributi smo kot pomembno prednost takega izbora atributov omenili regularen potek črt med posameznimi sosednjimi osmi, ki omogoča boljšo percepcijo primerov ter lažje sledenje posameznim primerom preko več koordinatnih osi. Taka ureditev atributov pa, kot rečeno, ni najprimernejša pri nadzorovanem učenju, saj tak prikaz kljub regularnosti črt ne omogoča sklepanja o tem, kako lahko ločimo med različnimi razredi.

Kljub temu, da sta korelacija med atributi in diskriminacija med razredi bistveno različna kriterija, pa vseeno obstaja tak način izbiranja in urejanja atributov, da je potek črt kar se da regularen in da je iz prikaza hkrati mogoče čim bolje ločiti med različnimi razredi. Tako ureditev atributov lahko najdemo s pomočjo vizualizacijske metode radviz ter metodo VizRank. Za atribute, ki nastopajo v zanimivih projekcijah radviz, ki jih najdemo z uporabo VizRanka, veljata namreč dve pomembni lastnosti. Večina sosednjih atributov na krožnici ima relativno podobne vrednosti, saj opravljajo podobno nalogo, kot je naprimer privlačiti k sebi primere istega razreda. Hkrati pa ti atributi niso izbrani glede na običajno korelacijo med njimi, ampak glede na njihovo sposobnost ločevanja med različnimi razredi. Če torej atribute v paralelnih koordinatah izberemo in uredimo v istem vrstnem redu kot so prikazani v neki zanimivi projekciji radviz, bomo s tem dosegli visoko regularnost v poteku črt med osmi, hkrati pa bo na osnovi prikazanih atributov mogoče uspešno ločevati med različnimi razredi.

Primer uporabe takega načina izbiranja in urejanja atributov pri podatkih o levkemiji je prikazan na sliki 4.3. Slika 4.3.a prikazuje najboljšo najdeno projekcijo radviz s šestimi atributi, na sliki 4.3.b pa je prikazana vizualizacija istih atributov v prikazu s paralelnimi koordinatami. Kot lahko sklepamo tudi iz prikaza radviz, imajo prvi trije atributi velike vrednosti pri razredu *ALL*, drugi trije atributi pa pri razredu *AML*. Vsaka od obeh vizualizacijskih metod ima svoje prednosti. V prikazu radviz je naprimer bolje opazna dobra ločenost med razredoma, prikaz s paralelnimi koordinatami pa omogoča opazovanje dejanskih vrednosti posameznih primerov. Pri paralelnih koordinatah lahko tako opazimo dva primera iz razreda *ALL*, ki pri atributu *ZYX* (gen *zyxin*) pomembno odstopata od ostalih primerov iz istega razreda in ju je vredno podrobneje raziskati. Glede na prednosti vsake od metod je pri analizi podatkov vsekakor smiselno hkrati uporabljati obe metodi.



(a) (b)

Slika 4.3: Vizualizacija istih atributov pri podatkih o levkemiji z metodo radviz (a) in metodo paralelnih koordinat (b). Ločenost razredov v prikazu radviz omogoča tudi ločevanje med razredi v prikazu s paralelnimi koordinatami.

Poglavje 5

Ocenjevanje in rangiranje mozaičnih diagramov

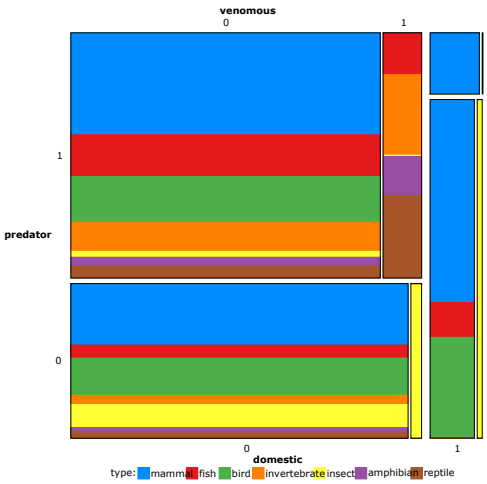
Mozaični diagram je vizualizacijska metoda, namenjena vizualizaciji diskretnih oziroma diskretiziranih zveznih atributov. Podatki so v tem primeru vizualizirani kot pravokotne celice, katerih velikost ponazarja delež primerov z danim naborom vrednosti atributov. V primeru nadzorovanega učenja lahko posamezne celice pobarvamo tako, da deleži posameznih barv v celici ponazarjajo porazdelitev primerov po razredih.

Kot je pokazala analiza vizualizacijskih metod v razdelku 2.5, je mozaični diagram odlična vizualizacijska metoda za odkrivanje znanja v podatkih. Pri nenadzorovanem učenju lahko z njo enostavno odkrivamo odvisnosti med atributi, pri nadzorovanem učenju pa identificiramo relevantne atribute ter odkrivamo pravila za ločevanje med razredi. Glede na to, da lahko z metodo hkrati vizualiziramo do štiri atribute, pa se kot pri ostalih večdimenzionalnih vizualizacijskih metodah tudi tu pojavi vprašanje, katere atribute vizualizirati in kako, da bo dobljeni prikaz omogočil nek nov, jasen vpogled v vsebovane zakonitosti.

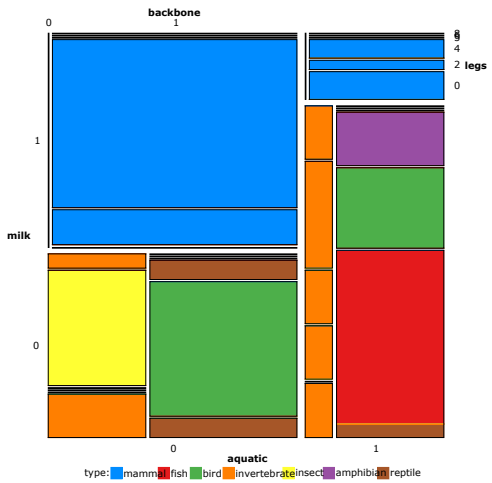
V tem poglavju se bomo ukvarjali z različnimi načini, s katerimi lahko pri analizi klasificiranih podatkov poiščemo zanimive mozaične diagrame. Predstavili bomo postopek, s katerim lahko vsak mozaični diagram ocenimo s stališča vizualiziranih atributov, nato pa še s stališča primernosti same postavitve teh atributov v diagramu.

5.1 Algoritem za ocenjevanje zanimivosti mozaičnih diagramov

Glavna faktorja, ki vplivata na zanimivost prikaza z mozaičnim diagramom, sta predvsem izbor podmnožice atributov ter vrstni red, v katerem so ti atributi v prikazu vizualizirani. V nadaljevanju bomo najprej opisali načine, s katerimi lahko ocenimo zanimivost različnih podmnožic atributov. Na osnovi te ocene lahko ločimo med bolj in manj zanimivimi kom-



$IG = 0,31$, $1 - D = 0,08$
$MDL = -0,02$, $\chi^2 = 0,92$
Interakcije = 0,03, $\overline{P} = 27,14\%$
(a)



$IG = 2,34$, $1 - D = 0,67$
$MDL = 0,77$, $\chi^2 = 42,32$
Interakcije = -1,06, $\overline{P} = 97,93\%$
(b)

Slika 5.1: Primer neinformativnega (a) ter informativnega (b) mozaičnega diagrama za domeno zoo.

binacijami atributov. Sledil bo opis algoritma, s katerim lahko za vsak izbor atributov ocenimo zanimivost prikazov, ki jih je mogoče dobiti s postavitvijo teh atributov v različnih možnih vrstnih redih. Vrstni red prikazanih atributov namreč določa, kakšen bo relativni medsebojni položaj posameznih celic v prikazu in bistveno vpliva na to, kako uspešno bomo zmožni detektirati pravila, ki vključujejo več celic hkrati.

5.1.1 Izbor atributov

Kot pri ostalih vizualizacijskih metodah je tudi pri mozaičnem diagramu zanimivost diagrama primarno odvisna od izbora vizualiziranih atributov. Z izborom vizualiziranih atributov primere, podobno kot pri odločitvenih drevesih, razbijemo na disjunktne množice (v diagramu so le-te prikazane kot pravokotne celice), glede na različne vrednosti atributov pri teh primerih. Ne glede na število vizualiziranih atributov so najmanj zanimivi prikazi gotovo tisti, pri katerih je v posameznih celicah porazdelitev primerov po razredih približno enaka apriorni verjetnostni porazdelitvi razredov. Iz takih prikazov lahko sklepamo zgolj to, da so prikazani atributi neuporabni za ločevanje med razredi. Izbori atributov, ki bi jih želeli najti in vizualizirati, so tisti, pri katerih se v posameznih celicah porazdelitev primerov po razredih čim bolj razlikuje od apriorne porazdelitve. V idealnem primeru je ta porazdelitev taka, da so v posamezni celici zgolj primeri enega razreda. Taki diagrami prikazujejo pomembno odvisnost med razredom in prikazanimi atributi ter omogočajo vizualno induciranje enega ali več pravil za ločevanje med razredi.

Kot primer si oglejmo dva mozaična diagrama za domeno zoo, ki sta prikazana na

sliki 5.1. Izbor atributov na sliki 5.1.a je popolnoma neinformativen, saj so v večini celic prisotni primeri vseh vrst živali. V nasprotju s tem so na sliki 5.1.b skoraj vse celice enobarvne, kar omogoča enostavno izpeljavo pravil za določanje vrste živali.

Za ocenjevanje zanimivosti različnih izborov atributov lahko definiramo različne kriterije. Okvirno jih lahko razdelimo v tri skupine:

- **Mere nečistoče ter asociativnosti.** Podobno kot mere nečistoče uporabljamo za ocenjevanje koristnosti posameznih atributov, jih je mogoče uporabiti tudi za ocenjevanje kombinacij atributov. Iz k atributov lahko sestavimo nov atribut A, pri katerem je množica njegovih vrednosti kartezični produkt vrednosti posameznih atributov A_1, ..., A_k, torej $A = A_1 \times ... \times A_k$. Za ocenitev takega atributa A lahko uporabimo poljubno mero nečistoče. Ena od pogosto uporabljenih mer, ki temelji na količini informacije, je informacijski prispevek [53] (ang. *information gain*), ki je definiran kot:

  $$\text{InfoGain}(A) = H(C) - H(C|A),$$

  pri čemer je $H(C)$ entropija razreda, $H(C|A)$ pa pogojna entropija razreda pri dani vrednosti atributa A. Slabost te mere je, da ocena atributa z večanjem števila vrednosti zgolj narašča, zaradi česar so bolje ocenjeni atributi z velikim številom vrednosti. Ker želimo najti zanimive kombinacije atributov, ki hkrati vsebujejo čim manjše število vrednosti, je primerneje uporabiti mero razdalje dogodkov $1 - D(A)$ [78]. Ta mera je definirana kot kvocient informacijskega prispevka atributa A ter entropije produkta vrednosti razreda C in atributa A:

  $$1 - D(A) = \frac{\text{InfoGain}(A)}{H(A, C)}.$$

  Zelo primerno mero lahko izpeljemo tudi z uporabo principa najkrajšega opisa (ang. *minimum description length, MDL*) [89]. Glede na ta kriterij je atribut tem bolj pomemben, čim bolj je z njim mogoče zakodirati informacijo o razredu pri posameznih vrednostih razreda. Rezultati raziskav kažejo, da je ta mera najprimernejša glede pristranskosti pri ocenjevanju večvrednostnih atributov [65].

  Za ocenitev posameznega izbora atributov lahko uporabimo tudi različne statistične mere asociativnosti, s katerimi izračunamo odvisnost med razredom ter konstruiranim atributom A. Ena najpogosteje uporabljenih mer asociativnosti je naprimer Pearsonov χ^2, ki ga izračunamo na osnovi razlike med pričakovanim in dejanskim številom primerov posameznega razreda v vsaki celici mozaičnega diagrama. Večji kot je χ^2 za neko kombinacijo atributov, pomembnejša je odvisnost med atributi in razredom. Sorodna mera je tudi Cramerjev ϕ, ki z upoštevanjem števila primerov v podatkih ter kardinalnosti atributa in razreda normalizira vrednost χ^2 na interval med 0 in 1 in jo lahko tako interpretiramo kot korelacijo med atributom in razredom.

- **Interakcije med atributi.** Atributi pogosto niso medsebojno neodvisni glede na razred. En tip odvisnosti predstavljajo *soodvisni* atributi, pri katerih gre za podvajanje iste informacije – soodvisna atributa A in B naprimer povesta manj o razredu C, kot bi pričakovali, če bi sešteli informacijo, ki jo podajata posamično. Poleg soodvisnih poznamo tudi *sodejavne* atribute, pri katerih o razredu izvemo več, če oba atributa upoštevamo hkrati. Klasičen primer sodejavnih atributov imamo v primeru, ko je razred C definiran kot $C = A \textbf{ xor } B$. V tem primeru sta atributa posamično popolnoma neinformativna, če jih obravnavamo hkrati, pa omogočata točno napoved vrednosti razreda.

  Soodvisnost in sodejavnost atributov je Jakulin [56, 57, 58] pojasnil s konceptom interakcij med atributi. Interakcijski prispevek med dvema atributoma A in B glede na razred C je definiral kot:

  $$I(A;B;C) = IG(A \times B;C) - IG(A;C) - IG(B;C),$$

  pri čemer je mera $IG(\cdot)$ informacijski prispevek, $A \times B$ pa nov atribut sestavljen kot kartezični produkt atributov A in B. Vrednost interakcijskega prispevka je pozitivna v primeru sodejavnosti in negativna v primeru soodvisnosti atributov.

  Različne izbore atributov je na osnovi interakcijskega prispevka smiselno rangirati od najbolj sodejavnih kombinacij atributov do najbolj soodvisnih. Pri tem se je potrebno zavedati, da ta kriterij ocenjuje zgolj *dodatno* informacijo, ki jo pridobimo v primeru, da atribute združimo v nov atribut, zaradi česar najbolje ocenjeni izbori atributov najpogosteje niso tisti z najboljšo ločenostjo razredov. Čeprav interakcijski prispevek rangira izbore atributov bistveno drugače kot ostale mere, ga je vseeno smiselno uporabiti, ko so v podatkih prisotni močno sodejavni atributi.

- **Napovedna točnost.** Izbore atributov lahko rangiramo tudi glede na napovedno točnost, ki jo z njimi dosežemo. En od načinov za izračun točnosti je, da z izbranimi atributi zgradimo enostavno odločitveno drevo, pri čemer v istih nivojih drevesa povsod uporabimo isti atribut. Rezultat bo ekvivalenten tudi v primeru, da iz izbranih atributov sestavimo nov atribut in z njim zgradimo eno-nivojsko odločitveno drevo. Porazdelitev primerov po razredih, ki jo na ta način dobimo v listih drevesa, je enaka porazdelitvi, ki jo lahko vidimo v mozaičnem diagramu z istimi atributi. Napovedno točnost takega drevesa lahko izračunamo naprimer z uporabo 10-kratnega prečnega preverjanja, pri tem pa lahko uporabimo poljubno cenilno funkcijo. V primeru, da za ocenjevanje uporabimo klasifikacijsko točnost, je dobljeno oceno mogoče interpretirati kot uteženo povprečno verjetnost večinskega razreda v posameznih celicah diagrama. Če želimo v oceni upoštevati dejanske verjetnostne porazdelitve razredov v posameznih celicah, pa je primernejša funkcija povprečna verjetnost pravilne klasifikacije $\overline{P}$.

	InfoGain	$1-D$	MDL	χ^2	Interakcije	$\overline{P}$
InfoGain	1	0,881	0,989	0,985	-0,646	0,995
$1-D$	0,881	1	0,930	0,891	-0,589	0,889
MDL	0,989	0,930	1	0,985	-0,636	0,990
χ^2	0,985	0,891	0,985	1	-0,594	0,992
Interakcije	-0,646	-0,589	-0,636	-0,594	1	-0,627
$\overline{P}$	0,995	0,889	0,990	0,992	-0,627	1

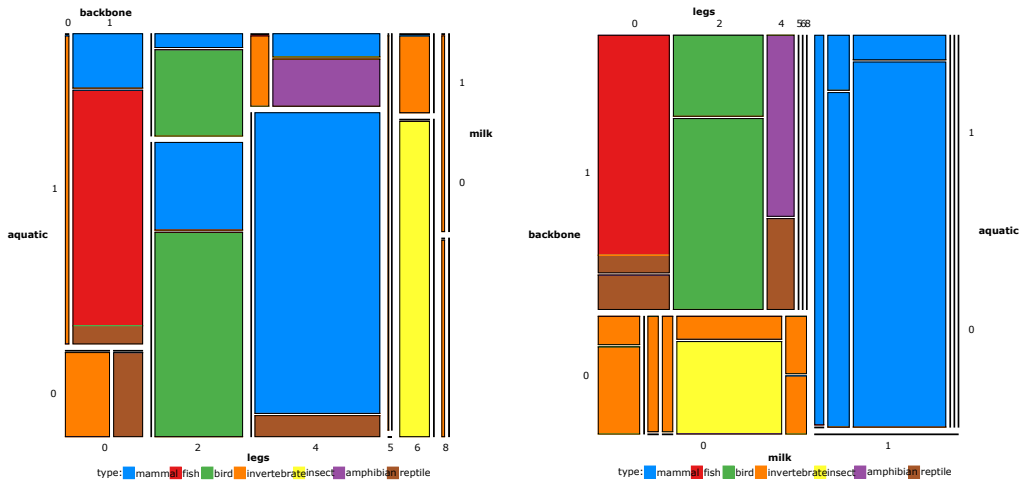
Tabela 5.1: Korelacije med ocenami različnih kriterijev za ocenjevanje zanimivosti različnih kombinacij atributov. Pri izračunu so bili upoštevani vsi možni izbori dveh, treh ter štirih atributov pri UCI domeni voting. Testirani kriteriji zanimivosti so informacijski prispevek (InfoGain), mera razdalje dogodkov ($1-D$), princip najkrajšega opisa (MDL), Pearsonov χ^2, interakcijski prispevek ter povprečna verjetnost pravilne klasifikacije ($\overline{P}$).

Z izjemo interakcij je večina opisanih mer medsebojno podobnih, zaradi česar so močno korelirane tudi dobljene ocene zanimivosti, ki jih z uporabo teh mer dobimo pri različnih izborih atributov. Za primerjavo smo izvedli poskus, pri katerem smo z uporabo opisanih mer ocenili vseh 2.516 različnih mozaičnih diagramov z dvema do štirimi atributi pri domeni voting iz repozitorija UCI. Korelacije med dobljenimi ocenami so prikazane v tabeli 5.1. Po pričakovanjih je iz tabele takoj razvidno, da interakcijski prispevek definira zanimivost izbora atributov bistveno drugače kot ostale mere. Ker v domeni ni izrazitih pozitivnih interakcij, je med merami celo relativno močna negativna korelacija. Razlog za to je, da interakcijski prispevek kombinacije informativnih atributov zaradi njihove soodvisnosti oceni kot nezanimive, medtem ko ostale mere take izbore atributov ocenijo kot zanimive, saj kljub odvisnosti dobro ločujejo med razredi.

Katero mero je torej najprimerneje izbrati za ocenjevanje zanimivosti različnih izborov atributov? Kadar bi iz prikazov želeli ločevati med razredi, interakcijski prispevek vsekakor ni najprimernejša mera. Ostale mere so mesebojno močno korelirane, zato izbira mere ne bo bistveno vplivala na rangiranje. Omenjeni statistični meri (Pearsonov χ^2 ter Cramerjev ϕ) sta malenkost manj primerni od ostalih mer, ker ne ocenjujeta same čistosti posameznih celic, ampak zgolj odstopanje od apriorne porazdelitve razreda. Če je pomembna interpretabilnost ocene, potem je verjetno najprimernejša mera povprečna verjetnost pravilne klasifikacije $\overline{P}$. V primeru, ko imajo nekateri atributi večje število vrednosti, pa sta smiselni izbiri mera razdalje dogodkov $1-D$ ter mera MDL, saj znata izbore atributov primerno kaznovati glede na kardinalnost atributov.

5.1.2 Vrstni red atributov in njihovih vrednosti

Izbor atributov nam (ne glede na to, v kakšnem vrstnem redu jih vizualiziramo) določa, kakšna bo zanimivost prikazanega mozaičnega diagrama. To je v nasprotju z metodo radviz, pri kateri lahko pri različnih postavitvah istega izbora atributov generiramo različno zanimive prikaze. S spreminjanjem vrstnega reda atributov pri mozaičnem diagramu ploščine posameznih celic sicer ostajajo enake, spremeni pa se njihov položaj v prikazu.



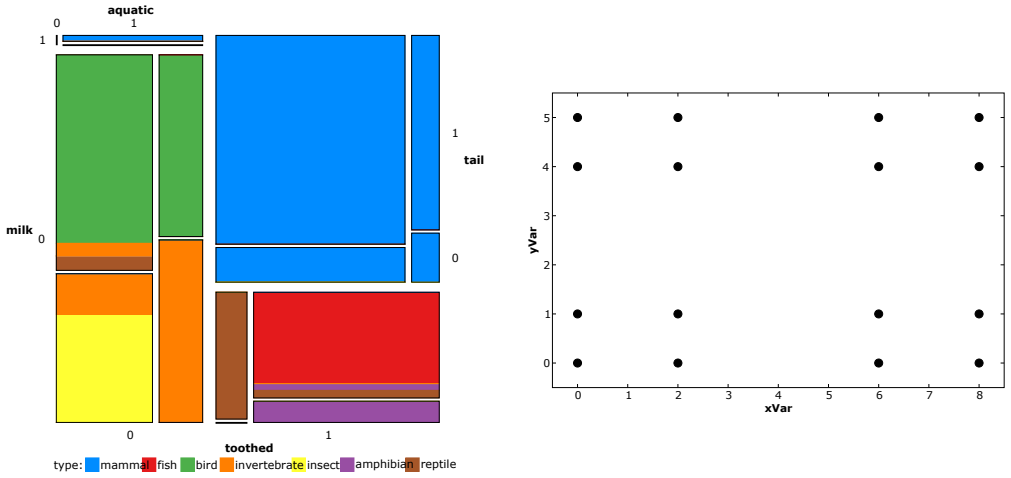
(a) $\overline{P} = 41,37\%$ (b) $\overline{P} = 63,73\%$

Slika 5.2: Primer dveh mozaičnih diagramov z istim izborom atributov za domeno zoo.

Položaj posameznih celic v prikazu bistveno vpliva na interpretabilnost prikaza. Primeri enega razreda namreč običajno ne ležijo zgolj v eni sami, ampak v več celicah. Če te celice ležijo v različnih delih prikaza, potem je vizualna zaznava skupne lastnosti, ki opisuje te primere, težja, kot če celice ležijo ena zraven druge. Da bi analitiku olajšali percepcijo zakonitosti, je torej smiselno poiskati tako urejenost atributov, pri kateri so celice z istim večinskim razredom lepo grupirane in kar se da ločene od celic z drugim večinskim razredom. V primeru, ko imamo opravka z vizualizacijo nominalnih atributov, pri katerih ne obstaja neka naravna ureditev vrednosti, je poleg različnega vrstnega reda atributov smiselno preučiti tudi različni vrstni red njihovih vrednosti.

Slika 5.2 prikazuje dva mozaična diagrama z istim izborom atributov. V levem diagramu razporeditev atributov ni najprimernejša. Vidimo naprimer, da celice, ki vsebujejo zgolj primere sesalcev, ležijo v različnih delih diagrama, zato je zelo težko definirati pravilo, s katerim bi te primere lahko opisali. Podobno težavo imamo tudi pri celicah, ki vsebujejo primere ptic in nevretenčarjev. Bistveno drugačna je razporeditev celic v desnem diagramu. V tem primeru ležijo vse celice s primeri sesalcev skupaj, kar omogoča trivialno detekcijo pravila za ločevanje sesalcev od ostalih skupin živali. Zelo enostavno so opazne tudi lastnosti, ki opisujejo nevretenčarje ter ptice. Dodatna zanimivost, ki je v levem diagramu ni mogoče opaziti, je podobnost med insekti in nevretenčarji. Insekti so zgolj posebna podvrsta nevretenčarjev, ki živi na kopnem in ima šest nog. Z ustreznim vrstnim redom atributov je torej poleg pravil mogoče detektirati tudi podobnosti ter razlike med različnimi razredi.

Zanimivost mozaičnih diagramov, ki jih dobimo z različnim vrstnim redom prikazanih atributov, je mogoče oceniti na različne načine. Pristop, ki smo ga izbrali, temelji na podobnem principu, kot smo ga uporabili za ocenjevanje točkovnih prikazov. Za vsako postavitev izbranih atributov izvede naš algoritem naslednja dva koraka:



(a) (b)

Slika 5.3: Pretvorba mozaičnega diagrama (a) v ustrezni razsevni diagram (b). Razdalja med primeri v razsevnem diagramu je določena glede na razdaljo med celicami v mozaičnem diagramu – razdalja med najbližjimi sosednjimi celicami je ena, med najbolj oddaljenimi sosednjimi celicami pa je v tem primeru štiri enote.

1. **Pretvorba mozaičnega diagrama v razsevni diagram.** Za dani vrstni red izbranih atributov generiramo ustrezni mozaični diagram ter ga nato pretvorimo v razsevni diagram tako, da vsako celico diagrama predstavimo z eno točko. V nasprotju z običajnimi primeri v razsevnem diagramu ta točka ne pripada enemu samemu razredu. Razred primera je namesto tega določen z verjetnostno porazdelitvijo, ki je enaka porazdelitvi po razredih za pripadajočo celico mozaičnega diagrama. Pomembno pri pretvorbi je tudi, kako primerom v razsevnem diagramu določimo njihov položaj v prikazu. Kot je razvidno iz mozaičnega diagrama na sliki 5.3, so razdalje med različnimi sosednjimi celicami različne – razdalje so največje pri vizualizaciji prvega (*toothed*) in najmanjše pri vizualizaciji zadnjega atributa (*tail*). To lastnost je smiselno upoštevati tudi v razsevnem diagramu. Razdalje med primeri so tako določene glede na to, kdaj je atribut v mozaičnem diagramu vizualiziran – pri vizualizaciji n atributov sta tako najbližja primera z različno vrednostjo i-tega vizualiziranega atributa narazen $n-i+1$ enot. Za mozaični diagram na sliki 5.3.a, je položaj točk v razsevnem diagramu, ki ga na ta način dobimo, prikazan na sliki 5.3.b.

2. **Ocenjevanje zanimivosti razsevnega diagrama.** Za ocenjevanje zanimivosti dobljenega razsevnega diagrama lahko uporabimo postopek VizRank, opisan v razdelku 3.1, le da je potrebno pred tem nekoliko prilagoditi učni algoritem k-najbližjih sosedov. Enostavna sprememba, ki jo mora znati algoritem k-NN upoštevati, je, da primeri ne pripadajo zgolj enemu razredu, ampak da imamo za vsak primer podano verjetnostno porazdelitev razredov. Dodatna sprememba, ki jo je pri ocenjevanju smiselno vpeljati, je, da vsakemu primeru v razsevnem diagramu določimo utež glede na število primerov, ki so prisotni v

pripadajoči celici mozaičnega diagrama – večjim celicam določimo večje uteži, manjšim celicam manjše. Ko k najbližjih primerov nato sodeluje pri predikciji razreda, se tako poleg uteževanja glede na oddaljenost od primera upošteva še utež, ki ponazarja pomembnost primera.

Iz napovedanih verjetnosti lahko za izračun končne ocene zanimivosti določenega vrstnega reda atributov uporabimo poljubno cenilno funkcijo. Kot pri točkovnih vizualizacijskih metodah tudi v tem primeru predlagamo uporabo povprečne verjetnosti pravilne klasifikacije $\overline{P}$, saj upošteva negotovost napovedi in je hkrati enostavno interpretabilna.

Z opisanim algoritmom je mogoče pri nekem izboru atributov oceniti zanimivost različnih permutacij teh atributov ter v primeru nominalnih atributov tudi različnih permutacij njihovih vrednosti. Analitik si lahko nato ogleda zgolj prikaz z najbolje ocenjeno permutacijo, saj naj bi ta omogočal najlažje vizualno induciranje pravil. Dokaz smiselnosti algoritma sta mozaična diagrama na sliki 5.2, ki prikazujeta najslabše ter najbolje ocenjeno permutacijo prikazanih atributov.

5.2 Primeri uporabe

V nadaljevanju so predstavljeni različni primeri uporabe opisanega algoritma za izbor in urejanje atributov za prikaz z mozaičnim diagramom.

5.2.1 Domena monks 1

Monks 1 je sintetična domena s šestimi diskretnimi atributi ($a1$ do $a6$). Binaren razred predstavlja koncept $(a1 = a2) \vee (a5 = 1)$.

Najbolje ocenjeni mozaični diagram je na sliki 5.4.a. Trivialno je opazen del koncepta $a5 = 1$, uspešno pa lahko detektiramo tudi prvi del koncepta, ki vključuje hkratno upoštevanje vrednosti atributov $a1$ in $a2$. Glede na to, da je mogoče detektirati pravila, ki vključujejo relacije med atributi (kot naprimer $a1 = a2$), je torej opisni jezik, s katerim lahko v mozaičnem diagramu opišemo hipoteze, bolj kot izjavnemu računu blizu predikatnemu računu prvega reda.

5.2.2 Domena car

Podatki car so primer sintetične domene, ki vsebuje šest diskretnih atributov, s katerimi so opisane lastnosti avtomobilov, razred pa predstavlja oceno primernosti avtomobila. Zbirka je bila razvita z uporabo enostavnega hierarhičnega odločitvenega modela, ki je služil kot demonstacija sistema DEX [6]. V strojnem učenju je bila domena kasneje uporabljena za evaluacijo orodja HINT, ki je uspel iz podatkov popolnoma rekonstruirati originalni hierarhični model.

Na sliki 5.4.b je najbolje ocenjeni diagram s štirimi atributi. Iz njega je takoj razvidno, da so nesprejemljivi vsi dvosedi ter avtomobili z nizko varnostjo. Primernost preostalih avtomobilov je nato odvisna tudi od cene avtomobila (atribut *buying*) ter

stroškov vzdrževanja. Najbolje so ocenjeni avtomobili, ki so zelo varni, srednje ali nizke cene ter z njimi hkrati ni previsokih stroškov vzdrževanja. Številne celice v diagramu vsebujejo primere različnih razredov, saj je domena takšna, da je za popolno ločenost razredov potrebno hkrati upoštevati vrednosti vseh šestih atributov.

5.2.3 Domeni breast-cancer-winsconsin ter wdbc

Obe zbirki vsebujeta podatke o pacientkah s tumorjem na prsih, zbranih na univerzi v Winsconsinu. Domena breast-cancer-winsconsin vsebuje 683 primerov, opisanih z devetimi atributi, ki predstavljajo ročno ocenjene lastnosti celic iz tumornega tkiva. Vsak atribut ima vrednosti od ena do deset – vrednost ena ustreza normalnemu stanju, vrednost deset pa najbolj nenormalnemu stanju. Domena wdbc vsebuje 569 primerov opisanih z 20 atributi, pri čemer so vrednosti primerov določene z avtomatsko analizo slik celičnih jeder v tumornem tkivu. Atributi opisujejo različne karakteristike celičnih jeder, kot so naprimer polmer, ploščina, tekstura ter simetričnost jedra. Pri obeh zbirkah je razred diagnoza pacientke – tumor je lahko benigni (vrednost B) ali maligni (vrednost M).

Najboljši mozaični diagram za domeno breast-cancer-winsconsin je na sliki 5.4.c. Vizualizirana sta zgolj dva atributa, saj vsi atributi vsebujejo deset vrednosti, zaradi česar pride sicer do prevelike razdrobljenosti primerov. Po pričakovanjih lahko iz diagrama ugotovimo, da je za žensko najbolje, če so vizualne lastnosti celic kar se da normalne – pri vseh pacientkah, ki imajo pri obeh vizualiziranih atributih vrednost ena (to je skoraj polovica vseh žensk), gre le za benigno tvorbo. Večje kot je odstopanje od normalnosti, večja je verjetnost, da je tvorba maligna. V primerjavi z odločitvenimi drevesi ima mozaični diagram v tem primeru pomembno prednost. Čeprav so primeri razdeljeni v kar 100 celic, lahko te celice vizualno združujemo glede na podobnost s sosednjimi celicami. Tako lahko naprimer zaključimo, da so ženske z benignim tkivom v spodnjem levem delu diagrama, tiste z malignim tkivom pa v zgornjem desnem delu diagrama. Poleg tega je na osnovi sosednjih celic mogoče sklepati tudi o potencialnih napakah v diagnozi. Pri vrednosti *Bare Nuclei* = 8 sta naprimer med samimi rdečimi celicami dve modri celici z zgolj enim primerom. Čeprav je seveda možno, da je diagnoza pravilna, sta taka primera vsekakor vredna podrobnejše analize.

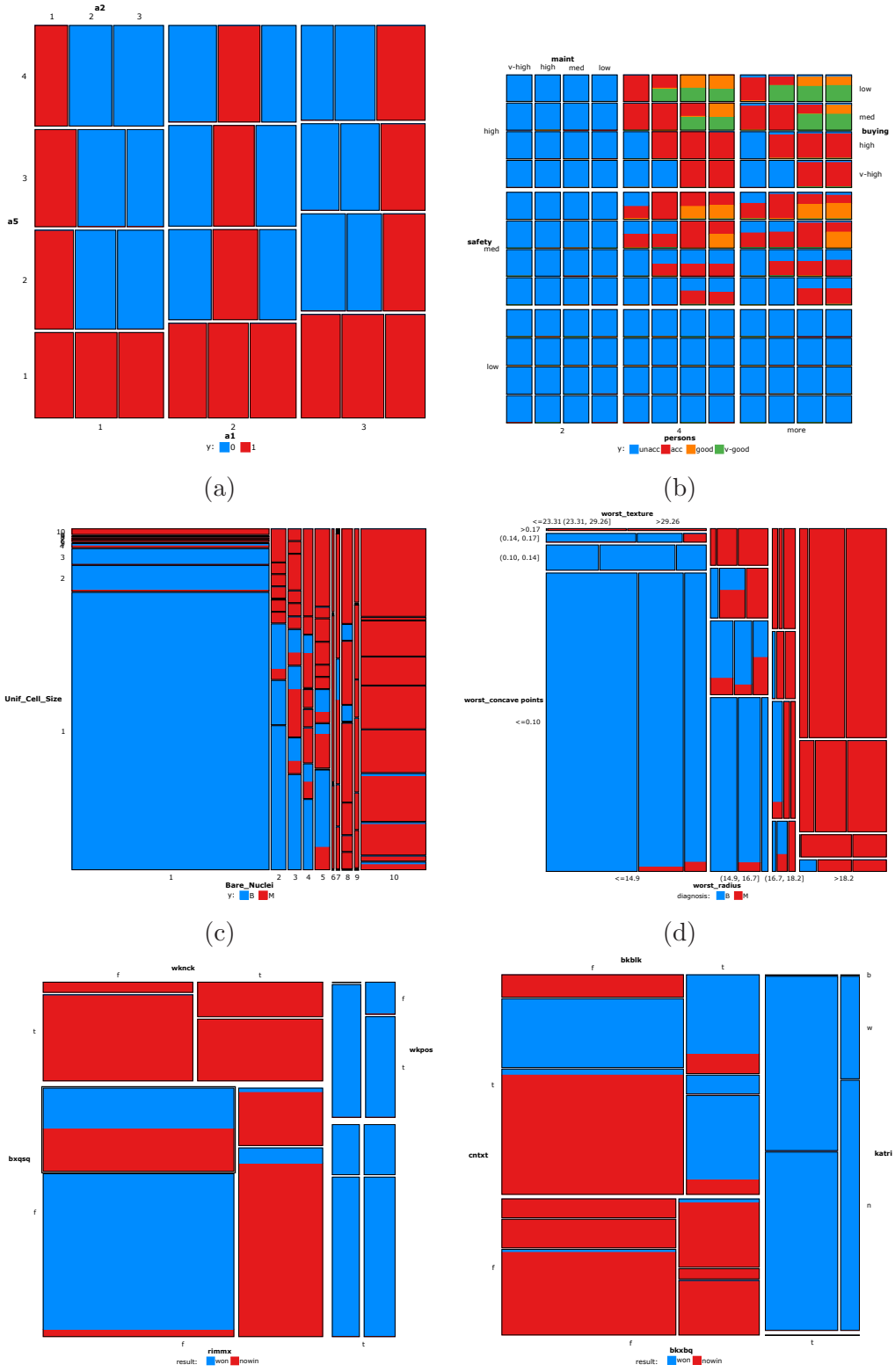
Slika 5.4.d prikazuje najboljši diagram za domeno wdbc. Ponovno lahko sklepamo, da je pri večjih vrednostih vizualiziranih atributov verjetnost malignega tumorja večja. Iz diagrama lahko tudi vidimo, da so atributi medsebojno pozitivno korelirani – največje celice so namreč v spodnjem levem ter zgornjem desnem delu diagrama.

5.2.4 Domena krkp

Domena predstavlja šahovsko končnico, pri kateri ima en igralec kralja in trdnjavo, drugi pa kralja in kmeta [95]. Beli kmet je na *a7* in lahko v eni potezi dobi kraljico, črna trdnjava ter oba kralja pa sta v različnih veljavnih pozicijah. Zbirka vsebuje 3196 primerov (pozicij) ter 36 atributov, ki opisujejo stanje na deski. Razred ima dve vrednosti: beli lahko zmaga

(*won*) ter beli ne more zmagati (*no-win*).

Najboljši mozaični diagram za celotno zbirko je na sliki 5.4.e. Lepo je razvidno, da beli gotovo zmaga takrat, ko lahko ujame črno trdnjavo ($rimmx = t$), izgubi pa takrat, ko ena od črnih figur blokira *a8*. Od preostalih pozicij je največja negotovost glede zmage pri kombinaciji vrednosti, ki je v diagramu označena s pravokotnikom. Če podrobneje analiziramo zgolj te primere, lahko ponovno odkrijemo zanimiv mozaični diagram, ki je prikazan na sliki 5.4.f. Taka kombinacija vizualizacije ter analize podatkov spominja na interaktivno gradnjo odločitvenih dreves, ima pa v primerjavi z njo pomembne prednosti. Z mozaičnimi diagrami namreč hkrati vizualiziramo več atributov, kar je primerljivo z gradnjo dreves s pogledom vnaprej (ang. *look-ahead*). Ko želimo v neki veji nadaljevati z gradnjo drevesa, nam tako ni potrebno uporabljati kratkovidnih postopkov za izbiranje atributa, ampak lahko za primere v tej veji z opisanim algoritmom avtomatsko poiščemo zanimive mozaične diagrame, ki vsebujejo take kombinacije atributov, s katerimi nam bo uspelo najbolje ločiti med različnimi razredi.



(a) (b) (c) (d) (e) (f)

Slika 5.4: Primeri najbolje ocenjenih mozaičnih diagramov na domenah monks 1 (a), car (b), breast-cancer-winsconsin (c), wdbc (d) ter krkp (e,f).

Poglavje 6

Zaključek

Vizualizacija podatkov je eno ključnih orodij pri analizi podatkov. Omogoča nam vizualno detekcijo kompleksnih struktur in vzorcev v podatkih, ki bi jih sicer težko odkrili na kakšen drug način. Zakonitosti, ki jih iščemo, so seveda odvisne od tipa podatkov. V primeru nenadzorovanega učenja so zanimivi tisti prikazi, ki nam razkrijejo strukture kot so naprimer gruče primerov, osamelci, trendi ter relacije med spremenljivkami. Podobno lahko pri vizualizaciji klasificiranih podatkov smatramo kot informativne tiste prikaze, v katerih so različni razredi čim bolje ločeni. Taki prikazi nam omogočajo vizualno odkrivanje pravil za ločevanje med razredi. Poleg tega, da lahko z njimi odkrijemo pomembne atribute, nam ti prikazi omogočajo tudi razumevanje podobnosti in razlik med različnimi razredi.

Ker različni prikazi podatkov ne omogočajo enakega vpogleda v vsebovane zakonitosti, je naloga uporabnika običajno ta, da ročno poišče najinformativnejše prikaze. Težava je v tem, da zbirke podatkov dandanes vsebujejo veliko število atributov. Iskanje zanimivih prikazov podatkov je zato za uporabnika težko in časovno zahtevno opravilo, saj število možnih prikazov narašča eksponentno s številom hkrati vizualiziranih atributov. Že pri uporabi enostavnega razsevnega diagrama, ki hkrati vizualizira le dva atributa, lahko za zbirko podatkov z m atributi generiramo $m(m-1)/2$ različnih diagramov. Veliko število možnih prikazov je tako eden glavnih razlogov, ki bistveno zmanjšuje uporabnost vizualizacije pri analizi podatkov ter zaradi katerega se (kljub številnim večdimenzionalnim metodam) najpogosteje uporabljajo zgolj enostavne eno- in dvodimenzionalne vizualizacijske metode.

Da bi odpravili to ključno pomanjkljivost in izboljšali uporabnost vizualizacije, smo v disertaciji razvili metodo VizRank, ki omogoča avtomatsko ocenjevanje zanimivosti prikazov klasificiranih podatkov. Zanimivost je ocenjena glede na to, kako uspešno lahko iz prikaza ločimo med različnimi razredi. VizRank je mogoče uporabiti z vsako vizualizacijsko metodo, ki vrednosti atributov preslika v položaj simbola v prikazu. Da bi izračunal oceno zanimivosti za posamezen prikaz, VizRank najprej generira novo zbirko podatkov,

ki vsebuje zgolj x in y položaje točk v prikazu ter informacijo o razredu. Na tej zbirki nato uporabi algoritem k-najbližjih sosedov in z njim oceni napovedno točnost. Dobljena napovedna točnost predstavlja oceno zanimivosti prikaza. Prikazi, v katerih je ločenost med razredi dobra, bodo dosegli visoko oceno, prikazi, v katerih se razredi medsebojno prekrivajo, pa temu primerno slabšo oceno. Uporabniku tako zanimivih prikazov ni potrebno iskati ročno, ampak si lahko ogleda zgolj majhno podmnožico najbolje ocenjenih prikazov, ki mu bodo omogočali najboljše ločevanje med razredi.

Ker ob uporabi večdimenzionalnih vizualizacijskih metod število možnih projekcij podatkov narašča eksponentno s številom vizualiziranih atributov, se moramo tudi ob uporabi postopka VizRank pogosto zadovoljiti z ocenjevanjem zgolj podmnožice možnih projekcij. Da pri tem ne bi izpustili pomembnih zanimivih projekcij, smo v disertaciji predstavili hevristiko, s katero vsaki projekciji določimo grobo oceno zanimivosti zgolj na osnovi zanimivosti atributov, ki se v njej pojavljajo. VizRank nato to oceno upošteva tako, da projekcije ocenjuje v vrstnem redu od tistih z najboljšo do tistih z najslabšo grobo oceno zanimivosti. Opisana hevristika se je pri iskanju zanimivih projekcij izkazala kot nepogrešljiva, saj VizRanku omogoča, da zelo hitro identificira zanimive projekcije ter se pri tem izogne ocenjevanju neinformativnih delov prostora možnih projekcij.

Za preverjanje hipoteze, da se ocene zanimivosti, ki jih projekcijam določi VizRank, ujemajo z ocenami, ki bi jih določil človek, smo izvedli eksperiment, v katerem je sodelovalo 30 ljudi. Vsak od njih je moral pri 20 naključno izbranih parih projekcij določiti, ali je ena od projekcij bolj zanimiva kot druga in če to je, koliko bolj zanimiva je. Rezultati so pokazali zelo visoko ujemanje med VizRankovimi ocenami in ocenami udeležencev eksperimenta. Dodatno pomembno odkritje je bilo, da je ujemanje največje, če za učni algoritem izberemo k najbližjih sosedov, za ocenjevanje točnosti algoritma pa povprečno verjetnost pravilne klasifikacije.

V disertaciji smo predstavili tudi različne postopke, s katerimi lahko iz seznama najboljših projekcij pridobimo dodatno znanje. Ena od možnosti je uporaba seznama projekcij za ocenjevanje pomembnosti posameznih atributov – večkrat kot se nek atribut pojavi med najboljšimi projekcijami, pomembnejši je za ločevanje razredov. Tovrstno ocenjevanje atributov je uspešnejše od kratkovidnih mer, saj upošteva potencialne odvisnosti med atributi. Opisali smo tudi eksperiment, v katerem se je ta mera izkazala celo kot uspešnejša od nekratkovidne mere ReliefF. Slabost postopka je vsekakor njegova počasnost (potrebno je oceniti večje število projekcij), zaradi česar je manj primeren v primerih, ko je ocenjevanje atributov potrebno ponavljati velikokrat (npr. pri gradnji odločitvenih dreves). S seznamom najboljših projekcij je mogoče uspešno odkrivati tudi osamelce oziroma primere, ki so netipični predstavniki svojega razreda. Netipičnost posameznega primera ocenimo tako, da preverimo, kakšna je okolica tega primera v najboljših projekcijah. Za tipične predstavnike razreda običajno velja, da ležijo med primeri svojega razreda, netipični predstavniki pa pogosto ležijo na robu svoje gruče točk ali pa celo med primeri drugega razreda. Uporabnost in pravilnost tega postopka smo demonstrirali na podatkih o pljučnem raku.

Zanimive projekcije lahko med drugim uporabimo tudi pri klasifikaciji novih primerov. Večina učnih algoritmov običajno zgradi netransparentne napovedne modele, ki ljudem ne omogočajo vpogleda vanje in posledično razumevanja izračunanih napovedi. Ker so v nasprotju s tem projekcije relativno enostavno razumljive, nas je zanimalo, kako uspešno jih lahko uporabimo pri klasifikaciji. Naši poskusi na različnih zbirkah podatkov so pokazali, da lahko z uporabo zgolj ene projekcije dosežemo zelo visoko napovedno točnost, ki je primerljiva s točnostmi najboljših učnih algoritmov. Visoka napovedna točnost projekcij je zanimiva tudi zato, ker potrjuje, da dobra ločenost različnih razredov v projekcijah ni rezultat pretiranega prilagajanja podatkom, temveč dejanska zakonitost, ki jo podatki vsebujejo.

Glede na širok izbor uporabnih vizualizacijskih metod smo princip avtomatskega ocenjevanja zanimivosti možnih prikazov razširili še na dve netočkovni vizualizacijski metodi. Prva je bila metoda paralelnih koordinat, pri kateri so atributi predstavljeni s paralelnimi osmi, primeri pa so vizualizirani kot črte, ki potekajo med prvo in zadnjo koordinatno osjo. Iz prikaza je mogoče razbrati porazdelitev vrednosti primerov vzdolž posameznih osi, z opazovanjem naklona črt med sosednjimi osmi pa lahko sklepamo tudi o tem, v kakšni relaciji sta sosednja atributa.

Kot pri ostalih vizualizacijskih metodah je tudi informativnost prikaza s paralelnimi koordinatami zelo odvisna od izbora in vrstnega reda prikazanih atributov. Ker je iz prikaza mogoče opaziti zgolj relacijo med sosednjimi atributi, smo v disertaciji predstavili algoritem, ki poskuša atribute urediti tako, da so čim bolje opazne pomembne relacije med sosednjimi atributi. Kot primera dveh pomembnih relacij smo predstavili korelacijo med sosednjima atributoma ter ločenost razredov. S korelacijo dosežemo regularen potek črt med osmi in s tem bistveno izboljšamo percepcijo primerov, z ločenostjo razredov pa uporabniku omogočimo lažjo vizualno detekcijo pravil za ločevanje med razredi. Algoritem je mogoče na istih podatkih uporabiti večkrat, pri čemer nam vsakič vizualizira zgolj tiste pomembne relacije med atributi, ki še niso bile vizualizirane v prejšnjih prikazih.

Postopek za rangiranje prikazov smo nazadnje razvili tudi za mozaične diagrame, ki so namenjeni vizualizaciji diskretnih atributov. Tudi v tem primeru sta izbor in postavitev vizualiziranih atributov ključna faktorja, ki določata zanimivost prikazov. Izbor atributov določa, kakšna bo porazdelitev primerov po celicah – bolj kot so posamezne celice "čiste" (vsebujejo primere zgolj enega razreda), bolj je izbor atributov zanimiv. Postavitev izbranih atributov nato določa relativni medsebojni položaj posameznih celic. Od različnih postavitev atributov nam je najzanimivejša tista, pri kateri so celice s primeri istega razreda čim bolj skupaj in oddaljene od celic, ki vsebujejo primere drugih razredov. Taka postavitev atributov omogoča enostavno vizualno detekcijo pravil, ki veljajo za posamezne skupine celic. V disertaciji smo predstavili različne kriterije za ocenjevanje zanimivosti posameznih izborov atributov ter algoritem, s katerim lahko poiščemo najprimernejšo postavitev atributov.

Pri vseh razvitih algoritmih smo njihovo uporabnost demonstrirali z različnimi primeri uporabe. V ta namen smo uporabili različne zbirke podatkov iz repozitorija UCI ter

podatke o raku, pridobljene z uporabo mikromrež.

6.1 Nadaljnje delo

Možnosti za nadaljnje delo je vsekakor veliko.

Ena od pomanjkljivosti metode VizRank je, da ocenjuje zanimivost zgolj celotne projekcije. Tako ocenjevanje je manj primerno kadar analiziramo podatke z večjim številom razredov, saj je malo verjetno, da bomo mogli z eno samo projekcijo ločiti med vsemi razredi. V takem primeru bi bilo bolj smiselno poiskati take prikaze, v katerih je le podmnožica razredov dobro ločena od ostalih, medtem ko se primeri ostalih razredov lahko prekrivajo.

Podobna težava je tudi pri mozačnih diagramih. Pogosto namreč odkrijemo diagrame, v katerih so nekatere celice čiste (in zaradi tega zanimive), preostale celice pa vsebujejo zelo neinformativno kombinacijo primerov iz različnih razredov. Ker trenutni algoritem oceni zanimivost celotnega prikaza, so taki diagrami pogosto slabše ocenjeni kot diagrami, ki ne vsebujejo nobene zelo zanimive celice, imajo pa "ustreznejšo" porazdelitev primerov, zaradi katere je povprečna zanimivost celic višja. Za podatke, pri katerih ne uspemo najti zanimivih diagramov, bi bilo zato koristno imeti na voljo kriterij, ki bi prikaze znal oceniti zgolj na osnovi podmnožice njihovih najzanimivejših celic.

V razdelku 3.5.2 smo pokazali, da je projekcije mogoče uspešno uporabiti za klasifikacijo. Uporabo projekcij v ta namen bi bilo mogoče še dodatno izboljšali, saj so projekcije bistveno bolje razumljive kot običajni napovedni modeli. Razvili bi lahko neke vrste "vizualno argumentacijo", pri kateri bi za primer, ki bi ga želeli klasificirati, poiskali različne zanimive projekcije, na osnovi katerih bi primer klasificirali v različne razrede. Vsako od projekcij bi smatrali kot argument za klasifikacijo v določen razred. Jakost argumenta bi določili glede na zanimivost celotne projekcije ter glede na položaj primera v projekciji (ali leži primer v središču ali na robu gruče točk), razred primera pa bi nato določili z glasovanjem teh argumentov.

Poleg opisanih vizualizacijskih metod bi bilo smiselno preučiti tudi prednosti ostalih metod za odkrivanje znanja v podatkih. Za uspešne metode bi nato ugotovili, katere so lastnosti, ki veljajo za zanimiv prikaz in nato definirali algoritem, s katerim bi bilo mogoče prikaze ocenjevati avtomatsko.

Dodatek A

Uporabljene zbirke podatkov iz mikromrež

V disertaciji smo uporabljali šest zbirk podatkov o različnih vrstah rakastih obolenj. Zbirke so bile pridobljene z uporabo DNA mikromrež, ki omogočajo hkratno merjenje izraženosti več tisoč genov v človeškem organizmu. Osnovne statistične informacije o zbirkah so opisane v tabeli A.1, v nadaljevanju pa bom podrobneje opisal vsako od njih.

Levkemija. To je ena najbolj znanih in pogosto analiziranih zbirk podatkov o izraženosti genov in vsebuje primere akutno limfoblastne (*ALL*, 47 primerov) ter akutno mieloične (*AML*, 25 primerov) levkemije. Originalna raziskava, ki jo je na teh podatkih opravil Golub s sod. [41], je bila ena prvih, ki je potrdila uspešnost klasifikacije raka na osnovi opazovanja izraženosti genov.

SRBCT. Zbirka vsebuje podatke o štirih vrstah majhnih okroglih modrih celičnih tumorjev (ang. *small round blue cell tumors*), ki se pojavljajo pri otrocih. Ti tumorji so Ewingov sarkom (*EWG*, 29 primerov), nevroblastom (*NB*, 18 primerov), Burkitov limfom (*BL*, 11 primerov) ter rabdomiosarkom (*RMS*, 25 primerov). Ker so vse štiri vrste tumorjev histološko gledano zelo podobne je njihova klinična diagnoza zelo zahtevna, pravilna diagnoza pa je bistvenega pomena, saj so od nje odvisni načini zdravljenja, pričakovani odzivi na terapijo in prognoza. Khan s sod. [62] je pri analizi teh podatkov uporabil zapleteno kombinacijo številnih algoritmov strojnega učenja.

MLL. To so ponovno podatki o levkemiji, pri čemer so nekateri primeri akutno limfoblastne levkemije podrobneje razdeljeni v podrazred bilinearne levkemije s kromosomsko translokacijo (*MLL*, 20 primerov). Primere tovrstne levkemije tipično najdemo pri otrocih ter pri levkemijah, ki nastanejo kot posledica kemoterapije, in imajo posebno slabo prognozo. Zmožnost ločevanja teh primerov od ostalih dveh vrst je v svoji raziskavi

Zbirka	Primerov (pacientov)	Atributov (genov)	Razredov	Večinski razred
levkemija	73	7.074	2	52,8%
SRBCT	83	2.308	4	34,9%
MLL	72	12.533	3	38,9%
DLBCL	77	7.070	2	75,3%
prostata	102	12.533	2	51,0%
pljučni rak	203	12.600	5	68,5%

Tabela A.1: Uporabljene zbirke podatkov o različnih vrstah raka.

prvi potrdil Armstrong s sod. [1].

DLBCL. Zbirka podatkov o dveh tipih B-celičnih malignosti: razpršenih velikih B-celičnih limfomih (*DLBCL*, 58 primerov) ter mešičkasih limfomih (*FL*, 19 primerov). Za obe vrsti je značilno, da se s časom spreminjata in pridobivata morfološke in klinične značilnosti druge vrste. Pomembno analizo teh podatkov je napravila Shipp s sod. [96].

Prostata. Podatki o raku na prostati, ki jih je zbral Singh s sod. [98]. Za razliko od ostalih zbirk v tem primeru cilj ni ločevati med različnimi vrstami raka, ampak med tumornim (52 primerov) in zdravim tkivom (50 primerov).

Pljučni rak. Meritve vsebujejo primerke zdravega tkiva (*NL*, 17 primerov) ter primere štirih vrst pljučnega raka: adenokarcinoma (*AD*, 139 primerov), majhno-celičnega karcinoma (*SMCL*, 6 primerov), luskavo-celičnega karcinoma (*SQ*, 21 primerov) ter pljučnega karcinoida (*COID*, 20 primerov). Podrobna analiza podatkov je objavljena v [5].

Dodatek B

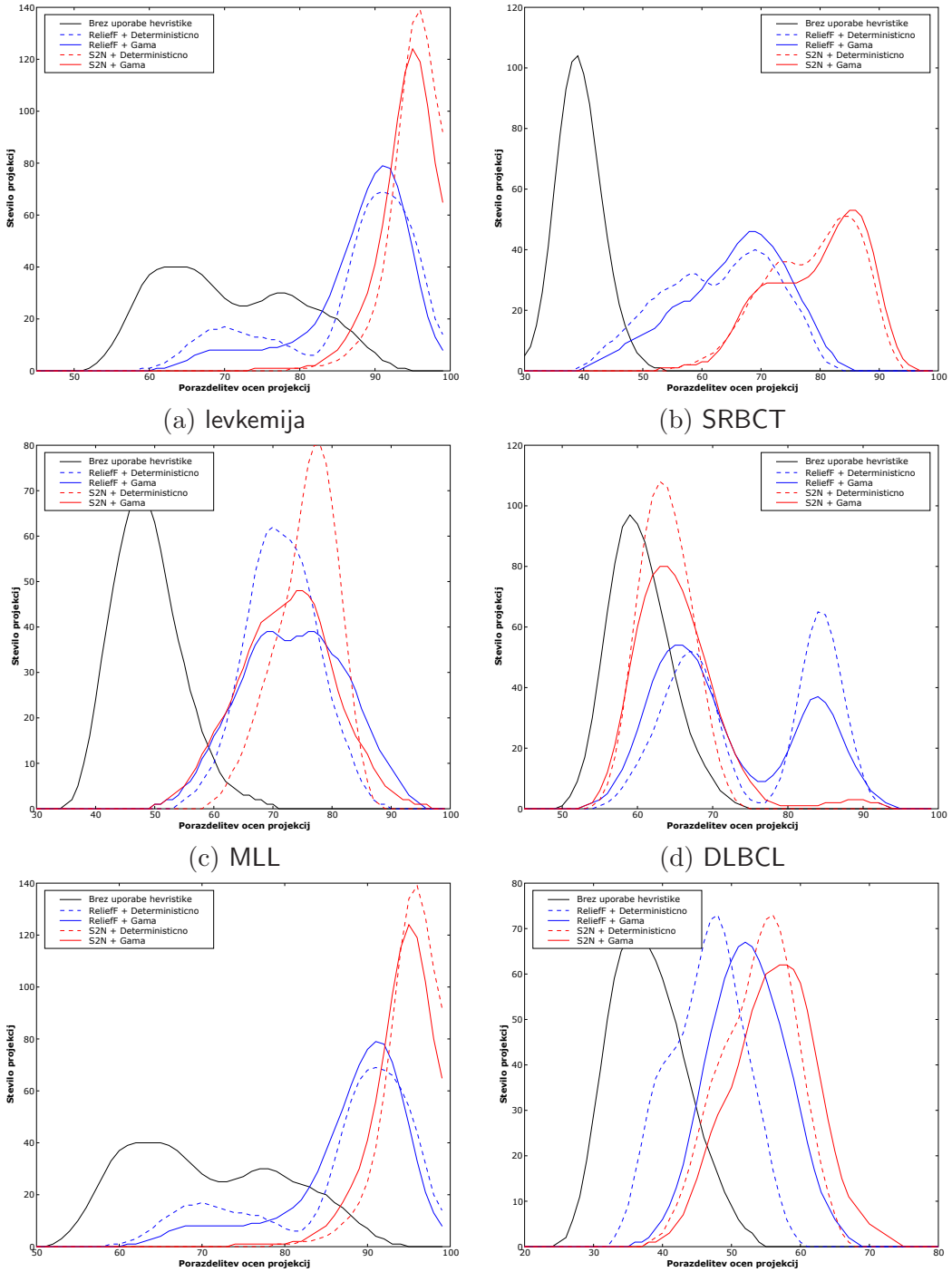
Ocena uspešnosti hevristik na različnih zbirkah podatkov

V razdelku 3.2 smo predstavili različne hevristike, s katerimi je mogoče hitreje odkriti zanimive projekcije podatkov. Njihovo uspešnost smo demonstrirali zgolj na eni zbirki podatkov in sicer na podatkih SRBCT. V tem poglavju bomo prikazali še rezultate na ostalih zbirkah podatkov iz dodatka A in s tem dodatno potrdili njihovo učinkovitost.

Sliki B.1 in B.2 prikazujeta uspešnost splošne hevristike. Iz vseh grafov je razvidno, da je hevristika nujno potrebna, če želimo identificirati projekcije, ki bodo dobile visoko oceno zanimivosti. Brez njene uporabe so slabše tako najvišje ocene projekcij kot tudi sama distribucija ocen ocenjenih projekcij. Če primerjamo med sabo uspešnosti mer S2N in ReliefF, lahko opazimo, da se je na izbranih zbirkah v večini primerov mera S2N obnesla bolje. Izjema je zbirka podatkov DLBCL, kjer smo z mero ReliefF pregledovali zanimivejše dele prostora možnih projekcij. Zanimiva je tudi primerjava med determinističnim izbiranjem atributov ter izbiranjem z uporabo porazdelitve gama. Če opazujemo zgolj porazdelitve ocen projekcij, bi lahko sklepali, da noben od načinov ne izstopa po uspešnosti – v nekaterih primerih je boljši prvi v drugih primerih pa drugi način izbora atributov. Sklep pa bo drugačen, če si poleg tega ogledamo tudi grafe ocen najboljših projekcij. Na teh grafih lahko namreč vidimo, da izbiranje atributov z uporabo porazdelitve gama skoraj v vseh primerih najde bistveno boljše projekcije. Ta lastnost je relativno pričakovana, saj je, kot že rečeno, pri determinističnem izbiranju atributov v izborih zelo majhna raznolikost atributov.

Na slikah B.3 in B.4 sta primerjani splošna hevristika ter posebna hevristika za metodo radviz. Iz grafov je razvidno, da je posebna hevristika bistveno uspešnejša pri identifikaciji zanimivih prikazov. Razlike so še posebej izrazite pri podatkih MLL, DLBCL ter podatkih o pljučnem raku. Zanimivo je tudi to, da ima pri posebni hevristiki način izbiranja atributov (deterministično ali verjetnostno) bistveno manjši vpliv na dobljene ocene najboljših projekcij kot pri splošni hevristiki.

Glede na to, da smo pri vseh omenjenih poskusih ocenili "zgolj" 60.000 projekcij, nas je zanimalo, kako se spremenita porazdelitev ocen ter ocena najboljše najdene projekcije, če ocenimo bistveno večje število projekcij. V ta namen smo za podatke SRBCT ocenili 6.000.000 različnih projekcij z uporabo splošne ter posebne hevristike. Dobljeni grafi so prikazani na sliki B.5. Grafa B.5.a in B.5.b prikazujeta uspešnost splošne hevristike pri različnih parametrih, grafa B.5.c in B.5.d pa primerjavo med splošno in posebno hevristiko. Če te grafe primerjamo z grafi za podatke SRBCT na slikah B.1–B.4, lahko med njimi opazimo veliko podobnost. Oblika krivulj pri porazdelitvi ocen ocenjenih projekcij ostane skoraj nespremenjena, ocena najboljše najdene projekcije pa se z večanjem števila ocenjenih projekcij še naprej dviguje.

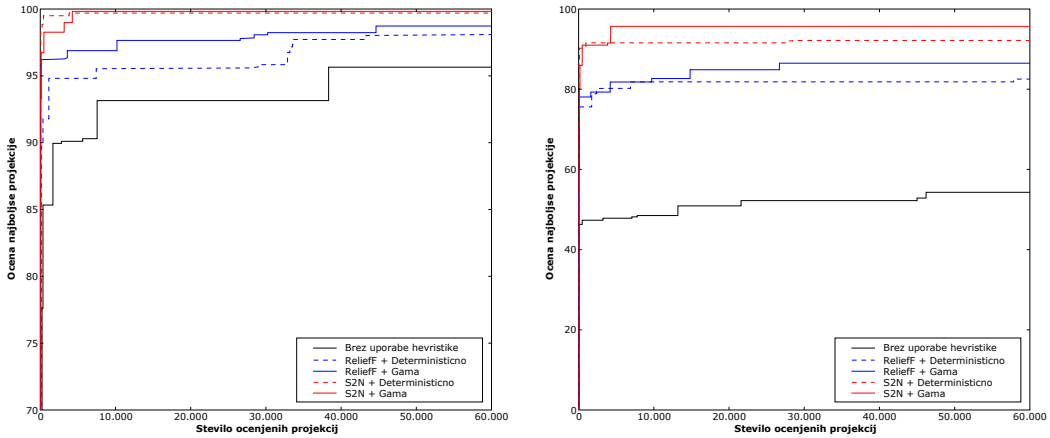


(a) levkemija (b) SRBCT

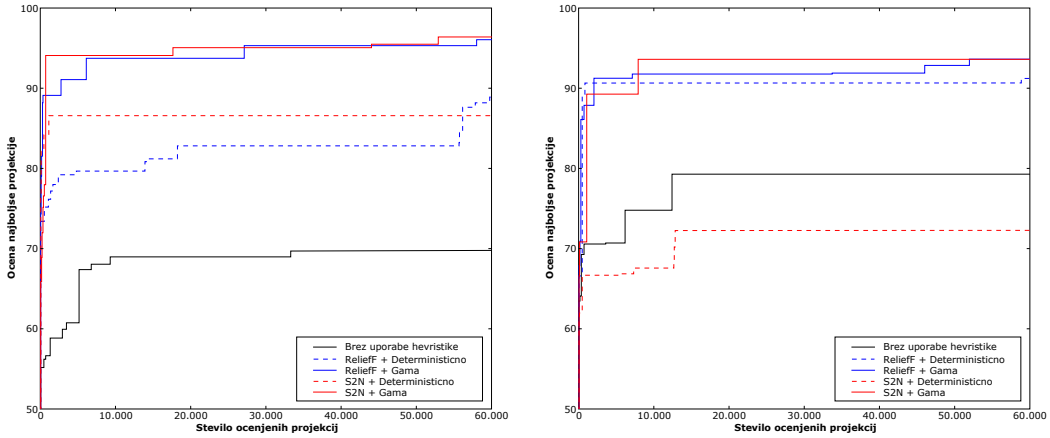
(c) MLL (d) DLBCL

(e) prostata (f) pljučni rak

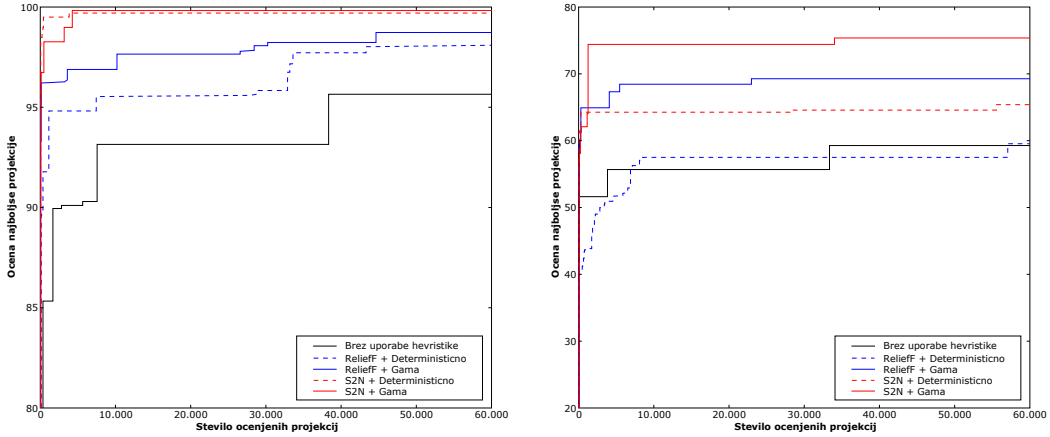
Slika B.1: Uspešnost splošne hevristike pri ocenjevanju 60.000 projekcij radviz. Grafi prikazujejo porazdelitve ocen projekcij pri različnih parametrih hevristike.



(a) levkemija (b) SRBCT

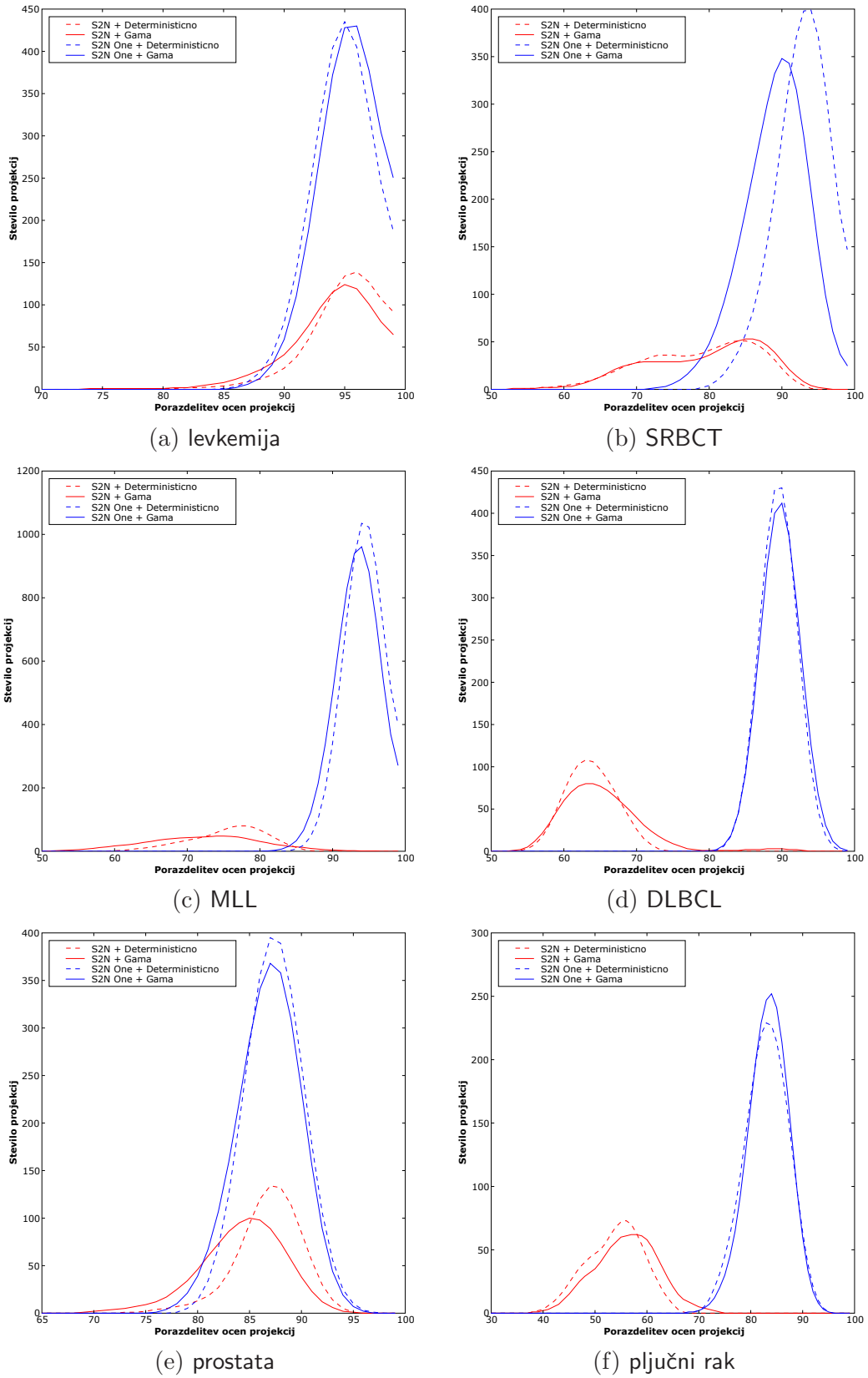


(c) MLL (d) DLBCL



(e) prostata (f) pljučni rak

Slika B.2: Uspešnost splošne hevristike pri ocenjevanju 60.000 projekcij radviz. Grafi prikazujejo, kako z naraščanjem števila ocenjenih projekcij narašča ocena najboljše najdene projekcije.

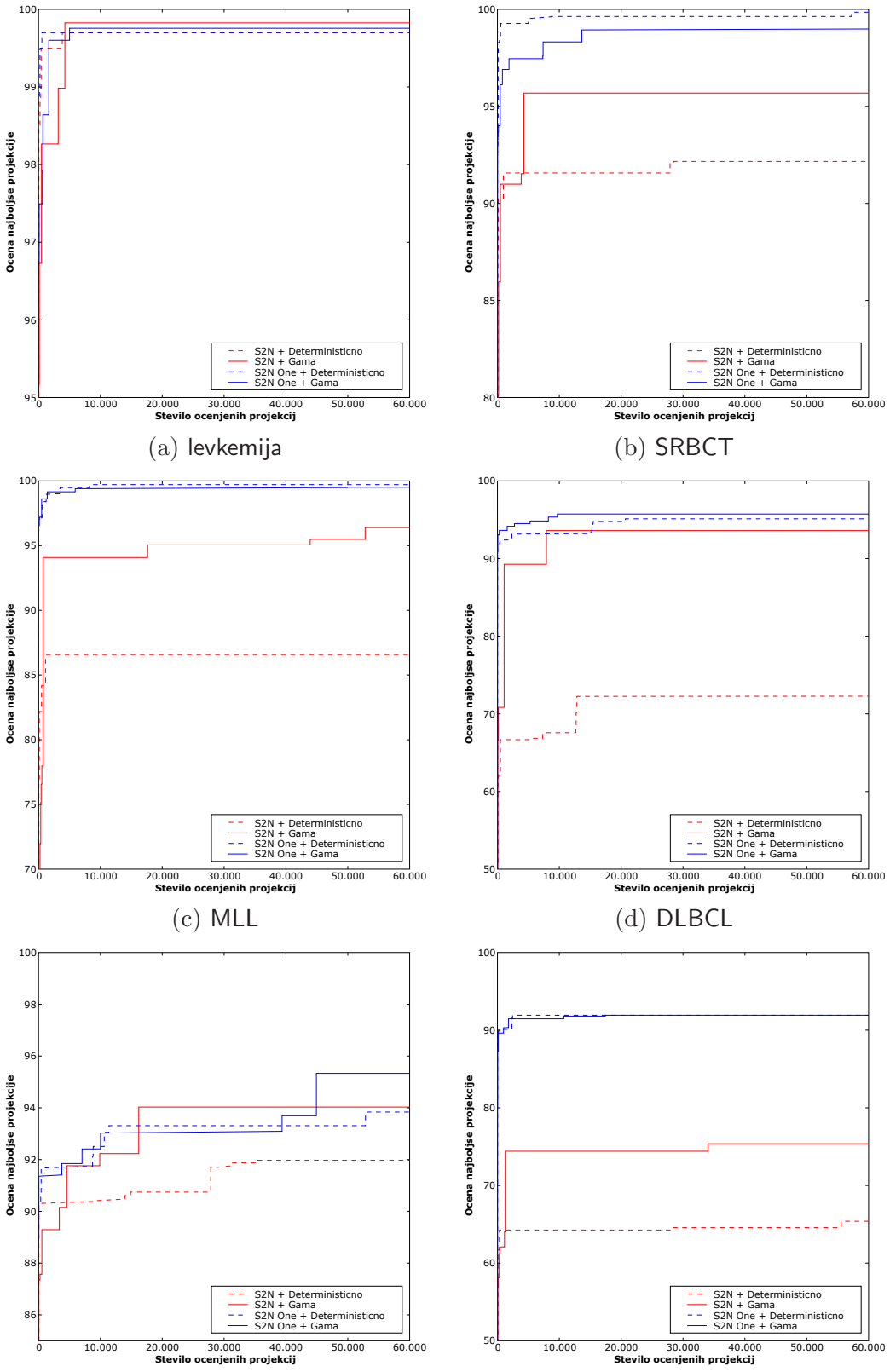


(a) levkemija (b) SRBCT

(c) MLL (d) DLBCL

(e) prostata (f) pljučni rak

Slika B.3: Primerjava med splošno in posebno hevristiko za metodo radviz pri ocenjevanju 60.000 projekcij. Grafi prikazujejo porazdelitve ocen projekcij pri različnih parametrih hevristike.

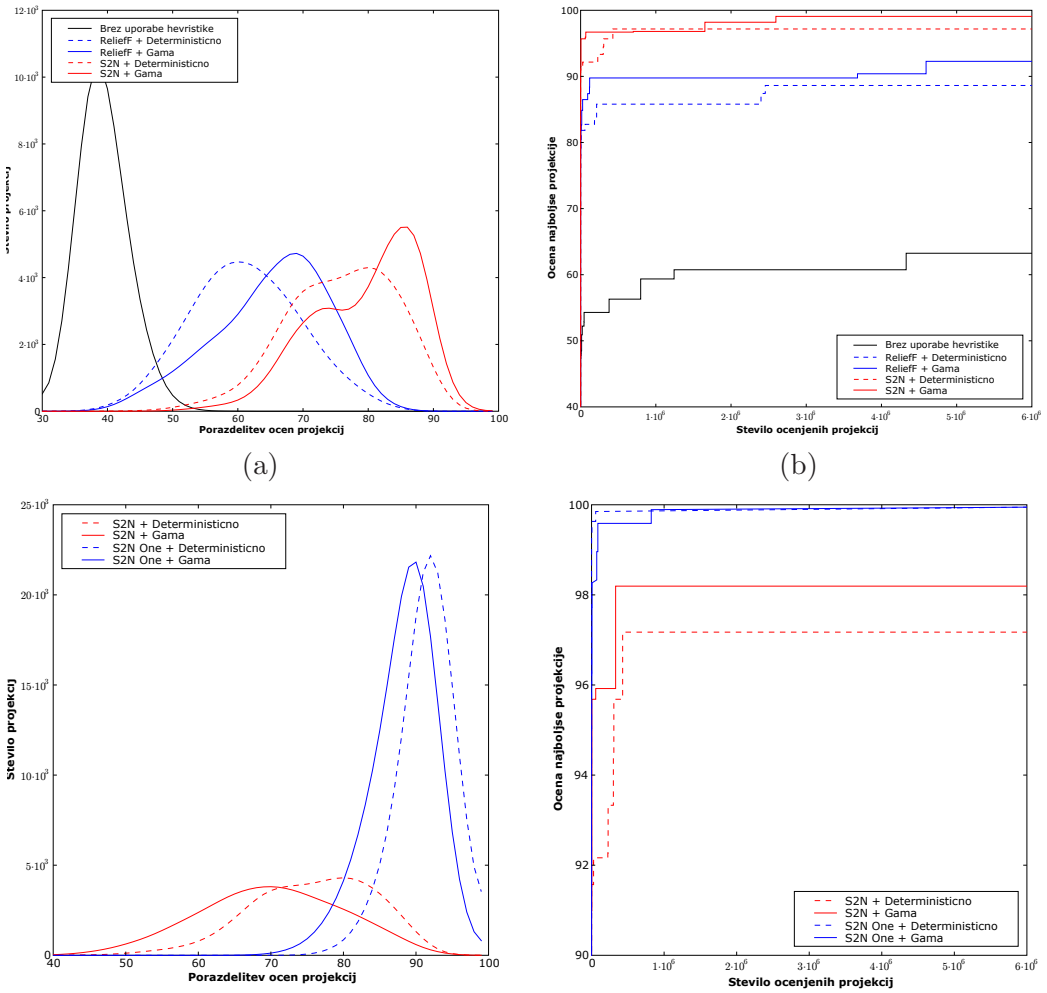


(a) levkemija (b) SRBCT

(c) MLL (d) DLBCL

(e) prostata (f) pljučni rak

Slika B.4: Primerjava med splošno in posebno hevristiko za metodo radviz pri ocenjevanju 60.000 projekcij. Grafi prikazujejo, kako z naraščanjem števila ocenjenih projekcij narašča ocena najboljše najdene projekcije.



Slika B.5: Uspešnost hevristik pri ocenjevanju $6 \cdot 10^6$ projekcij. Grafa (a) in (b) prikazujeta ocene projekcij dobljene s splošno hevristiko pri različnih parametrih, grafa (c) in (d) pa primerjavo ocen splošne in posebne hevristike.

Izjava

Izjavljam, da sem doktorsko disertacijo izdelal samostojno pod vodstvom mentorjev akad. prof. dr. Ivana Bratka ter izr. prof. dr. Blaža Zupana. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.

Literatura

[1] S. A. Armstrong, J. E. Staunton, L. B. Silverman, *et al.* MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nature Genetics*, 30(1):41–47, 2001.

[2] D. Asimov. The grand tour: A tool for viewing multidimensional data. *SIAM Journal on Scientific and Statistical Computing*, 61:128–143, 1985.

[3] R. E. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[4] J. L. Bentley. Binary search trees used for associative searching. *Communications of the ACM*, 18:509–517, 1975.

[5] A. Bhattacharjee, W. G. Richards, J. Staunton, *et al.* Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences*, 98(24):13790–13795, 2001.

[6] M. Bohanec and V. Rajkovič. Expert system for decision making. *Sistemica*, 1(1):145–157, 1990.

[7] G. W. Brier. Verification of forecasts expressed in terms of probabilities. *Monthly Weather Review*, 78:1–3, 1950.

[8] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Sciences*, 103(5):1168–1172, 2006.

[9] M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. S. Furey, M. Jr. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97(1):262–267, 2000.

[10] C. Brunsdon, A. S. Fotheringham, and M. E. Charlton. An investigation of methods for visualising highly multivariate datasets. *Case Studies of Visualization in the Social Sciences*, strani 55–80, 1998.

[11] J. G. Bryan. The generalized discriminant function: Mathematical foundation and computational routine. *Harvard Educational Review*, 21:90–95, 1951.

[12] T. C. Callaghan. Interference and domination in texture segregation. *Proceedings of the First International Conference on Visual Search*, strani 81–87, 1988.

[13] T. C. Callaghan. Interference and dominance in texture segregation: hue, geometric form, and line orientation. *Percept & Psychophys*, 46(4):299–311, 1989.

[14] S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1999.

[15] J. M. Chambers, W. S. Cleveland, B. Kleiner, and P. A. Tukey. *Graphical Methods for Data Analysis*. Wadsworth Press, 1983.

[16] H. Chernoff. The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association*, 68:361–368, 1973.

[17] D. Cook, A. Buja, and J. Cabrera. Projection pursuit indexes based on orthonormal function expansions. *Journal of Computational and Graphical Statistics*, 2(3):225–250, 1993.

[18] D. Cook, A. Buja, J. Cabrera, and C. Hurley. Grand tour and projection pursuit. *Journal of Computational and Graphical Statistics*, 4(3):155–172, 1995.

[19] M. F. Cox and M. A. A. Cox. *Multidimensional Scaling*. Chapman and Hall, 2001.

[20] S. L. Crawford. Genetic optimization for exploratory projection pursuit. *Proceedings of the 23rd Symposium of the Interface between Computing Science and Statistics*, strani 318–321, 1991.

[21] G. A. Croes. A method for solving traveling-salesman problems. *Operations Research*, 5:791–812, 1958.

[22] J. E. Cutting and P. M. Vishton. Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. In *Handbook of perception and cognition*, strani 69–117. Academic Press, San Diego, CA, 1995.

[23] B. W. Dasarathy. *Nearest neighbor (NN) norms: NN pattern classification techniques*. IEEE Computer Society Press, 1991.

[24] J. Demšar, G. Leban, and B. Zupan. Freeviz - an intelligent visualization approach for class-labeled multidimensional data sets. *Proceedings of IDAMAP 2005, Edinburgh*, 2005.

[25] J. Demšar, B. Zupan, and G. Leban. From experimental machine learning to interactive data mining, a white paper. *Fakulteta za računalništvo in informatiko, Ljubljana*, 2004.

[26] J. L. DeRisi, V. R. Iyer, and P. O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680–686, 1997.

[27] S. C. Eick, J. L. Steffen, and E. E. Sumner. SeeSoft - a tool for visualizing line oriented software statistics. *IEEE Transactions on Software Engineering*, 18(11):957–968, 1992.

[28] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[29] S. Feiner and C. Beshers. Worlds within worlds: metaphors for exploring n-dimensional virtual worlds. *Readings in information visualization: using vision to think*, strani 96–106, 1999.

[30] S. K. Feiner and C. Beshers. Visualizing n-dimensional virtual worlds with n-vision. *Proceedings of the 1990 symposium on Interactive 3D graphics*, strani 37–38, 1990.

[31] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[32] M. M. Flood. The traveling-salesman problem. *Operations Research*, 4:61–75, 1956.

[33] J. H. Friedman. Exploratory projection pursuit. *Journal of American Statistical Association*, 82:249–266, 1987.

[34] J. H. Friedman, J. L. Bentley, and R. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3):209–226, 1977.

[35] J. H. Friedman and W. Stuetzle. Projection pursuit regression. *Journal of the American Statistical Association*, 76(376):817–823, 1981.

[36] J. H. Friedman and W. Stuetzle. Projection pursuit methods for data analysis. In R. L. Launer and A. F. Siegel, editors, *Modern Data Analysis*, strani 179–200. Academic Press, New York, 1982.

[37] J. H. Friedman, W. Stuetzle, and A. Schroeder. Projection pursuit density estimation. *Journal of the American Statistical Association*, 79(387):599–608, 1984.

[38] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers*, 23:881–890, 1974.

[39] M. Friendly. Mosaic displays for loglinear models. *Proceedings of the Statistical Graphics Section*, strani 61–68, 1992.

[40] M. Friendly. Mosaic displays for multi-way contingency tables. *Journal of the American Statistical Association*, 89:190–200, 1994.

[41] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[42] R. C. Gonzales and P. Wintz. *Digital image processing (2nd ed.)*. Addison-Wesley Longman Publishing Co., Boston, MA, USA, 1987.

[43] M. Graham and J. Kennedy. Using curves to enhance parallel coordinate visualisations. In *Seventh International Conference on Information Visualization (IV'03)*, strani 10–17, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[44] G. Grinstein, R. Pickett, and M. G. Williams. EXVIS: An exploratory visualization environment. *Proceedings on Graphics Interface'89*, 1989.

[45] G. Grinstein, M. Trutschl, and U. Cvek. High-dimensional visualizations. *Proceedings of the Visual Data Mining Workshop, KDD*, 2001.

[46] P. Hall. On polynomial-based projection indices for exploratory projection pursuit. *The Annals of Statistics*, 17(2):589–605, 1989.

[47] J. A. Hartigan and B. Kleiner. Mosaics for contingency tables. *Computer Science and Statistics: Proceedings of the 13th Symposium on the Interface*, 22:286–273, 1981.

[48] J. A. Hartigan and B. Kleiner. A mosaic of television ratings. *The American Statistician*, 38:32—-35, 1984.

[49] H. Hauser, F. Ledermann, and H. Doleisch. Angular brushing for extended parallel coordinates. In *Proceedings of the IEEE Symposium on Information Visualization*, strani 127–130, 2002.

[50] P. Hoffman, G. Grinstein, and D. Pinkney. Dimensional anchors: A graphic primitive for multidimensional multivariate information visualizations. *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation*, strani 9–16, 1999.

[51] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417—-441, 1933.

[52] P. J. Huber. Projection Pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.

[53] E. B. Hunt, J. Marin, and P. J. Stone. *Experiments in induction*. Academic Press New York, 1966.

[54] A. Inselberg. *n-dimensional graphics, part i–lines and hyperplanes*. Technical Report G320-2711. IBM Los Angeles Scientific Center, 1981.

[55] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *VIS '90: Proceedings of the 1st conference on Visualization '90*, strani 361–378, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.

[56] A. Jakulin and I. Bratko. Analyzing attribute dependencies. *PKDD*, strani 229–240, 2003.

[57] A. Jakulin and I. Bratko. Analyzing attribute interactions. *Lecture Notes in Artifficial Intelligence*, 2838:229–232, 2003.

[58] A. Jakulin and I. Bratko. *Machine Learning Based on Attribute Interactions*. Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2005.

[59] M. C. Jones and R. Sibson. What is projection pursuit? *Journal of the Royal Statistical Society. Series A (General)*, 150(1):1–37, 1987.

[60] D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[61] D. A. Keim and H. P. Kriegel. VisDB: Database exploration using multidimensional visualization. *IEEE Computer Graphics and Applications*, 14:40–49, 1994.

[62] J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, *et al.* Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(1):673–679, 2001.

[63] K. Kira and L. A. Rendell. The feature selection problem: Traditional methods and a new algorithm. *Proceedings of the Tenth National Conference on Artificial Intelligence*, strani 129–134, 1992.

[64] I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. *Proceedings of the European Conference on Machine Learning*, strani 171–182, 1994.

[65] I. Kononenko. On biases in estimating multi-valued attributes. *Proceedings of International Joint Conference on Artificial Intelligence IJCAI-95*, strani 1034–1040, 1995.

[66] I. Kononenko and E. Simec. Induction of decision trees using ReliefF. In *Mathematical and statistical methods in artificial intelligence*. Springer Verlag, 1995.

[67] I. Kononenko, E. Šimec, and M. Robnik-Šikonja. Overcoming the myopia of inductive learning algorithms with RELIEFF. *Applied Intelligence*, 7(1):39–55, 1997.

[68] Y. Koren and L. Carmel. Visualization of labeled data using linear transformations. *Proceedings of IEEE Information Visualization*, strani 121–128, 2003.

[69] Y. Koren and L. Carmel. Robust linear dimensionality reduction. *IEEE Transactions on Visualization and Computer Graphics*, 10(4):459–470, 2004.

[70] J. B. Kruskal. Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new 'index of condensation'. *Statistical Computation*, strani 427–440, 1969.

[71] G. Leban, I. Bratko, U. Petrovič, T. Curk, and B. Zupan. VizRank: finding informative data projections in functional genomics by machine learning. *Bioinformatics*, 21(3):413–414, 2005.

[72] G. Leban, M. Mramor, I. Bratko, and B. Zupan. Simple and effective visual models for gene expression cancer diagnostics. *Conference on Knowledge Discovery in Data*, strani 167–176, 2005.

[73] G. Leban, B. Zupan, G. Vidmar, and I. Bratko. VizRank: data visualization guided by machine learning. *Data Mining and Knowledge Discovery*, 13(2):119–136, 2006.

[74] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring N-dimensional databases. *Proceedings of the 1st conference on Visualization*, strani 230–237, 1990.

[75] E. Lee, D. Cook, S. Klinke, and T. Lumley. Projection pursuit for exploratory supervised classification. *Journal of Computational and Graphical Statistics*, 14(4):831, 2005.

[76] H. Levkowitz. Color icons: Merging color and texture perception for integrated visualization of multiple parameters. *Proceedings of IEEE Visualization'91, San Diego, CA*, 1991.

[77] H. Lohninger. INSPECT, a program system to visualize and interpret chemical data. *Chemometrics and intelligent laboratory systems*, 22(1):147–153, 1994.

[78] R. L. Mantaras. ID3 Revisited: A distance based criterion for attribute selection. *Proceedings of the International Symposium on Methodologies for Intelligent Systems*, 1989.

[79] B. H. McCormick, T. A. DeFanti, and M. D. Brown. Visualization in scientific computing – A synopsis. *IEEE Computer Graphics & Applications*, 7(7):61–70, 1987.

[80] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz. UCI repository of machine learning databases, 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[81] G. M. Nielson and L. Rosenblum. *Proceedings of IEEE Visualization 1991.* IEEE Computer Society Press, 1991.

[82] C. L. Nutt, D. R. Mani, R. A. Betensky, *et al.* Gene expression-based classification of malignant gliomas correlates better with survival than histological classification. *Cancer Research*, 63(7):1602–1607, 2003.

[83] A. Perez-Jimenez and J. C. Perez-Cortes. Genetic algorithms for exploratory data analysis. *Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, strani 743–751, 2002.

[84] R. M. Pickett and G. Grinstein. Iconographic displays for visualizing multidimensional data. *Proceedings of IEEE Conference on Systems, Man and Cybernetics*, strani 514–519, 1988.

[85] C. Posse. Projection pursuit discriminant analysis for two groups. *Communications in statistics. Theory and methods*, 21(1):1–19, 1992.

[86] C. R. Rao. The Utilization of Multiple Measurements in Problems of Biological Classification. *Journal of the Royal Statistical Society. Series B (Methodological)*, 10(2):159–203, 1948.

[87] H. Riedwyl and M. Schupbach. Siebdiagramme: Graphische darstellung von kontingenztafeln. technical report 12. *Institute for Mathematical Statistics, University of Bern, Bern, Switzerland*, 1983.

[88] H. Riedwyl and M. Schupbach. Parquet diagram to plot contingency tables. *Softstat '93: Advances In Statistical Software*, strani 293–299, 1993.

[89] J. Rissanen. A universal prior for integers and estimation by minimum description length. *The Annals of Statistics*, 11(2):416–431, 1983.

[90] M. Robnik-Šikonja and I. Kononenko. Theoretical and empirical analysis of ReliefF and RReliefF. *Machine Learning*, 53(1):23–69, 2003.

[91] Cleveland W. S. *Visualizing Data.* Hobart Press, 1993.

[92] S. Santini and R. Jain. The use of psychological similarity measure for queries in image databases. Technical report, Visual Computing Laboratory, University of California San Diego, 1996.

[93] S. Santini and R. Jain. Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):871–883, 1999.

[94] R. Sedgewick. *Algorithms in C: fundamentals, data structures, sorting, searching, and graph algorithms.* Addison-Wesley Professional, 2001.

[95] A. D. Shapiro. *Structured induction in expert systems.* Addison-Wesley Longman Publishing Co., 1987.

[96] M. A. Shipp, K. N. Ross, P. Tamayo, *et al.* Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, 8(1):68–74, 2002.

[97] H. Siirtola. Direct manipulation of parallel coordinates. In *CHI '00: CHI '00 extended abstracts on Human factors in computing systems*, strani 119–120, New York, NY, USA, 2000. ACM Press.

[98] D. Singh, P. G. Febbo, K. Ross, *et al.* Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2):203–209, 2002. TY - JOUR.

[99] R. D. Snee. Graphical display of two-way contingency tables. *The American Statistician*, 28(1):9–12, 1974.

[100] A. Statnikov, C. F. Aliferis, I. Tsamardinos, *et al.* A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, 21(5):631–643, 2004.

[101] M. C. Stone, K. Fishkin, and E. A. Bier. The moveable filter as a user interface tool. *Proceedings of ACM Conference On Human Factors in Computing Systems (CHI' 94), New York*, strani 306–312, 1994.

[102] D. F. Swayne, A. Buja, and D. T. Lang. Exploratory visual analysis of graphs in GGobi. In Jaromir Antoch, editor, *CompStat: Proceedings in Computational Statistics, 16th Symposium*. Physica-Verlag, 2004.

[103] D. F. Swayne, D. T. Lang, A. Buja, and D. Cook. GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis*, 43:423–444, 2003.

[104] P. Switzer. Numerical classification. *Geostatistics. Plenum Press, New York*, 1970.

[105] J. Trilk. Software visualization, 2001. http://www.broy.informatik.tu-muenchen.de/ trilk/sv.html.

[106] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley Menlo Park, CA, 1977.

[107] Y. Wang, I. V. Tetko, M. A. Hall, E. Frank, A. Facius, K. F. X. Mayer, and H. W. Mewes. Gene selection from microarray data for cancer classification – A machine learning approach. *Computational Biology and Chemistry*, 29:37–46, 2005.

[108] M. O. Ward. XmdvTool: Integrating multiple methods for visualizing multivariatedata. *Visualization, 1994., Visualization'94, Proceedings., IEEE Conference on*, strani 326–333, 1994.

[109] E. J. Wegman and Q. Luo. High dimensional clustering using parallel coordinates and the Grand Tour. *Computing Science and Statistics*, 28:352—-360, 1997.

[110] P. Wong and R. Bergeron. 30 years of multidimensional multivariate visualization. *Scientific Visualization - Overviews, Methodologies and Techniques*, strani 3–33, 1997.