Uporaba predznanja o povezanosti značilk pri gradnji napovednih modelov

Marko Toplak

doktorska disertacija

predana

Fakulteti za računalništvo in informatiko

kot del izpolnjevanja pogojev za pridobitev naziva

doktor znanosti

s področja

računalništva in informatike

Ljubljana, 2016

IZJAVA

Izjavljam, da sem avtor dela in da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali na drugem visokošolskem zavodu, razen v primerih, kjer so navedeni viri.

— Marko Toplak —

julij 2016

Oddajo so odobrili

dr. Marko Robnik-Šikonja
izredni profesor za računalništvo in informatiko
član ocenjevalne komisije

dr. Blaž Zupan
redni profesor za računalništvo in informatiko
mentor in član ocenjevalne komisije

dr. Sašo Džeroski
redni profesor za računalništvo in informatiko
zunanji član ocenjevalne komisije
Institut Jožef Stefan

PREDHODNA OBJAVA

Izjavljam, da so bili rezultati obravnavane raziskave predhodno objavljeni/sprejeti za objavo v recenzirani reviji ali javno predstavljeni v naslednjih primerih:

[1] Marko Toplak, Tomaž Curk, Janez Demšar, in Blaž Zupan. Does replication groups scoring reduce false positive rate in SNP interaction discovery? *BMC Genomics*, 11(1):1, 2010.

[2] Marko Toplak, Tomaž Curk, in Blaž Zupan. Similarity of transcription profiles for genes in gene sets. Predstavljeno na konferenci ter objavljeno v zborniku *International Conference Adaptive and Natural Computing Algorithms*, strani 393–399. Springer, 2011.

[3] Marko Toplak, Rok Močnik, Matija Polajnar, Zoran Bosnić, Lars Carlsson, Catrin Hasselgren, Janez Demšar, Scott Boyer, Blaž Zupan, in Jonna Stålring. Assessment of machine learning reliability methods for quantifying the applicability domain of QSAR regression models. *Journal of Chemical Information and Modeling*, 54(2):431–441, 2014.

Potrjujem, da sem pridobil pisna dovoljenja vseh lastnikov avtorskih pravic, ki mi dovoljujejo vključitev zgoraj navedenega materiala v pričujočo disertacijo. Potrjujem, da zgoraj navedeni material opisuje rezultate raziskav, izvedenih v času mojega podiplomskega študija na Univerzi v Ljubljani.

Univerza *v Ljubljani*
Fakulteta *za računalništvo in informatiko*
Marko Toplak
Uporaba predznanja o povezanosti značilk pri gradnji napovednih modelov

POVZETEK

Z ustreznim predznanjem lahko zgradimo točnejše napovedne modele. Eno od področij, kjer je gradnja napovednih modelov razmeroma težka zaradi malo učnih primerov v tipičnem naboru podatkov, a kjer imamo na voljo veliko predznanja, je področje molekularne biologije. Osnovne entitete na področju, geni, proteini ali presnovni produkti, so opisani in razvrščeni v kategorije v raznih prosto dostopnih bazah podatkov. Ta dodatna znanja lahko s pridom izkoristimo pri gradnji napovednih modelov. V disertaciji smo osredotočeni na metode, ki transformirajo prostor značilk v prostor skupin značilk, pri čemer skupine pridobimo iz obstoječih baz podatkov in predstavljajo predznanje.

Značilke na podatkovnih naborih s področja molekularne biologije, ki smo jih uporabljali v disertaciji, predstavljajo gene. Metode, ki obravnavajo skupine genov, temeljijo na predpostavki, da so izrazni profili genov, ki pripadajo isti skupini, podobni. V disertaciji to predpostavko potrdimo in pokažemo, da so pari izraznih profilov genov iz skupin v bazah KEGG in BioGRID bolj podobni kot pari izraznih profilov naključno izbranih genov, a tudi pokažemo, da so te razlike majhne. Razlike ostajajo enake ne glede na verzijo podatkovnih baz skupin.

V delu predlagamo metodo transformacije podatkov v prostor skupin značilk s sočasno matrično faktorizacijo, ki matriki podatkov in skupin značilk hkrati razcepi na produkt faktorjev z manjšimi rangi od rangov izvirnih matrik. Na resničnih podatkih po transformaciji s sočasno faktorizacijo lahko zgradimo modele, ki dosegajo primerljivo točnost kot modeli zgrajeni na netransformiranih podatkih. Predlagan pristop pri pretvorbi v prostor skupin uporabi tudi značilke, ki so podobne značilkam v skupini, a skupini ne pripadajo, kar ga loči od ostalih transformacijskih metod.

Pri transformaciji v prostor skupin značilk moramo nastaviti parametre transformacij, kot so uteži značilk iz skupine. Transformacije, ki pri izračunu parametrov

uporabljajo tudi ciljno spremenljivko, ustvarijo vrednosti skupin značilk, ki so vsaj deloma prilagojene ciljni spremenljivki. Učne metode bodo zato značilkam, ki so preveč prilagojene razredu, pripisale prevelik pomen, kar lahko poslabša točnost na novih podatkih. Predlagamo rešitev s skladanjem. Predlagana rešitev deluje z obstoječimi metodami transformacije v prostor skupin značilk in na nekaterih podatkovnih naborih bistveno izboljša točnost končnih napovednih modelov.

V disertaciji preučimo tehnike transformacije podatkov v prostor vnaprej definiranih skupin značilk. V največji študiji doslej pokažemo, da z gradnjo napovednih modelov na podatkih s področja molekularne biologije, ki smo jih transformirali z obstoječimi ali predlaganimi metodami, v povprečju ne izboljšamo točnosti napovednih modelov na netransformiranih podatkih. Točnosti napovednih modelov, ki jih zgradimo na transformiranih podatkih, so še vedno podobne točnostim na netransformiranih podatkih. Ker je modele na podatkih transformiranih v skupine značilk laže interpretirati, je transformacije v prostor skupin smiselno uporabiti.

Ključne besede: strojno učenje, predznanje, povezane značilke, sočasna matrična faktorizacija, skladanje, bioinformatika

University *of Ljubljana*
Faculty *of Computer and Information Science*
Marko Toplak
Induction of prediction models using domain knowledge about related features

ABSTRACT

Domain knowledge can help us build more accurate prediction models. Molecular biology is one of the fields where induction of prediction models is relatively hard due to few learning instances in a typical data set, but there exists vast domain knowledge. Basic entities of the field—genes, proteins, and metabolic products—are described and categorized in various freely accessible databases. This thesis focuses on methods that transform data from the space of features into the space of feature groups, which can be assembled from existing data bases and represent prior knowledge.

Features in data sets from the field of molecular biology that we used in the thesis represent genes. Methods working with gene groups assume that gene expression profiles belonging to the same group are similar. We show that gene expressions of gene pairs from groups in databases KEGG and BioGRID are more similar than gene expression of random gene pairs, but the differences are small. The differences do not change with the database version.

We propose a technique for transformation of data into a space of feature groups with collective matrix factorization, which simultaneously factorizes matrices representing data and feature groups into a product of latent factors with ranks smaller than ranks of original matrices. The models induced from the transformed data can be as accurate as models on the non-transformed data. In contrast to existing approaches, the proposed approach can also use features that are not in predefined groups of features but are similar to features in a group.

Transformation techniques that transform data into a space of feature groups require estimation of transformation parameters such as, for example, feature weights. Techniques that use values of the target variable for parameter estimation, produce values for the feature groups that are at least partially fitted to the target variable. The induced models could therefore overestimate the importance of class-overfitted features, which

can decrease their accuracy on novel data. We propose a solution that uses stacking. The proposed solution can work with any transformation technique and, for some data sets, boosts accuracy substantially.

In the thesis we throughly study transformation of data into predefined feature groups. We show, in the largest study so far, that, on average, models induced from data sets transformed with feature groups do not obtain better prediction accuracies than models induced on non-transformed data sets. As the accuracies on transformed and non-transformed data sets are similar, the transformed data may still be preferred as models on feature groups are easier to interpret.

Keywords: machine learning, domain knowledge, related features, collective matrix factorization, stacking, bioinformatics

ZAHVALA

Zahvaljujem se vsem sedanjim in bivšim članom Laboratorija za bioinformatiko, ki so soustvarjali prijetno in spodbudno okolje. Posebej se zahvaljujem mentorju Blažu Zupanu za uvajanje v raziskovalno delo, usmeritve in kritične komentarje, Minci Mramor, ki me je spoznala s področjem napovedovanja s skupinami značilk in mi z medicinskim znanjem pomagala pri ilustraciji interpretabilnosti, Marinki Žitnik za vzor dobro opravljenega znanstvenega dela in diskusije o matrični faktorizaciji ter Janezu Demšarju za večno otroško radovednost. Hvala tudi ožji in širši družini, ker vas je glede mojega doktorata že kar malo skrbelo in ste me spodbujali, naj ga vendarle zaključim. Disertacijo je podpirala in tudi zelo skrbno prebrala Agnieszka Rovšnik – hvala!

— Marko Toplak, Ljubljana, julij 2016.

KAZALO

I

Uvod

1.1 Motivacija

Na tekmovanju konference IJCNN 2007 so pokazali, da lahko v domenah, kjer imamo na voljo predznanje, zgradimo točnejše napovedne modele [1]. Tekmovalci so morali izdelati čim boljše napovedne modele za pet problemov iz različnih domen. Za vsako domeno so bili na voljo podatki dveh vrst. Tekmovalci, ki so tekmovali s predznanjem, so prejeli izvirne podatke, kjer so pomen značilk poznali. Nasprotno so tekmovalci, ki so tekmovali brez predznanja, prejeli pripravljeno tabelo podatkov z značilkami, katerih pomena niso poznali. Pri enem problemu so morali tekmovalci ustvariti model, ki zna ročno napisane števke razdeliti v dva razreda: na soda in liha števila. Tekmovalci s predznanjem so uporabljali slike števk, tekmovalci brez predznanja pa so imeli na voljo zgolj intenzivnosti posameznih točk slike, a v naključnem vrstnem redu, tako da slike niso mogli rekonstruirati. Štiri od petih problemov so tekmovalci s predznanjem bolje rešili. Opazimo še eno zanimivost: tekmovalci brez predznanja so na začetku tekmovanja hitreje prišli do svoje najboljše rešitve, tekmovalci s predznanjem pa so za dobro rešitev porabili več časa in so šele kasneje prehiteli slabše uvrščene tekmovalce brez predznanja. Ta nakazuje, da je predznanje težko ustrezno uporabiti.

Obstoječe predznanje bi lahko občutno pripomoglo k izboljšanju napovednih točnosti na področju molekularne biologije, konkretneje na področju nadzorovanega učenja na podlagi podatkov o genskih izrazih (angl. gene expression data) [2, 3]. Tam se poskušamo naučiti modela, ki bi za nek vzorec, opisan s tipično vsaj nekaj tisoč vrednostmi značilk (vsaka opisuje stopnjo izražanja nekega gena), lahko napovedal vrednost razreda, recimo prisotnost bolezni. Če bi znali zgraditi dobre napovedne modele, bi lahko izboljšali diagnostiko in prognostiko nekaterih bolezni. Pri raku, kjer je natančno diagnosticiranje tipa bolezni težko, bi lahko z boljšo diagnostiko terapijo lažje prilagajali posameznikom [4]. Ker je pridobivanje podatkov takega tipa razmerama drago in zamudno, je vzorcev glede na število značilk zelo malo, najpogosteje nekaj sto, kar gradnjo napovednih modelov otežuje.

Na podatkih o genskih izrazih so sprva ocenjevali stopnjo povezanosti posameznih genov s ciljno spremenljivko in tako dobili urejen seznam genov [5]. Tak seznam je težko interpretirati. Zaradi majhnega števila vzorcev glede na število genov in omejene natančnosti mikromrež mRNA, s katerimi podatke o genskih izrazih zajemamo, so rezultati nezanesljivi: če uporabimo druge vzorce ali zgolj zamenjamo laboratorij, ki zbira podatke, se lahko razvrstitev genov precej spremeni [6, 7].

Človeški geni in geni modelnih organizmov so opisani in razvrščeni v kategorije skladno z različnimi ontologijami v raznih prosto dostopnih bazah podatkov. Iz opisov v teh bazah lahko pridobimo skupine funkcijsko, lokacijsko ali procesno povezanih genov. V bazi KEGG [8] so glede na vlogo v metabolizmu geni razvrščeni v grafe, ki opisujejo presnovne poti. V projektu genske ontologije (angl. gene ontology, GO) [9] so gene strukturirali v skupine glede na to, ali njihovi produkti – proteini – sodelujejo v istih celičnih procesih, se pojavljajo v istih delih celice, ali opravljajo podobno funkcijo. Baze proteinskih interakcij, kot je STRING [10], opisujejo pare genov, ki kodirajo skupaj delujoče proteine. Baza OMIM opisuje gene glede na bolezni, pri katerih ti igrajo ključno vlogo [11]. O genih imamo torej na voljo veliko predznanja, ki bi ga bilo moč izkoristiti pri gradnji napovednih modelov tako za izboljšanje točnosti kot za lažjo podajo razlage fenomenov, katerih principe delovanja bi lahko izluščili iz podatkov.

Če ocenjujemo povezanost posameznih skupin s ciljno spremenljivko, dobimo urejene sezname genskih skupin, ki so zanesljivejši kot seznami z geni, poleg tega pa je sezname skupin lažje interpretirati [6, 12], ker skupine ponavadi opisujejo poznane biološke procese ali strukture [13]. Pričakujemo, da bodo genske skupine uporabne tudi pri napovedovanju ciljne spremenljivke: da lahko z njimi zgradimo točnejše in stabilnejše modele ter dobimo drugačen vpogled v podatke, ki ga je morda laže interpretirati. Kljub temu, da metode za napovedovanje z genskimi skupinami razvijajo že vsaj od leta 2005 [14], dosedanje primerjalne študije na več naborih podatkov poročajo, da z modeli, ki napovedujejo na podlagi genskih skupin, dosežemo zgolj primerljivo dobre in ne boljših rezultatov kot z napovedovanjem na podlagi posameznih genov [15–18].

1.2 Pregled disertacije

1.2.1 Podobnost izraznih profilov genov v genskih skupinah

Metode, ki obravnavajo skupine genov, temeljijo na predpostavki, da so izrazni profili genov, ki pripadajo isti skupini, povezani. Da bi ocenili, če predpostavka drži, smo izmerili podobnosti med izraznimi profili genov (stopnje izraženosti za nek gen čez več vzorcev) med geni iz genskih skupin iz vira presnovnih poti KEGG in iz vira interakcij BioGrid.

V poglavju predstavljamo analizo podobnosti med izraznimi profili genov v genskih skupinah ali interakcijah na velikem naboru podatkov. Podobnosti med izraznimi profili genov smo merili s Pearsonovim koeficientom korelacije in interakcijskim

prispevkom. Interakcijski prispevek je nadzorovana mera podobnosti, ki temelji na informacijski teoriji [19]. Uporabili smo veliko več naborov podatkov kot predhodne raziskave in dodatno mero podobnosti med izraznimi profili genov, ki upošteva vrednost razredne spremenljivke.

Naši rezultati kažejo, da so pari izraznih profilov genov iz skupin v bazah KEGG in BioGRID bolj podobni kot pari izraznih profilov naključno izbranih genov, kar potrjuje rezultate predhodnih raziskav [20–22]. Kljub temu, da smo lahko razlike med porazdelitvami ocen podobnosti zanesljivo opazili, so bile le-te precej majhne. Podobnosti genskih izrazov iz skupin preizkušenih baz ostajajo enake ne glede na verzije baz podatkov, a jih z večanjem števila skupin v posamezni bazi ocenimo kot statistično bolj značilne.

V poglavju predstavljamo naslednja prispevka k znanosti:

- Na velikem številu naborov podatkov smo pokazali, da so izrazni profili genov znotraj genskih skupin bolj podobni kot izrazni profili naključnih genov.
- Pokazali smo, da so novejše interakcije v bazi BioGRID glede na podobnosti izraznih profilov enakovredne starejšim.

1.2.2 *Sočasna matrična faktorizacija za napovedovanje s skupinami značilk*

Matrična faktorizacija razcepi matriko na produkt faktorjev z manjšimi rangi od izvirne matrike tako, da produkt čim bolje aproksimira izvorno matriko. Še posebej popularna je ta tehnika modeliranja podatkov postala po zmagi na tekmovanju priporočilnih sistemov Netflix prize [23]. V zadnjem času jo uporabljajo na raznih področjih, denimo za iskanje skupnosti v omrežjih [24]. Dobro se obnese tudi pri zlivanju podatkov [25, 26].

V poglavju predlagamo postopek za transformacijo učnih in testnih primerov iz prostora značilk v prostor skupin značilk s sočasno matrično faktorizacijo. Za faktorizacijo uporabimo algoritem za zlivanje podatkov DFMF [27], transformiramo pa z veriženjem razcepnih faktorjev [28].

Predlagano metodo ovrednotimo na umetno generiranih in resničnih podatkovnih naborih s področja molekularne biologije. Rezultati s klasifikacijo z logistično regresijo na resničnih naborih podatkov kažejo, da lahko s transformacijo s sočasno faktorizacijo zgradimo modele, ki dosegajo primerljivo točnost kot modeli zgrajeni na originalnih (netransformiranih) podatkih. Pri klasifikaciji z naključnimi gozdovi se matrična faktorizacija ni dobro obnesla. S sočasno faktorizacijo dobimo boljše rezultate kot z ločeno

faktorizacijo obeh vhodnih matrik, matriki podatkov in skupin značilk. Za razliko od drugih metod transformacije v prostor skupin značilk, sočasna faktorizacija pri pretvorbi v prostor skupin uporabi tudi značilke, ki so podobne značilkam v skupini, a skupini ne pripadajo. Pokažemo, da so dodatne značilke, ki jih sočasna matrična faktorizacija uporabi, smiselne.

V poglavju predstavljamo naslednja prispevka k znanosti:

- Predlagamo metodo za transformacijo vhodnih značilk v značilke, ki opisujejo skupine. Metoda temelji na sočasni matrični faktorizaciji in uporablja vir podatkov, ki opisuje skupine značilk.
- Pokazali smo, da predlagana metoda skupine značilk smiselno razširja s podobnimi značilkami iz izvornih podatkov.

1.2.3 Napovedovanje s skladanjem transformiranih vrednosti

Pri napovedovanju z vnaprejšnjo transformacijo iz prostora originalnih značilk v prostor skupin značilk moramo nastaviti parametre modelov, ki skupino značilk transformirajo v vrednost skupine. Takšni parametri so denimo uteži značilk iz skupine. Nekatere transformacije [15, 29–36] pri izračunu parametrov uporabijo tudi ciljno spremenljivko in se zato vsaj deloma prilagodijo ciljni spremenljivki. Učni algoritem, ki gradi napovedni model, po hkratni transformaciji celotnih učnih podatkov nato ne more razločiti, ali neka transformirana značilka dobro opisuje razred ali pa so se parametri transformacije pretirano prilagodili razredu. Učna metoda bo značilkam, ki so preveč prilagojene razredu, pripisala prevelik pomen, kar vodi do slabše napovedne točnosti na novih podatkih.

Večina obstoječih raziskav, ki se ukvarjajo s transformacijskimi metodami, ki uporabljajo ciljno spremenljivko, težave s preveliko prilagoditvijo ne komentira [16, 18, 34, 37], druge raziskave pa problem rešujejo z uporabo različnih delov učne množice podatkov za nastavitev parametrov transformacije in izbor transformiranih značilk [31, 33, 35]. Ob tem odstranijo značilke skupin, katerih transformacije dobro opišejo le podmnožico, ki smo jo uporabili za nastavitev parametrov transformacije.

Predlagamo metodo, ki problem ciljni spremenljivki preveč prilagojenih transformacij značilk rešuje s skladanjem (angl. stacking) [38] transformacij skupin. V nasprotju z obstoječimi predlogi za uporabo transformiranih značilk, vse transformirane značilke obdržimo vse do gradnje končnega napovednega modela. Pokazali smo, da skladanje

odločitvenim modelom lahko omogoči bolje oceniti kvaliteto značilk, vendar na resničnih podatkih razlike med skladanjem in transformacijo s celotno učno množico niso značilne. Opišemo tudi prilagoditev metode SetSig, po kateri lahko transformacijo s skladanjem izvedemo v enakem času, kot bi ga potrebovali za transformacijo brez skladanja.

V poglavju predstavljamo naslednja prispevka k znanosti:

- Predlagali in preizkusili smo uporabo skladanja za napovedovanje s transformiranimi značilkami transformacijskih metod, ki pri transformaciji uporabljajo vrednosti ciljne spremenljivke.
- Metodo SetSig smo prilagodili tako, da je časovna kompleksnost transformacije v prostor skupin značilk s skladanjem enaka časovni kompleksnosti transformacije brez skladanja.

1.3 Glavna prispevka k znanosti

Poudarili bi sledeča ključna prispevka disertacije:

- Poglobljeno smo preučili tehnike transformacije podatkov iz prostora značilk v prostor vnaprej definiranih skupin značilk. V največji študiji doslej smo pokazali, da z gradnjo napovednih modelov na transformiranih podatkih ne izboljšamo točnosti, še vedno pa dobimo zadovoljivo dobre napovedne modele, da jih je zaradi prednosti pri interpretaciji smiselno uporabljati.
- Predlagali smo metodo za transformacijo vhodnih značilk v značilke, ki opisujejo skupine. Metoda temelji na sočasni matrični faktorizaciji in uporablja vir podatkov, ki opisuje skupine značilk.

2

Pregled področja

Pri nadzorovanem strojnem učenju se iz označenih primerov, ki so predstavljeni z vrednostmi značilk in pripadajočo oznako ali razredom, poskušamo naučiti modela, ki zna določiti razred novim, dotlej nevidenim, primerom. Tipične metode za gradnjo napovednih modelov, kot so metoda najbližjih sosedov [39], metoda podpornih vektorjev (angl. support vector machines, SVM) [40] ali metoda naključnih gozdov (angl. random forests) [41], poleg samih podatkov ne upoštevajo nobenih dodanih informacij o učni domeni oziroma predznanja. Kot smo omenili v uvodu, lahko v domenah, kjer je predznanje na voljo, z njegovo pomočjo zgradimo točnejše napovedne modele [1, 42]

Pristopi globokega učenja (angl. deep learning) delujejo na podlagi različnih nivojev predstavitve. Nivoji v globokih nevronskih mrežah so medsebojno povezani s preprostimi nelinearnimi funkcijami, ki predstavitve nekega nivoja predelajo v vedno bolj abstraktne predstavitve [43]. V zadnjih letih globoko učenje dosega dobre rezultate na raznih področjih, na primer v računalniškem vidu, razpoznavanju govora in klasifikaciji besedil [44]. Globoke nevronske mreže lahko gradimo zaradi velike količine podatkov, ki nam je na voljo v sodobnem času. Medtem ko osnovni algoritmi za gradnjo napovednih modelov zahtevajo, da so učni primeri označeni, nekateri pristopi gradnje globokih nevronskih mrež za učenje vmesnih nivojev nevronskih mrež uporabljajo tudi neoznačene primere: iz njih se lahko naučijo predstavitve učnih podatkov, na kateri bo gradnja končnega napovednega modela lažja [44]. Pristopi globokega učenja običajno ne uporabljajo eksplicitne predstavitve predznanja, vendar si strukturo, ki na nek način opisuje predznanje, zgradijo na neoznačenih primerih. Lake et al. [45] so zasnovali algoritem za razpoznavanje pisave, ki daje dobre rezultate že z enim označenim učnim primerom na znak nove pisave, ki naj bi jo razpoznal, vendar so tako učinkovito učenje dosegli s poprejšnjim učenjem na 30 različnih abecedah, na katerih se je algoritem naučil splošnih modelov pisanja – predznanja.

V pričujoči disertaciji se z gradnjo predznanja iz neoznačenih ali iz sorodnih učnih primerov ne ukvarjamo. V nadaljevanju opisujemo zgolj algoritme, ki delujejo z eksplicitno podanim predznanjem, kar ima prednost pri interpretaciji modelov. Medtem ko globoko učenje temelji na dostopnosti izjemno velikih podatkovnih baz, se v našem delu osredotočamo na analizo manjših, danes dostopnih baz s področja molekularne biologije.

2.1 Napovedovanje s predznanjem o povezanih značilkah

V nekaterih primerih lahko metodam gradnje napovednih modelov pomagamo, če značilke predelamo v ustrezno obliko ali pa izberemo ustrezno podmnožico značilk. Izbor podmnožice značilk lahko opravimo s filtriranjem glede na mero, ki značilke vrednoti bodisi ločeno, kot je razmerje informacijskega prispevka (angl. gain-ratio) [46], bodisi v kontekstu ostalih, kot je ReliefF [47]. Alternativni način izbora je izbor po principu ovojnice, kjer značilke dodajamo ali odstranjujemo iz izbrane množice glede na kakovost odločitvenega modela na izbranih značilkah [48]. Ko so značilke tako povezane, da nam skupaj podajajo več informacije, kot bi nam je ob posamični obravnavi [19], se nam lahko izplača, če iz njih tvorimo nove značilke. Primer sistema za tvorbo novih značilk je HINT [49].

Pri gradnji napovednih modelov s predznanjem o povezanih značilkah imamo na vhodu dva nabora podatkov različnih tipov. Primarni nabor podatkov v obliki primerov, ki so opisani z vektorji vrednosti značilk, opisuje objekte, katerih lastnosti oziroma vrednosti razredne spremenljivke želimo modelirati in napovedovati novim primerom. Dodatni nabor podatkov vsebuje podatke o povezanih skupinah značilk, ki predstavljajo predznanje in jih lahko pridobimo neodvisno od primarnega nabora podatkov. Cun in Fröhlich [50] razdelita pristope za napovedovanje s predznanjem v obliki skupin ali grafov značilk v dve skupini:

1. V skupino na podatke osredotočenih pristopov (angl. data centric), ki gradijo napovedni model na značilkah originalnih podatkov, vendar gradnjo modela usmerjajo tako, da upošteva predznanje.

2. V skupino na predznanje osredotočenih pristopov (angl. network centric, originalni izraz smo posplošili), ki pretvorijo originalne podatke v nov prostor, ki ga določa predznanje, in nato znotraj novega prostora gradijo napovedne modele.

S hkratno analizo več virov podatkov se ukvarja področje zlivanja podatkov (angl. data fusion), ki ga je v doktorski disertaciji temeljito obdelala Marinka Žitnik [26]. Pavlidis et al. [51] so pristope zlivanja podatkov razdelili v tri skupine: v (1) zgodnje združevanje (angl. early integration), (2) pozno združevanje (angl. late integration) in (3) vmesno združevanje (angl. intermediate integration). Zgodnje združevanje vse podatke združi v eno veliko matriko in na njej zgradi napovedni model. Pozno združevanje

za vsak vir podatkov zgradi svoj model ter združi le napovedi. Pri vmesnem združevanju je faza modeliranja prilagojena tako, da se razni viri podatkov uporabijo znotraj gradnje modela, vendar tako, da tega ne moremo opisati le z združevanjem matrik. Na pristope za napovedovanje s predznanjem o povezanih značilkah gledamo kot na specializirano zlivanje podatkov. Glede na delitev po Pavlidis et al. [51] vsi v nadaljevanju opisani pristopi spadajo med pristope vmesnega združevanja. Prispevek Pavlidis et al. [51] opisuje soroden problem večličnega učenja (angl. multi-view learning) [52], ki hkrati obravnava več naborov podatkov, kjer vsak nabor z drugimi značilkami opisuje iste primere.

Predznanje o povezanih značilkah lahko strukturirano opišemo v obliki:

1. Skupin značilk, kjer so si značilke v skupini enakovredne.

2. Grafov, kjer značilke predstavljajo vozlišča grafa. Povezave grafa so lahko utežene.

Tako predznanje imenujejo znanje o povezanosti (angl. relevance knowledge) oziroma o bližini (angl. proximity knowledge; Liu in Motoda [53, 17. poglavje]).

V doktorski nalogi smo se omejili na na predznanje osredotočene pristope, ki uporabljajo predznanje v obliki podatkov o skupinah značilk. Taki pristopi so še posebej zanimivi, ker nam omogočajo interpretacijo kasneje zgrajenih napovednih modelov v prostoru genskih skupin, ki jih je laže interpretirati kot modele zgrajene v prostoru posameznih genov [54]. V nadaljevanju razdelka na kratko opišemo nekaj na podatke osredotočenih pristopov in na predznanje osredotočenih pristopov, ki uporabljajo grafe značilk, temeljiteje pa se posvetimo na predznanje osredotočenim pristopom, ki uporabljajo skupine značilk.

2.1.1 *Na podatke osredotočeni pristopi*

Na podatke osredotočeni pristopi na primarnih podatkih direktno zgradijo napovedni model, vendar gradnjo modela usmerjajo tako, da ob tem smiselno upoštevajo kot predznanje podane grafe ali skupine značilk.

Induktivno logično programiranje Če podatke predstavimo kot predikate, kar lahko storimo tako za primarne podatke kot za podatke o povezanih skupinah, lahko za napovedovanje uporabimo induktivno logično programiranje (ILP) [55]. ILP zgradi

klasifikacijski model kot računalniški program, ki uporablja v obliki predikatov predstavljene podatke.

Ryeng in Alsberg [56] sta kot predznanje v obliki predikatov predstavila gensko ontologijo. Zaradi splošnosti predikatnega zapisa sta lahko predstavila tudi relacije med razredi genske ontologije. Ker so pristopi ILP zaradi velike izrazne moči počasni, sta morala avtorja pred uporabo predlagane metode podatke zmanjšati z izborom genov.

Trajkovski et al. [57] z induktivnim logičnim programiranjem sicer niso gradili napovednih modelov, ampak so ocenjevali povezanost posameznih genskih skupin s ciljno spremenljivko. Kot vir predznanja so dodali še podatke o interakcijah med geni in pokazali, da njihov pristop koristno uporabi tako skupine iz genske ontologije kot interakcije. Algoritem so preizkušali na majhnem izboru genov.

Izbor značilk Johannes et al. [58] so združili metodo podpornih vektorjev, kjer so značilke rekurzivno odstranjevali s SVM-RFE [59], in algoritmu PageRank [60] podobno določanje pomembnosti vozlišč v grafu značilk. Na vsakem koraku so posameznim značilkam (genom) v grafu določili uteži, ki so jih izračunali iz podatkov o genskih izrazih z metriko povezanosti posamezne značilke z razredno spremenljivko. Nato so z algoritmom GeneRank [61] uteži rangirali ter odstranili 10% najslabše ocenjenih značilk glede na kombinacijo uteži SVM in ranga, ki ga je vrnil GeneRank.

Cun in Fröhlich [62] sta glede na podan graf značilk izračunala zglajeno statistiko t posamezne značilke, s permutacijskim testom izbrala nekaj najbolje ocenjenih značilk in z njimi zgradila končni model.

Razširitev regularizacije Predznanje so pogosto poskušali upoštevati s prilagoditvijo regularizacije obstoječih učnih metod. Tibshirani et al. [63] so linearno regresijo z $\ell 1$ regularizacijo oziroma Lasso [64] razširili s členom, s katerim dosežejo, da se določeni pari regresijskih koeficientov ne razlikujejo preveč. Sprejmejo lahko predznanje v obliki neusmerjenega grafa značilk. Predlagano metodo so poimenovali Fused Lasso.

Li in Li [65] sta regularizacijo $\ell 1$ linearne regresije razširila z regularizacijo posameznih značilk glede na uteži povezav v grafu značilk. V njunem predlogu bosta značilki, ki imata visoko vrednost uteži povezave, dodatek regularizacijskemu členu najmanj povečali, če je razlika med pripadajočima koeficientoma linearne regresije čim manjša.

Takeuchi et al. [66] so posplošili metodo Fused Lasso [63], tako da lahko namesto zgolj parov značilk upošteva skupine značilk, za katere želimo, da imajo podobne vrednosti regresijskih koeficientov. Njihov algoritem omogoča tudi delo s prekrivajočimi

skupinami.

Zhu et al. [67] so razširili regularizacijo podpornih vektorjev, tako da ta lahko izbira skupine značilk: predlagajo regularizacijo celotne skupine z regularizacijo $\ell 1$ glede na največji koeficient značilke v skupini.

Druge prilagoditve učnih metod Rapaport et al. [68] so razvili jedrno funkcijo za metodo podpornih vektorjev, ki glede na predznanje, podano v grafu značilk, pri računanju podobnosti upošteva le po Fourierjevi transformaciji dobljene nizkofrekvenčne komponente, visokofrekvenčne pa odstrani, ker predstavljajo šum.

Lavi et al. [69] metodi podpornih vektorjev dodajo regularizacijo glede na podan graf značilk, ki sosednji značilki v grafu kaznuje za kvadrat razlike njunih uteži znotraj metode podpornih vektorjev. Regularizacijo formulirajo kot novo jedro za linearni SVM.

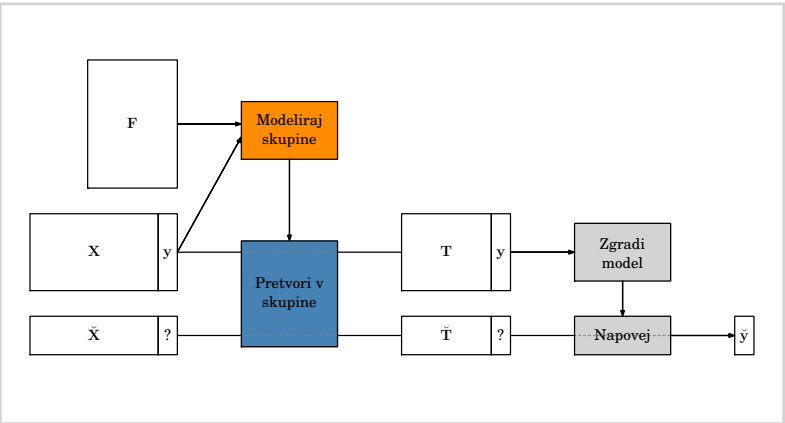
Anděl et al. [70] so omejili gradnjo naključnih gozdov (angl. random forests) [41] glede na podan graf značilk. Prilagojeni naključni gozdovi pri gradnji posameznih dreves značilk, po kateri delijo primere, ne izbirajo iz naključne podmnožice vseh značilk, temveč izbor utežijo glede na oddaljenost in strukturo grafa. Za računanje uteži uporabljajo naključne sprehode (angl. random walk).

2.1.2 *Na predznanje osredotočeni pristopi*

Če želimo predznanje uporabiti s standardnimi algoritmi za strojno učenje, ki uporabljajo učne primere predstavljene v atributnem prostoru, moramo značilke s konstruktivno indukcijo preoblikovati v nove značilke [71], ki predznanje upoštevajo. Po delitvi, ki jo je zasnoval Kramer [72], lahko značilke gradimo na podlagi analize učnih podatkov (angl. data-driven), zgrajenih hipotez (angl. hypothesis-driven) ali podanega predznanja (angl. knowledge-driven constructive induction).

Primer pristopa, ki kot vir predznanja uporablja grafe značilk, so razvili Chuang et al. [73]. Njihov pristop poskuša izbrati povezane dele grafa tako, da požrešno ponavlja postopek, kjer množici trenutno izbranih značilk na vsakem koraku doda tisto značilko izmed sosednjih značilk trenutno izbranega dela, ki najbolj poveča povezanost povprečne vrednosti izbranih značilk z razredom.

Na predznanje osredotočene pristope, ki uporabljajo skupine značilk in so jedro disertacije, opisujemo v naslednjem razdelku.



Slika 2.1

Napovedovanje z značilkami skupin. Skupine iz **F** modeliramo na množici učnih podatkov **X**. Učno množico pretvorimo v prostor skupin in na tako transformiranem naboru podatkov zgradimo napovedni model. Tudi testne primere $\check{\mathbf{X}}$ moramo transformirati pred napovedovanjem ciljne spremenljivke.

2.2 *Napovedovanje z značilkami, ki opisujejo skupine*

Naj primarna podatkovna množica $\mathbf{X} \in \mathbb{R}^{m \times n}$ vsebuje m primerov $\mathbf{x} \in R^n$ v vrsticah, kjer je vsak primer opisan z vrednostmi n značilk: x_{ij} je vrednost j-te značilke i-tega primera. Vrednosti razredne spremenljivke so zapisane v vektorju $\mathbf{y} \in \{0,1\}^m$; y_i je vrednost razreda i-tega primera. V prostor skupin značilk transformiran primer $\mathbf{x}$ je vektor transformiranih vrednosti za vsako skupino značilk G iz množice skupin $\mathbf{F}$, torej $\mathbf{t}(\mathbf{x}) = (t_G(\mathbf{x}) : G \in \mathbf{F})$. Transformacijo v prostor skupin za nek primer $\mathbf{x}$ in skupino značilk G označimo s $t_G(\mathbf{x})$. Transformacije $\mathbf{t}(\mathbf{x})$ za vse $\mathbf{x}$ iz učnih podatkov $\mathbf{X}$ tvorijo transformirano matriko podatkov $\mathbf{T}$ (slika 2.1).

Množico primerov z razredom c bomo označili $\mathbf{X}^{y=c}$: $\mathbf{X}^{y=c} = \{\mathbf{x}_i : y^{(i)} = c\}$. Z nadpisanim (G) označimo izbor značilk iz skupine: $\mathbf{X}^{(G)}$ na primer predstavlja $\mathbf{X}$ z le tistim značilkami, ki so v skupini G.

Po transformaciji lahko uporabimo katerikoli standardni algoritem za gradnjo napovednih modelov, denimo logistično regresijo, metodo najbližjih sosedov [39], metodo podpornih vektorjev (angl. support vector machines, SVM) [40] ali naključne gozdove (angl. random forests) [41]. Z metodami za gradnjo napovednih modelov se v disertaciji sicer ne ukvarjamo.

Klasifikatorje bi lahko gradili tudi na originalnih in transformiranih značilkah skupaj. V disertaciji smo se osredotočili na pretvorbo v prostor skupin značilk in se s

kombiniranjem originalnih in transformiranih značilk ne ukvarjamo. Glede na preliminarne rezultate z naivnim kombiniranjem, kjer smo transformirane in originalne značilke pred gradnjo klasifikatorjev združili v eno tabelo, bi nam to v disertacijo dodalo nov nivo kompleksnosti, saj so razlike med metodami manjše, same metode so pa tudi drugače razvrščene.

2.2.1 *Aritmetična sredina in mediana*

Skupino značilk lahko opišemo z aritmetično sredino ali mediano vrednosti značilk skupine [14]:

$$t_G^{\text{aritmetična sredina}}(\mathbf{x}) = \operatorname{mean} \mathbf{x}_G \quad \text{in} \tag{2.1}$$

$$t_G^{\text{mediana}}(\mathbf{x}) = \operatorname{median} \mathbf{x}_G. \tag{2.2}$$

2.2.2 *Analiza glavnih komponent (PCA)*

Guo et al. [14] so predlagali uporabo analize glavnih komponent (angl. principal component analysis, PCA). Vrednost $t_G^{\text{PCA}}(\mathbf{x})$, ki za nek primer $\mathbf{x}$ opiše skupino G, je odmik primera v smeri, v kateri se primeri najbolj razlikujejo, če pri računanju opazujemo le značilke iz skupine G. Transformirana vrednost $t_G^{\text{PCA}}(\mathbf{x})$ je 0, če so vrednosti značilk iz G primera $\mathbf{x}$ enake povprečnim vrednostim značilk iz G; pozitivne (ali negativne) vrednosti predstavljajo odmik v smeri (ali nasprotni smeri) največje variacije vrednosti značilk iz skupine. Za vsako skupino značilk uporabimo le prvo glavno komponento.

Najprej izračunamo centrirano $\mathbf{X}^{(G)}$ tako, da vsakemu primeru odštejemo povprečje vseh primerov: $\dot{\mathbf{x}}^{(G)} = \mathbf{x}^{(G)} - \mathbf{x}^{(G)}_{\text{mean}}$, kjer je $\mathbf{x}^{(G)}_{\text{mean}}$ vektor povprečij značilk iz G. Naj bo $\mathbf{v}_1$ lastni vektor, ki ustreza največji lastni vrednosti centrirane kovariančne matrike $(\dot{\mathbf{X}}^{(G)})^T \dot{\mathbf{X}}^{(G)}$.

Vrednost skupine značilk izračunamo kot skalarni produkt centriranega primera $\mathbf{x}$ in lastnega vektorja $\mathbf{v}_1$:

$$t_G^{\text{PCA}}(\mathbf{x}) = (\mathbf{x}^{(G)} - \mathbf{x}^{(G)}_{\text{mean}})\mathbf{v}_1. \tag{2.3}$$

2.2.3 *Analiza glavnih komponent z izborom značilk (SPCA)*

Analiza glavnih komponent z izborom značilk najprej izbere tiste značilke iz skupine, ki so dobro povezane z razredno spremenljivko in nato uporabi metodo PCA [29, 30].

Značilke izbiramo glede na statistiko t [30]. Za značilko j dobimo $t_j = t\text{-statistika}(\{x_j : \mathbf{x} \in \mathbf{X}^{y=0}\}, \{x_j : \mathbf{x} \in \mathbf{X}^{y=1}\})$. Naj filtrirana skupina G' vsebuje zgolj značilke, za katere

statistika t preseže mejo τ: $G' = \{j \in G : |t_j| > \tau\}$. Potem je

$$t_G^{\text{SPCA}}(\mathbf{x}) = t_{G'}^{\text{PCA}}(\mathbf{x}). \tag{2.4}$$

Mejo τ bi v idealnem primeru morali nastaviti z notranjim prečnim preverjanjem. V naših poskusih smo τ nastavili kot vrednost, ki ustreza $p = 0.01$ permutacijskega testa.

Tukaj bi opozorili, da statistika t ni ustrezna mera povezanosti z razredom, če porazdelitev vrednosti značilk za posamezni razred ne ustreza normalni porazdelitvi. Študije, ki se ukvarjajo s podatki mikromrež mRNA, tipično predpostavljajo, da so rezultati statistike t smiselni [29, 30, 32, 33]. Namesto statistike t bi lahko uporabili tudi kako drugo mero, denimo ReliefF [47].

2.2.4 *Delni najmanjši kvadrati (PLS)*

Metoda delnih najmanjših kvadrov (angl. partial least squares regression, PLS) je podobna analizi glavnih komponent, le da prva latentna komponenta PLS ustreza smeri, ki maksimizira kovarianco med značilkami in oznakami razredov (tukaj opisujemo PLS za eno razredno spremenljivko). Transformirana vrednost primera je položaj primera glede na to smer [31, 74].

Pri izračunu najprej centriramo $\mathbf{X}^{(G)}$ (kot pri PCA) in vektor vrednosti razredne spremenljivke $\mathbf{y}$: ($\dot{\mathbf{y}} = \mathbf{y} - \text{mean } \mathbf{y}$). Prva latentna komponenta za primer $\mathbf{x}$ in skupino značilk G je $\mathbf{w_1} = (\dot{\mathbf{X}}^{(G)})^T \dot{\mathbf{y}} / \|(\dot{\mathbf{X}}^{(G)})^T \dot{\mathbf{y}}\|$. Nadaljujemo kot pri t^{PCA}:

$$t_G^{\text{PLS}}(\mathbf{x}) = (\mathbf{x}^{(G)} - \mathbf{x}_{\text{mean}}^{(G)})\mathbf{w}_1. \tag{2.5}$$

2.2.5 *Analiza genskih skupin (GSA)*

Metoda analize genskih skupin (angl. gene set analysis, GSA) [32] za vsako značilko v skupini ugotovi, s katerim razredom je višja vrednost tiste značilke bolj povezana, in ustvari podskupine značilk glede na bolj povezan razred. Transformirana vrednost skupine je povprečna vrednost značilk močnejše podskupine.

Za značilko j iz skupine G izračunamo statistiko t: $t_j = t\text{-statistika}(\{x_j : \mathbf{x} \in \mathbf{X}^{y=0}\}, \{x_j : \mathbf{x} \in \mathbf{X}^{y=1}\})$. Vrednosti statistike t transformiramo v vrednosti statistike z, $z_j = \Phi^{-1}(F_{n-2}(t_j))$, kjer Φ označuje porazdelitveno funkcijo normalne porazdelitve in F_{n-2} označuje porazdelitveno funkcijo porazdelitve t z dvema prostostnima stopnjama.

Naj bosta G_+ in G_- podmnožici genov iz G s pozitivnimi in negativnimi vrednostmi z_j (povezani s prvim oziroma z drugim razredom). Moč povezave podskupine z razredom izračunamo kot

$$\overline{z_+} = \operatorname*{mean}_{j \in G_+} z_j \quad \text{in} \quad \overline{z_-} = \operatorname*{mean}_{j \in G_-} z_-.$$

Transformirana vrednost je povprečna vrednost značilk močnejše skupine:

$$t_G^{\mathrm{GSA}}(\mathbf{x}) = \begin{cases} \operatorname*{mean}_{j \in G_+} x_j & ; \overline{z_+} \geqslant \overline{z_-} \\ \operatorname*{mean}_{j \in G_-} x_j & \text{sicer} \end{cases}. \tag{2.6}$$

2.2.6 SetSig

Metoda SetSig [15] za primer $\mathbf{x}$, ki ga želimo transformirati v vrednosti skupine značilk G, izračuna Pearsonove koeficiente korelacije do primerov iz obeh razredov, ob čemer upošteva le značilke iz skupine G:

$$R_0(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=0}\} \quad \text{in} \quad R_1(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=1}\},$$

kjer sta $\mathbf{x}^{(G)}$ in $\mathbf{x}'^{(G)}$ primera $\mathbf{x}$ in $\mathbf{x}'$ opisana zgolj z vrednostmi značilk iz skupine G. Transformirana vrednost primera $\mathbf{x}$ za skupino G je s statistiko t ocenjena razlika med vrednostmi obeh množic:

$$t_G^{\mathrm{SetSig}}(\mathbf{x}) = t\text{-statistika}(R_0(\mathbf{x}; G), R_1(\mathbf{x}; G)). \tag{2.7}$$

2.2.7 Aktivnost glede na odzivne gene (CORG)

Metoda CORG [33] izračuna povprečje podmnožice značilk iz skupine, katerih vrednosti se najbolj razlikujejo med razredi. Povezanost značilke z razredom merijo s t-testom. Za vsako skupino značilke požrešno dodajajo v izbor (v vrstnem redu glede na povezanost značilke z razredom), dokler se statistika t povprečja vrednosti značilk izbora ne neha izboljševati. Takšnemu izboru značilk iz skupine rečejo "na stanje odzivni geni" (angl. condition-responsive genes, CORG).

Aktivnost (pod)skupine značilk G za vzorec $\mathbf{x}$ definirajo kot $A(\mathbf{x}, G) = \operatorname{mean} \mathbf{x}^{(G)}/\sqrt{|G|}$. Kvaliteto podmnožice $Q(G)$ izračunajo s primerjavo aktivnosti $A(\mathbf{x}, G)$ vseh primerov glede na razred s statistiko t. Posamezne značilke iz skupine $g \in G$ najprej uredijo glede

na $Q(\{g\})$, in to padajoče, če je mean$_{g\in G} Q(\{g\})$ pozitiven, sicer pa naraščajoče. Začnejo s prazno množico genov $G' = \emptyset$. Nato gene dodajajo, dokler se $Q(G')$ povečuje. Ocena skupine značilk je

$$t_G^{\text{CORG}}(\mathbf{x}) = A(x, G'), \tag{2.8}$$

kjer je G' množica na stanje odzivnih genov.

2.2.8 *Redke linearne napovedi (SpLin)*

Prej predstavljeni transformaciji PCA in PLS iz prostora značilk v prostor skupin pretvarjata z linearnimi kombinacijami vrednosti značilk. Metoda SPCA temu doda še izbor značilk, vendar sta pri SPCA fazi izbora značilk in določanja uteži značilke v linearni kombinaciji ločeni. Vse opisane transformacije so značilke izbirale neodvisno od drugih značilk v skupini.

Wu et al. [34] so predlagali metodo, ki elegantno izbere in uteži značilke v skupini znotraj enega optimizacijskega problema. Metodo, ki jo sicer uporabijo za izračun obogatenosti skupine genov čez vse primere, lahko uporabimo tudi za transformacijo vrednosti za posamezni primer $\mathbf{x}$, če vrednosti značilk pomnožimo s pripadajočimi utežmi modela skupine značilk:

$$t_G^{\text{SpLin}}(\mathbf{x}) = \sum_{g\in G} w_g^G x_g, \tag{2.9}$$

kjer so w_g^G uteži, ki nam jih vrne metoda za gradnjo linearnega napovednega modela na podatkih z značilkami iz skupine G, $\mathbf{X}^{(G)}$. Če uporabimo $\ell 1$ regularizacijo, ki večino uteži nastavi na 0, implicitno dobimo še izbor značilk. Za pretvorbo smo uporabili logistično regresijo [75] z $\ell 1$ regularizacijo, ki jo uporablja algoritem Lasso [64].

V poskusih smo za vse skupine značilk uporabili enako stopnjo regularizacije: $C = 1$ v nastavitvah knjižnice LibLinear [76]. Bolje, a časovno bolj potratno, bi bilo vrednosti za vsako skupino posebej nastaviti z notranjim prečnim preverjanjem.

2.2.9 *Druge metode*

Transformacija, ki jo predlagajo Su et al. [35], izračuna podporo skupine obema razredoma, ob čemer predpostavlja neodvisnost med značilkami in ločene Gaussove porazdelitve vrednosti za vsako značilko in razred. Avtorji za vsako značilko in razred izračunajo verjetje ter s seštevkom normaliziranih logaritmiranih razmerij verjetij obeh razredov iz skupine izračunajo transformirano vrednost skupine značilk.

Metoda ASSESS [36], po vzoru metode GSEA [6], na urejenem seznamu korelacij značilk z razredom oceni skupine značilk s statistiko podobno statistiki Kolmogorov-Smirnov. Medtem ko GSEA ocenjuje korelacije značilk z razredom čez vse primere, ASSESS ocenjuje korelacije za vsak primer posebej. Vrednost transformirane skupine značilk je največja vrednost na nekem koraku seštevanja korelacij, kjer seštevamo v vrstnem redu urejenega seznama korelacij vseh značilk in prištevamo le korelacije značilk iz trenutne skupine, pri drugih značilkah pa prištevamo (oziroma odštevamo) takšno konstanto, da na celem seznamu dobimo rezultat enak 0.

Li [77] predlaga transformacijo skupin značilk v manjše skupine z zmanjševanjem dimenzionalnosti z metodo SDR [78], ki transformira značilke v nižje dimenzionalni prostor, ob čemer poskuša ohraniti čim več informacije o razredu.

Hwang [79] predlaga računanje aritmetične sredine genov iz genske skupine, ob čemer uporabi 50% genov iz skupine, ki imajo najvišje vrednosti statistike t.

2.3 Primerjava metod za transformacijo v prostor skupin

Metode opisane v poglavju 2.2, ki transformirajo podatke iz prostora značilk v prostor skupin, smo preizkusili na resničnih podatkih o genskih izrazih ter jih primerjali s klasifikacijo na netransformiranih podatkih.

2.3.1 Podatki

Podatki o genskih izrazih Uporabili smo 42 naborov podatkov o človekovih genskih izrazih. Vsak nabor podatkov je vseboval vsaj 20 primerov (vzorcev), ki so bili razporejeni v dva diagnostična razreda (vsak razred je vseboval vsaj 8 vzorcev). Uporabili smo podatke iz dveh virov: iz Gene Expression Omnibus (GEO) [80] (imena teh podatkovnih naborov se začnejo z GDS) ter iz The Broad Institute[1]; našteti so v tabeli 2.1. V povprečju so podatki vsebovali 66 primerov. Če so podatki vsebovali več meritev za isti gen, smo uporabili njihovo mediano. Vse stolpce smo standardizirali, da so imeli povprečje enako 0 in varianco enako 1.

Viri skupin značilk Skupine značilk smo pridobili iz baze MSigDB [6] verzije 3.0: uporabili smo vire z oznakami C2.CP, C5.BP in C5.MF.

[1] Podatkovni nabori iz The Broad institute so opisani na http://www.biolab.si/supp/bi-cancer/ .

2.3.2 Učne metode

Primarne podatke z značilkami, ki so predstavljale gene, in iste podatke transformirane v prostor skupin značilk smo klasificirali z regularizirano logistično regresijo in naključnimi gozdovi. Regularizacijski parameter knjižnice LibLinear [76] smo z notranjim prečnim preverjanjem nastavljali na vrednosti iz množice $\{2^i; i \in \{-10, -8, \ldots 8, 10\}\}$. Naključni gozdovi [41] so vsebovali 1000 dreves; v vsakem vozlišču so značilke za delitev izbirali iz naključno izbrane množice velikosti korena vseh značilk in med njimi izbrali najustreznejšo glede na razmerje informacijskega prispevka (gain-ratio) [46].

2.3.3 Testiranje

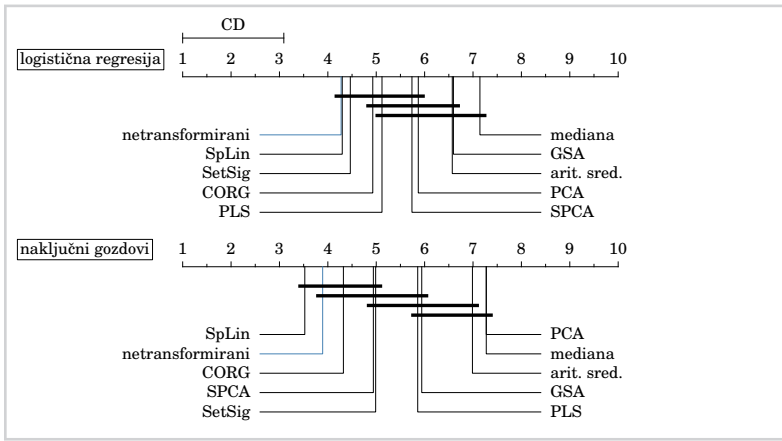
Testirali smo s 5-kratnim prečnim preverjanjem, ki smo ga pognali štirikrat. Točnost smo ovrednotili s površino pod krivuljo ROC (AUC [81]). Stabilnost smo izračunali z mero, ki jo je definirala Kuncheva [82]: kot povprečje ujemanja najbolje ocenjenih k značilk na vseh parih različnih učnih množic nabora podatkov, kjer ujemanje med paroma izračunamo kot $\frac{rn-k^2}{k(n-k)}$; r označuje velikost preseka najbolje ocenjenih značilk para različnih učnih množic. Pri izračunu stabilnosti smo izbrali $k = 100$ značilk z največjo absolutno vrednostjo koeficienta logistične regresije.

Pri vseh poskusih v disertaciji smo uporabljali paket za odkrivanje znanja iz podatkov Orange [83]. Za poenotenje imen genov iz različnih virov smo uporabili dodatek Orange Bioinformatics[2], znotraj katerega smo implementirali preizkušene transformacijske metode.

2.3.4 Rezultati in diskusija

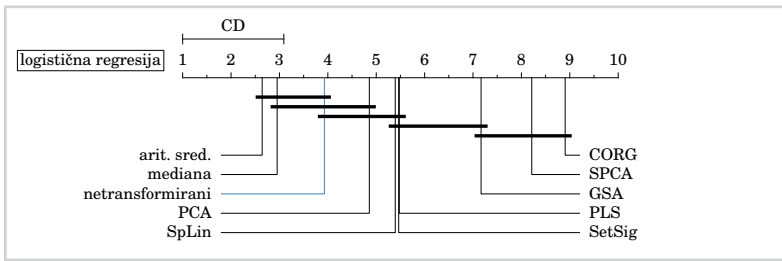
V povprečju je logistična regresija najtočnejše klasifikatorje zgradila na podatkih transformiranih z metodami SpLin, SetSig in CORG (slika 2.2), a na transformiranih podatkih z nobeno metodo ne dobimo boljših rezultatov kot na netransformiranih. Pri gradnji klasifikatorjev z naključnimi gozdovi je med najboljšimi tremi transformacijskimi metodami namesto metode SetSig metoda SPCA, klasifikatorji zgrajeni na podatkih transformiranih z metodo SpLin pa celo premagajo klasifikatorje na netransformiranih podatkih (slika 2.2). Opazimo lahko, da so metode, ki pri gradnji modelov transformacije uporabljajo tudi vrednosti ciljne spremenljivke, z izjemo metode GSA pri logistični

[2] https://github.com/biolab/orange-bio/



Slika 2.2

Povprečni rangi napovedne točnosti čez 42 podatkovnih naborov. Končne klasifikatorje smo gradili z linearno regresijo ali naključnimi gozdovi. CD označuje kritično razdaljo, znotraj katere razlike med povprečnimi rangi končnih klasifikatorjev niso značilne glede na Nemenyi-ev test ($\alpha = 0.05$).



Slika 2.3

Povprečni rangi stabilnosti čez 42 podatkovnih naborov. Končne klasifikatorje smo gradili z linearno regresijo. CD označuje kritično razdaljo, znotraj katere razlike med povprečnimi rangi končnih klasifikatorjev niso značilne glede na Nemenyi-ev test ($\alpha = 0.05$).

regresiji, bolje uvrščene kot metode, ki ciljne spremenljivke ne uporabljajo. Skoraj nobena razlika ni statistično značilna, kar je lahko posledica nizke moči testa, kjer smo paroma primerjali 10 metod. Rezultate na posameznih naborih prikazujeta tabeli 2.1 in 2.2.

Naši rezultati so v skladu z rezultati drugih medsebojnih primerjav metod za napovedovanje s predznanjem o skupinah in grafih, ki poročajo, da trenutne metode z uporabo genskih skupin ne izboljšajo napovednih točnosti, a le-te ostanejo primerljive s točnostmi na podatkih o posameznih genih [15–18]. Rezultati se razlikujejo od rezultatov predstavitev posameznih metod [32, 33, 36], kjer so na manjšem naboru podatkov opazili izboljšanje točnosti tehnik, ki upoštevajo skupine. Naša primerjava tudi sicer uporablja največje število podatkovnih virov.

Tabela 2.1

Rezultati AUC različnih transformacijskih metod, kjer smo kot končni klasifikator uporabili logistično regresijo.

nabor podatkov	primerov	značilk	skupin	netransformirani	SpLin	SetSig	CORG	PLS	SPCA	PCA	arit. sred.	GSA	mediana
DLBCL	77	6219	1904	.982	.989	.999	.987	.986	.994	.984	.974	.980	.980
GDS232	23	932	811	.629	.491	.515	.461	.548	.608	.596	.697	.680	.706
GDS531	173	9459	1977	.767	.756	.762	.750	.779	.716	.753	.763	.777	.787
GDS806	60	20007	1985	.701	.663	.704	.677	.686	.684	.699	.634	.680	.663
GDS971	23	9695	1980	.871	.967	.988	.938	.975	.892	.975	.950	.954	.846
GDS1059	53	6200	1867	.690	.713	.699	.711	.692	.649	.673	.680	.722	.652
GDS1062	27	14903	1984	.715	.691	.774	.767	.679	.692	.667	.658	.667	.706
GDS1209	54	14903	1984	.989	.981	.972	.971	.968	.968	.972	.998	.981	1.000
GDS1210	30	6277	1917	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1220	54	14903	1984	.894	.894	.911	.896	.894	.892	.897	.893	.896	.890
GDS1221	28	9697	1980	.529	.610	.584	.573	.610	.553	.429	.511	.557	.481
GDS1282	32	14903	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1329	43	14902	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1375	63	14903	1984	.983	.997	.990	.990	.995	.997	.973	.997	.980	.992
GDS1390	20	14903	1984	.855	.855	.842	.842	.803	.829	.816	.842	.803	.829
GDS1562	28	1394	454	.994	.989	.972	.989	.989	.989	.989	.983	.967	.983
GDS1618	20	14903	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1650	39	9697	1980	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1667	36	34700	1968	.977	1.000	1.000	1.000	1.000	1.000	1.000	.960	.991	.976
GDS1714	28	16246	1990	.814	.764	.781	.811	.736	.836	.725	.708	.719	.633
GDS1887	46	9697	1980	.455	.407	.376	.410	.412	.461	.407	.446	.451	.370
GDS2113	75	14903	1984	.645	.594	.571	.598	.587	.535	.613	.613	.596	.582
GDS2201	37	14903	1984	1.000	1.000	1.000	.996	1.000	1.000	1.000	.978	.992	.965
GDS2250	38	34700	1968	.990	1.000	1.000	.992	.993	1.000	.985	.972	.990	.957
GDS2415	59	14345	1964	.750	.749	.742	.729	.737	.725	.759	.737	.759	.707
GDS2489	44	6278	1917	.980	.988	.972	.997	.988	.990	.985	.993	.988	.995
GDS2520	44	9697	1980	.986	.983	.989	.976	.978	.981	.988	.963	.954	.968
GDS2547	122	10629	1163	.777	.763	.765	.775	.761	.763	.757	.710	.745	.705
GDS2609	22	34700	1968	.908	.950	.938	.963	.917	.950	.938	.900	.883	.883
GDS2735	46	18022	1974	.678	.671	.612	.715	.649	.731	.720	.749	.636	.627
GDS2771	187	14902	1984	.789	.781	.803	.777	.777	.752	.759	.747	.757	.762
GDS2785	43	9697	1980	.822	.838	.860	.860	.829	.810	.811	.787	.808	.835
GDS2842	36	9697	1980	.642	.573	.602	.555	.537	.527	.518	.554	.530	.594
GDS3268	202	29391	1971	.925	.922	.890	.895	.895	.888	.880	.876	.890	.782
GDS3929	183	19136	1968	.479	.554	.455	.501	.502	.468	.578	.404	.406	.391
GDS3952	88	31623	1966	.477	.471	.471	.509	.489	.461	.463	.457	.446	.396
GDS4228	166	3779	1457	.841	.744	.761	.732	.750	.749	.729	.714	.739	.695
GDS4228_agent	125	3779	1457	.891	.879	.866	.833	.842	.843	.846	.836	.838	.834
GSE412	110	6776	1888	.955	.975	.978	.926	.937	.901	.928	.934	.918	.902
GSE3726	52	14166	1986	.933	.930	.939	.974	.978	.960	.937	.949	.934	.946
leukemia	72	4680	1819	.996	.999	.996	.992	.989	.992	.991	.998	.994	.998
prostata	102	9582	1976	.964	.966	.954	.964	.933	.926	.946	.931	.912	.923
povprečje	66	13686	1857	.840	.836	.834	.834	.829	.826	.826	.821	.822	.808

Tabela 2.2

Rezultati AUC različnih transformacijskih metod, kjer smo kot končni klasifikator uporabili naključne gozdove.

nabor podatkov	primerov	značilk	skupin	netransformirani	SpLin	SetSig	CORG	PLS	SPCA	PCA	arit. sred.	GSA	mediana
DLBCL	77	6219	1904	.961	.995	.993	.952	.974	.976	.936	.927	.953	.914
GDS232	23	932	811	.526	.471	.519	.500	.605	.556	.579	.697	.711	.658
GDS531	173	9459	1977	.745	.763	.761	.741	.767	.722	.691	.726	.762	.702
GDS806	60	20007	1985	.770	.662	.712	.731	.705	.738	.728	.711	.713	.733
GDS971	23	9695	1980	.988	.988	.988	.988	.925	.942	.854	.875	.900	.867
GDS1059	53	6200	1867	.667	.687	.702	.704	.685	.657	.690	.723	.734	.683
GDS1062	27	14903	1984	.783	.756	.792	.763	.787	.708	.733	.812	.723	.742
GDS1209	54	14903	1984	.996	.981	.978	.998	.987	.992	.992	.998	1.000	1.000
GDS1210	30	6277	1917	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1220	54	14903	1984	.928	.925	.904	.899	.886	.901	.890	.901	.914	.884
GDS1221	28	9697	1980	.485	.552	.498	.526	.482	.547	.449	.427	.456	.480
GDS1282	32	14903	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1329	43	14902	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1375	63	14903	1984	.997	.997	.995	.997	.997	.997	.997	.997	1.000	.997
GDS1390	20	14903	1984	.868	.868	.842	.829	.829	.776	.750	.882	.789	.824
GDS1562	28	1394	454	.975	.989	.972	.975	.967	.989	.950	.933	.942	.953
GDS1618	20	14903	1984	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1650	39	9697	1980	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1667	36	34700	1968	.996	1.000	.996	1.000	.987	1.000	.956	.953	.953	.937
GDS1714	28	16246	1990	.822	.778	.749	.800	.756	.792	.703	.614	.739	.539
GDS1887	46	9697	1980	.415	.427	.347	.346	.382	.477	.538	.534	.480	.485
GDS2113	75	14903	1984	.590	.598	.563	.578	.575	.495	.562	.561	.582	.632
GDS2201	37	14903	1984	1.000	1.000	.983	.996	.975	.962	.895	.887	.971	.879
GDS2250	38	34700	1968	1.000	1.000	1.000	1.000	1.000	1.000	1.000	.895	.963	.885
GDS2415	59	14345	1964	.712	.752	.733	.737	.730	.724	.672	.707	.749	.664
GDS2489	44	6278	1917	1.000	1.000	.977	1.000	.993	1.000	.969	.975	.995	.977
GDS2520	44	9697	1980	.982	.991	.973	.968	.971	.975	.948	.926	.941	.944
GDS2547	122	10629	1163	.773	.752	.772	.766	.764	.734	.739	.707	.748	.695
GDS2609	22	34700	1968	.979	.942	.938	.950	.904	.950	.925	.896	.904	.971
GDS2735	46	18022	1974	.593	.643	.444	.560	.444	.734	.405	.502	.410	.410
GDS2771	187	14902	1984	.740	.793	.767	.742	.745	.753	.716	.684	.705	.698
GDS2785	43	9697	1980	.850	.808	.851	.857	.805	.833	.761	.711	.799	.748
GDS2842	36	9697	1980	.582	.597	.623	.581	.562	.555	.510	.536	.554	.567
GDS3268	202	29391	1971	.836	.904	.800	.810	.798	.784	.767	.752	.781	.727
GDS3929	183	19136	1968	.480	.505	.409	.440	.430	.487	.500	.434	.429	.468
GDS3952	88	31623	1966	.485	.480	.491	.526	.525	.535	.433	.511	.489	.457
GDS4228	166	3779	1457	.726	.761	.753	.734	.714	.735	.712	.674	.722	.721
GDS4228_agent	125	3779	1457	.844	.867	.819	.825	.814	.819	.823	.828	.825	.815
GSE412	110	6776	1888	.915	.980	.944	.894	.870	.875	.875	.871	.863	.855
GSE3726	52	14166	1986	.971	.984	.949	.974	.923	.940	.877	.890	.896	.887
leukemia	72	4680	1819	.998	.998	.987	.998	.991	.998	.978	.944	.989	.950
prostata	102	9582	1976	.939	.955	.903	.932	.884	.895	.825	.881	.887	.866
povprečje	66	13686	1857	.831	.837	.820	.824	.813	.823	.793	.797	.809	.791

Razlike v stabilnosti med metodami po meri Kuncheve [82] so veliko večje (slika 2.3). Glede na naše rezultate se stabilnost pri preprostejših metodah izboljša, pri kompleksnejših, ki iz posameznih skupin izbirajo podmnožice značilk ali uporabljajo razredno spremenljivko, pa poslabša. Na preprostejše mere osredotočena študija poroča o izboljšanju stabilnosti [16], na kompleksnejše mere osredotočena pa o poslabšanju stabilnosti [18].

Možnih razlag za razmeroma slabo napovedno točnost po transformaciji v prostor značilk, ki opisujejo skupine, je več [15, 18]. Razdelimo jih lahko na pomanjkljivosti virov predznanja, potencialno slabo konstrukcijo novih značilk, in neustrezno napovedovanje s konstruiranimi značilkami. Z viri predznanja se v sklopu disertacije nismo ukvarjali, poskušali pa smo izboljšati konstrukcijo značilk skupin (poglavje 4) in napovedovanje s konstruiranimi značilkami (poglavje 5).

2.4 *Interpretabilnost napovednih modelov s skupinami značilk*

Kljub temu, da transformacije v prostor genskih skupin s preizkušenimi metodami v povprečju niso izboljšale niti točnosti niti stabilnosti napovednih modelov, lahko iz zgrajenih modelov razberemo biološke procese, ki so pomembni za ločevanje med razredi. Modeli na transformiranih podatkih, v našem primeru na genskih skupinah, z vidika interpretacije dopolnjujejo modele na netransformiranih podatkih. Tu bi poudarili, da je v domeni molekularne biologije interpretabilnost rezultatov analiz ključna in je eden razlogov za obstoj in vzdrževanje virov podatkov o genskih skupinah [13]. Raziskave s področja molekularne biologije redno uporabljajo genske skupine za interpretacijo rezultatov [84–86], nekatere pa tudi za načrtovanje novih poskusov [87].

Kot primer interpretacije napovednih modelov na transformiranih podatkih predstavljamo podatkovni nabor *leukemia* [3], ki vsebuje genske ekspresije za dve vrsti levkemije: akutna limfoblastna levkemija (ALL) in akutna mieloblastna levkemija (AML). Uporabili smo transformacijsko metodo SetSig, ker je edina metoda med najboljšimi tremi, ki ne izvaja izbora genov in je zato najlažja za interpretacijo, saj ocene skupine opisujejo celo skupino in ne zgolj njene podmnožice. Za napovedovanje smo uporabili logistično regresijo. Najboljših 10 značilk smo izbrali z rekurzivnim odstranjevanjem značilk [59]. V vsaki iteraciji smo odstranili 1% značilk. Končni modeli z 10 značilkami so imeli AUC 0.981, če so bile značilke geni, in 0.997 za značilke transformirane z metodo SetSig. Značilke smo izbirali le na učnih podatkih.

V tabeli 2.3, ki prikazuje 10 genov oziroma skupin genov v končnem modelu, vi-

Tabela 2.3

Modeli logistične regresije z 10 najboljšimi značilkami na naboru podatkov *leukemia* na genih (levo) in genskih skupinah (desno). Če je vrednost izraza pomnožena z utežjo pozitivna (negativna), se verjetnost za razred ALL poveča (zmanjša).

gen	*utež*
NME4	−0.969
NEK3	−0.748
CD33	−0.747
STMN1	0.745
CST3	−0.716
STOM	−0.681
ZYX	−0.625
MPO	−0.600
PRG1	−0.555
DF	−0.544

genska skupina	*utež*
KEGG hematopoietic cell lineage	0.676
anatomical structure formation	0.613
Reactome muscle contraction	0.603
KEGG lysosome	0.590
KEGG vibrio cholerae infection	0.583
Reactome formation of platelet plug	0.573
Reactome platelet activation	0.567
Reactome gap junction trafficking	0.561
BioCarta ucalpain	0.557
electron transport GO:0006118	0.554

dimo, da imajo vzorci ALL nižje vrednosti izrazov gena zyxin (ZYX) in višje vrednosti skupine, ki predstavlja signalno pot μ-kalpaina (BioCarta ucalpain). Kalpaini so pri raku pomembni predvsem pri množitvi celic in v zaviranju naravne celične smrti; pri limfoblastni levkemiji, kjer se celice množijo hitreje, naj bi bili kalpaini višje izraženi [54]. Aktivacija signalne poti μ-kalpaina poveča razgradnjo proteinskih kompleksov v fokalnih adhezijah, kjer je gen zikcin (zyxin) najbolj prisoten [88]. Višje vrednosti skupine signalna pot μ-kalpaina v vzorcih ALL lahko razložijo nižjo izraženost gena zikcin v istih vzorcih [54].

Utežene genske skupine nam dobro podajo splošno sliko domene in predstavijo rezultate z bolj sistemskega vidika, bolj specifično razlago pa predstavlja seznam genov, ki je tipično razumljiv le specialistom. Primer kaže tudi na odlično dopolnjevanje seznamov genov in genskih skupin.

3

Podobnost izraznih profilov genov v genskih skupinah

3.1 Uvod

Objekti, ki jih v bioinformatiki najpogosteje analiziramo, so geni, beljakovine, kemikalije in presnovni produkti. Ker jih že deloma poznamo, lahko za analizo podatkov uporabljamo na znanju osnovane metode [89]. Trenutne metodologije se pogosto opirajo na domensko znanje o skupinah proučevanih objektov, ki jih najdemo v raznih bazah. Ena najbolj znanih baz je genska ontologija (angl. gene ontology, GO) [9], ki posameznim genom, katerih funkcijo poznamo, pripiše enega ali več razredov (angl. term) iz vnaprej definirane ontologije. Skupina zanimivih objektov je lahko skupina genov, ki jih GO uvršča v isti razred ontologije. Sorodna podatkovna baza, Kyoto Encyclopedia of Genes and Genomes (KEGG) [8], vsebuje biološke poti s pripadajočimi proteini, vmesnimi produkti in geni; geni v neki metabolični poti lahko pripadajo določeni skupini. Baza BioGRID [90] je organizirana drugače: vsebuje pare proteinov ali genov, ki so v interakcijah. Na gena, ki sta v interakciji, ali pa sta v interakcijah proteina, ki jih gena kodirata, lahko gledamo kot na (mini) skupino.

Pri analizi podatkov o genskih izrazih imamo za vsak vzorec podane vrednosti več tisoč značilk, kjer vsaka značilka opisuje stopnjo izraženosti določenega gena. Na takšnih podatkih lahko z znanjem o skupinah genov ugotavljamo obogatenost posameznih skupin [12]. Če poznamo še razredno spremenljivko, lahko uporabimo metode nadzorovanega strojnega učenja. S pomočjo poznanih genskih skupin lahko pred gradnjo napovednih modelov tvorimo nove značilke, ki ne opisujejo več posameznih genov, ampak večje enote – skupine [15, 33].

Metode, ki obravnavajo skupine, temeljijo na predpostavki, da so izrazni profili genov, ki pripadajo isti skupini, povezani. Da bi ocenili, če predpostavka drži, smo izmerili podobnosti med izraznimi profili genov (stopnje izraženosti za nek gen čez več vzorcev) iz genskih skupin iz vira genskih poti KEGG in vira interkcij BioGrid.

Predhodne raziskave so potrdile, da so izrazni profili genov, ki kodirajo beljakovine v interakcijah, med seboj bolj podobni kot izrazni profili naključnih genov, če podobnost merimo s Pearsonovim koeficientom korelacije [20–22]. Tisti geni kvasovk *Saccharomyces cerevisiae*, ki kodirajo proteine velikih proteinskih kompleksov (angl. permanent complex), kot sta ribosom ali proteasom, imajo še posebej podobne izrazne profile [22]. Študije na manjšem naboru podatkov so te rezultate potrdile, ob čemer so se osredotočile na koevolucijo genskih izrazov [21] ali primerjavo med več vrstami [20]. Nasprotno so Jelizarow et al. [91] poročali, da med podobnostmi genov v KEGG-ovih

poteh in podobnostmi med naključnimi geni ni razlik. V veliki študiji na 60 naborih podatkov so pregledovali vzorce koreliranih izraznih profilov genov med nabori podatkov in primerjali združene rezultate z razredi iz genske ontologije, ampak niso ocenili posameznih naborov podatkov [92].

V tem poglavju predstavljamo analizo podobnosti med izraznimi profili genov v genskih skupinah ali interakcijah na velikem naboru podatkov. Podobnosti med izraznimi profili genov smo merili s Pearsonovim koeficientom korelacije in interakcijskim prispevkom. Interakcijski prispevek [19] je nadzorovana mera podobnosti, ki temelji na informacijski teoriji. Naš prispevek pričujočega poglavja je, da smo enak test izvedli na večjem naboru podatkov kot predhodne študije, uporabili pa smo tudi mero podobnosti, ki upošteva vrednost razredne spremenljivke. Dele tega poglavja smo predhodno objavili [93, 94].

3.2 Podatki in metode

V tem razdelku opišemo metode, ki smo jih uporabili za analizo podobnosti genskih profilov, eksperimentalno metodologijo in uporabljene podatke.

3.2.1 Podatki o genskih izrazih

Podatki o genskih izrazih opisujejo koncentracijo informacijske RNA (angl. messenger RNA) za tisoče genov za vsak vzorec. Uporabili smo 42 naborov podatkov o človekovih genskih izrazih, ki smo jih opisali v razdelku 2.3.1.

3.2.2 Mere podobnosti izrazov para genov

Podobnosti izraznih profilov genov smo merili s Pearsonovim koeficientom korelacije, ki vrednosti ciljne spremenljivke ne upošteva, ter interakcijskim prispevkom, ki vrednost razredne spremenljivke upošteva. Interakcije med izraznimi profili bi lahko merili tudi drugače, denimo z mero HFCC [95], vendar se je v naših predhodnih poskusih interakcijski prispevek obnesel bolje kot mera HFCC [93].

Pearsonov koeficient korelacije

Pearsonov koeficient korelacije [96] ocenjuje stopnjo linearne povezanosti med dvema spremenljivkama. Naj bosta X in Y dve spremenljivki, ki predstavljata izrazna profila

dveh genov (vektorja dolžine N). Pearsonov koeficient korelacije med spremenljivkama X in Y je definiran kot

$$\rho_{XY} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}},$$

kjer sta $\bar{x}$ in $\bar{y}$ povprečni vrednosti spremenljivk X in Y. Pearsonov koeficient korelacije zavzame vrednosti na intervalu $[-1, 1]$.

Kot mero podobnosti so ga uporabili tudi v sorodnih raziskavah [20–22, 92].

Interakcijski prispevek

Interakcijski prispevek (angl. interaction gain, bivariate synergy) ocenjuje količino informacije o razredu, ki jo pridobimo, če izrazna profila genov upoštevamo hkrati, glede na informacijo, ki jo pridobimo, če izrazna profila genov obravnavamo ločeno [19, 97]. Interakcijski prispevek izraznih profilov X and Y glede na razredno spremenljivko C je

$$\mathrm{IntGain}_C(X, Y) = \mathrm{Gain}_C(X \times Y) - \mathrm{Gain}_C(X) - \mathrm{Gain}_C(Y),$$

kjer $\mathrm{Gain}_C(X)$ označuje informacijski prispevek izraznega profila X glede na razredno spremenljivko C in kjer $X \times Y$ označuje kartezični produkt izraznih profilov. Informacijski prispevek je

$$\mathrm{Gain}_C(X) = -\sum_{c \in D_C} p(c)\log_2 p(c) + \sum_{v \in D_X} p(v) \sum_{c \in D_C} p(c|v)\log_2 p(c|v),$$

kjer sta D_C in D_X množici možnih vrednosti razredne spremenljivke in izraznega profila. Podobna izrazna profila bosta imela negativen interakcijski prispevek, ker oba nosita podobno informacijo o razredu.

Da bi lahko interakcijski prispevek uporabili za merjenje podobnosti izraznih profilov genov, ki so ponavadi podani kot zvezne spremenljivke, smo izrazne profile predhodno diskretizirali v tri intervale tako, da je vsak interval diskretizirane značilke vseboval isto število primerov. Za zgolj tri intervale smo se odločili, ker imamo v tem primeru pri izračunu informacijskega prispevka kartezičnega produkta že $3^2 = 9$ različnih vrednosti, kar je veliko glede na število učnih primerov (najmanj 20, v povprečju 66). Da ne pokvarimo izračuna interakcijskega prispevka, smo uporabili diskretizacijo, ki ne uporablja razredne spremenljivke, ker tako ohranimo razmerja med interakcijskimi prispevki posameznih značilk in kartezičnega produkta.

3.2.3 Skupine genov

Za vire genskih skupin smo uporabili biološke poti iz baze KEGG [8], katerih najnovejšo verzijo smo pridobili 5.2.2016. Za primerjavo smo uporabili še verzije iz 21.8.2012, 5.7.2013, 3.4.2014 in 14.1.2015.

Mini skupine (z dvema genoma) smo dobili iz baze interakcij med geni in proteini BioGRID [90]. Uporabili smo verzije 3.4.133 (25.1.2016), 3.2.109 (25.1.2014), 3.1.85 (25.1.2012), 2.0.61 (25.1.2010) in 2.0.37 (27.1.2008).

3.2.4 Opis poskusa

Za vsak nabor podatkov smo izmerili stopnjo podobnosti med pari izraznih profilov genov. Merili smo podobnosti med vsemi geni, kjer sta bila gena

1. iz iste genske skupine: v isti biološki poti (KEGG) ali označena kot v interakciji (BioGRID) ali

2. izbrana naključno.

Prvi seznam podobnosti smo tvorili s podobnostmi med vsemi pari genov iz genskih skupin, ki smo jih našli na konkretnem naboru podatkov. Za drugi seznam smo izbrali podobnosti med enakim številom naključno izbranih parov genov. Dobljena seznama smo, kot v [20, 21], primerjali s Kolmogorov-Smirnovim testom.

Kolmogorov-Smirnov test dveh vzorcev je neparametrični statistični test, ki testira hipotezo, da dva vzorca vrednosti pripadata isti porazdelitvi. Test izmeri največjo razdaljo med porazdelitvenima funkcijama obeh vzorcev in, glede na velikost vzorca, izračuna p-vrednost [96].

3.3 Rezultati in razprava

Tabela 3.1 prikazuje vrednost statistike Kolmogorov-Smirnov in pripadajočih p vrednosti za vse nabore podatkov. Če podobnost merimo s Pearsonovim koeficientom korelacije, so p-vrednosti manjše od 0.00001 ($-log_{10}p > 5$) pri vseh 42 naborih podatkov za skupine genov iz baze KEGG in pri 41 naborih podatkov za skupine genov iz baze BioGRID. Če podobnost merimo z interakcijskim prispevkom, dobimo za KEGG 28 in za BioGRID 26 naborov podatkov s p-vrednostjo manjšo od 0.00001.

Porazdelitve podobnosti štirih naborov podatkov prikazujeta sliki 3.1 in 3.2. Na sliki 3.1 vidimo, da so Pearsonovi koeficienti korelacije izraznih profilov genov iz baze

Tabela 3.1

Vrednosti statistike Kolmogorov-Smirnov (KS) in pripadajoče p vrednosti z najnovejšimi verzijami baz skupin genov za vse nabore podatkov. Logaritmi so desetiški.

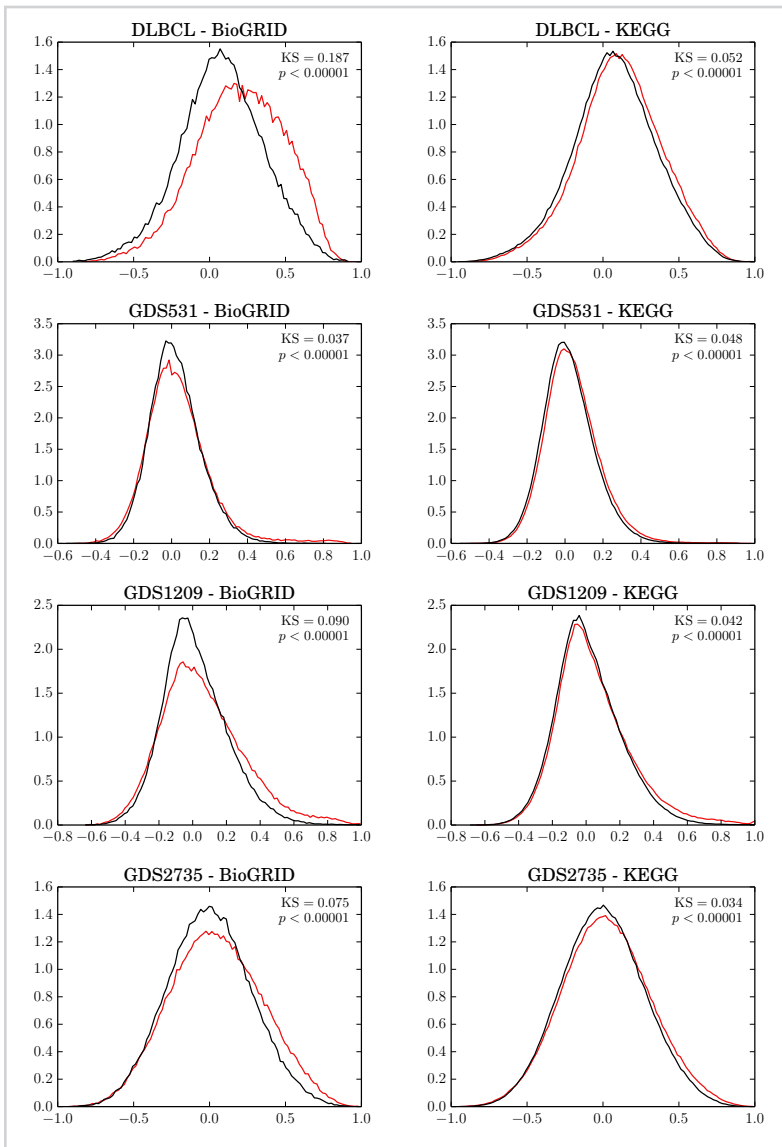
	Pearsonov koeficient korelacije				interakcijski prispevek			
	BioGRID		KEGG		BioGRID		KEGG	
	KS	$-\log p$	KS	$-\log p$	KS	$-\log p$	KS	$-\log p$
DLBCL	0.187	> 100	0.052	> 100	0.076	> 100	0.012	52.5
GDS232	0.026	3.4	0.026	21.0	0.016	0.8	0.010	2.4
GDS531	0.037	> 100	0.048	> 100	0.004	1.2	0.004	8.4
GDS806	0.087	> 100	0.049	> 100	0.005	3.1	0.001	0.1
GDS971	0.058	> 100	0.021	> 100	0.054	> 100	0.008	33.7
GDS1059	0.049	> 100	0.027	> 100	0.008	3.3	0.004	2.8
GDS1062	0.036	> 100	0.032	> 100	0.014	23.0	0.008	40.6
GDS1209	0.090	> 100	0.042	> 100	0.055	> 100	0.007	31.6
GDS1210	0.056	> 100	0.037	> 100	0.024	28.8	0.004	4.3
GDS1220	0.015	28.7	0.035	> 100	0.016	30.7	0.026	> 100
GDS1221	0.049	> 100	0.044	> 100	0.011	9.6	0.009	41.3
GDS1282	0.060	> 100	0.030	> 100	0.060	> 100	0.019	> 100
GDS1329	0.032	> 100	0.043	> 100	0.019	43.9	0.004	11.2
GDS1375	0.058	> 100	0.033	> 100	0.083	> 100	0.031	> 100
GDS1390	0.027	89.1	0.020	> 100	0.023	66.8	0.004	11.4
GDS1562	0.114	> 100	0.059	62.8	0.047	19.9	0.029	14.4
GDS1618	0.133	> 100	0.045	> 100	0.140	> 100	0.069	> 100
GDS1650	0.100	> 100	0.037	> 100	0.057	> 100	0.022	> 100
GDS1667	0.040	> 100	0.033	> 100	0.034	> 100	0.025	> 100
GDS1714	0.089	> 100	0.038	> 100	0.015	25.6	0.005	11.2
GDS1887	0.034	98.8	0.031	> 100	0.009	5.7	0.006	19.2
GDS2113	0.014	23.1	0.038	> 100	0.010	10.9	0.003	4.4
GDS2201	0.024	69.3	0.026	> 100	0.023	66.8	0.007	28.1
GDS2250	0.074	> 100	0.038	> 100	0.006	5.4	0.006	32.3
GDS2415	0.087	> 100	0.064	> 100	0.005	1.4	0.003	3.6
GDS2489	0.043	91.8	0.039	> 100	0.012	6.8	0.003	1.7
GDS2520	0.038	> 100	0.032	> 100	0.014	16.6	0.006	18.1
GDS2547	0.064	70.2	0.060	> 100	0.017	4.3	0.020	17.5
GDS2609	0.078	> 100	0.062	> 100	0.085	> 100	0.109	> 100
GDS2735	0.075	> 100	0.034	> 100	0.004	1.6	0.003	4.7
GDS2771	0.054	> 100	0.054	> 100	0.014	25.1	0.012	96.5
GDS2785	0.023	45.5	0.020	> 100	0.049	> 100	0.020	> 100
GDS2842	0.025	54.0	0.034	> 100	0.004	1.0	0.006	15.3
GDS3268	0.101	> 100	0.058	> 100	0.017	45.7	0.018	> 100
GDS3929	0.131	> 100	0.060	> 100	0.007	7.7	0.011	> 100
GDS3952	0.149	> 100	0.066	> 100	0.006	5.8	0.003	5.5
GDS4228	0.138	> 100	0.051	> 100	0.023	6.8	0.005	1.7
GDS4228_agent	0.130	> 100	0.046	> 100	0.027	9.2	0.011	11.9
GSE412	0.270	> 100	0.041	> 100	0.040	> 100	0.021	> 100
GSE3726	0.213	> 100	0.042	> 100	0.034	> 100	0.014	> 100
leukemia	0.161	> 100	0.040	> 100	0.029	36.7	0.010	21.3
prostata	0.049	> 100	0.012	77.9	0.023	47.5	0.009	41.6

BioGRID večji kot tisti na naključnih parih, kar so prikazale že predhodne raziskave podobnosti izraznih profilov parov v proteinskih interakcijah [20–22]. Kot so poročali Jansen et al. [22], so razlike med porazdelitvami majhne, a so zaradi velikega števila vzorcev (parov genov) statistično značilne (p-vrednosti so zelo majhne). V nasprotju z Jelizarow et al. [91], ki niso potrdili razlike med geni iz KEGG in med naključnimi geni, naši rezultati potrjujejo razliko med naključnimi porazdelitvami in porazdelitvami iz skupin genov. Tudi negativnih korelacij je na genih iz skupin včasih več kot na naključno izbranih genih (na primer pri GDS531 in GDS1209 za gene iz BioGRID na sliki 3.2). Za pogostejše pozitivne korelacije med geni iz preizkušenih virov so Lee et al. [92] našli biološke razloge.

Porazdelitev interakcijskih prispevkov parov genov iz preizkušenih virov skupin je bila zamaknjena proti negativnim vrednostim interakcijskega prispevka. Negativen interakcijski prispevek pomeni, da taki pari izraznih profilov vsebujejo prekrivajočo oziroma medsebojno redundantno informacijo o razredu. V povprečju so bile p-vrednosti večje (manj značilne) kot s Pearsonovo korelacijo. Vzrok temu bi lahko bilo majhno število vzorcev v naborih podatkov (za natančne meritve interakcijskega prispevka potrebujemo več vzorcev) ali diskretizacija vrednosti pred računanjem interakcijskega prispevka.

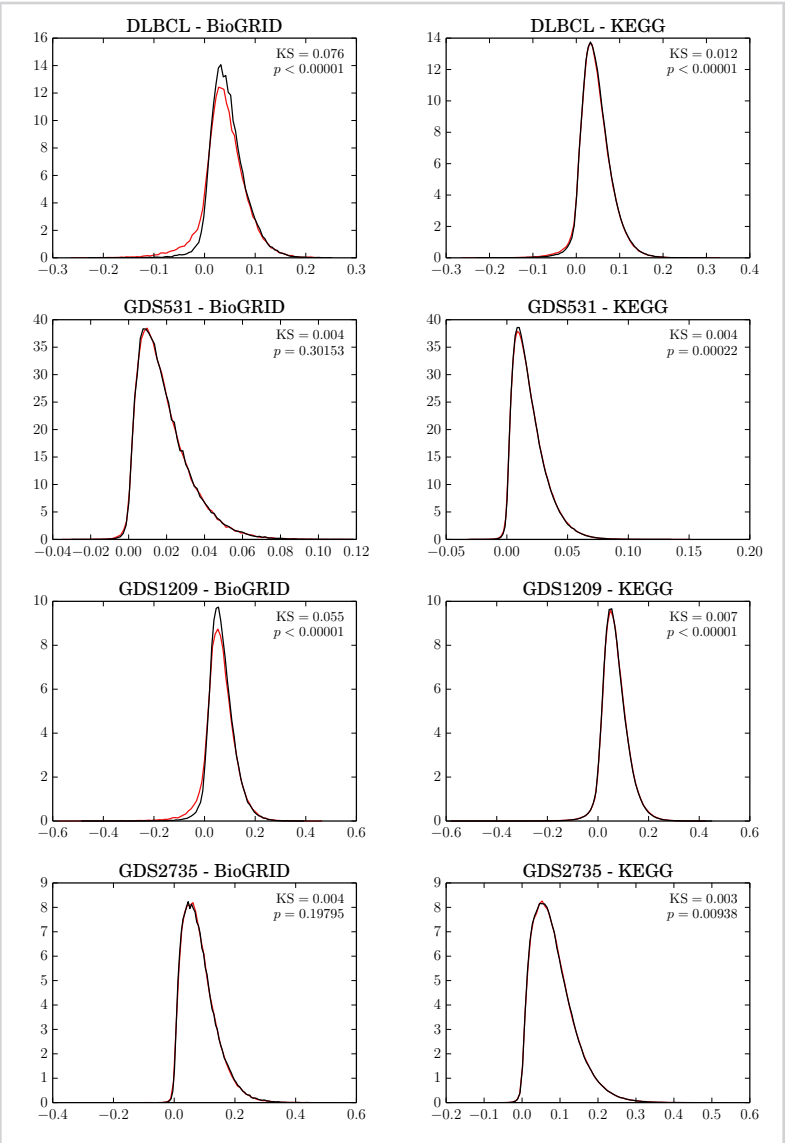
Kljub temu, da so negativni Pearsonovi koeficienti korelacije bolj podobni v preizkušenih skupinah kot med naključnimi pari genov, to za pozitivni interakcijski prispevek ne velja. Če bi bil pozitivni interakcijski prispevek pogostejši, bi to pomenilo, da potrebujemo drugačne na znanju osnovane metode podatkov: takšne, ki bi znale modelirati interakcije znotraj genskih skupin.

Tabela 3.2 prikazuje, kako se rezultati spremenijo, če uporabimo druge verzije baz genskih skupin. Vidimo lahko, da se je število parov genov iz baze BioGRID, ki smo jih našli v povprečnem naboru podatkov, povečalo za skoraj 10-krat med verzijama 2.0.37 (januar 2008) in 3.4.133 (januar 2016). Vrednosti statistike Kolmogorov-Smirnov se pri tem niso bistveno spremenile, kar nakazuje, da so novejše različice baz glede na podobnost genov znotraj skupin podobne starejšim različicam. Zaradi povečanja števila parov genov, ki jih dobimo iz posameznih baz, p-vrednosti postanejo bolj značilne. Baza KEGG, za katero imamo sicer podatke le za zadnja štiri leta, se je v tem času manj spreminjala kot baza BioGRID.



Slika 3.1

Porazdelitve Pearsonovega koeficienta korelacije na izbranih naborih podatkov. Rdeča barva prikazuje vrednosti na parih iz skupin genov, črna pa vrednosti na naključnih parih. Levi stolpec prikazuje rezultate na genskih skupinah iz baze BioGRID, desni iz baze KEGG.



Slika 3.2

Porazdelitve interakcijskega prispevka na izbranih naborih podatkov. Rdeča barva prikazuje vrednosti na parih iz skupin genov, črna pa vrednosti na naključnih parih. Levi stolpec prikazuje rezultate na genskih skupinah iz baze BioGRID, desni iz baze KEGG.

Tabela 3.2

Povzetek rezultatov za različne verzije baz skupin genov KEGG in BioGRID: prikazujemo povprečne vrednosti statistike Kolmogorov-Smirnov (KS), povprečne $log_{10}p$, kjer smo pri računanju povprečja za $p < 10^{-100}$ vzeli 10^{-100}, in pa povprečna števila parov genov iz skupin, ki smo jih našli na določenem naboru podatkov ($|N|$).

a) Pearsonov koeficient korelacije

BioGRID	2.0.37	2.0.61	3.1.85	3.2.109	3.4.133
povprečje KS	0.072	0.071	0.068	0.082	0.079
povprečje $log_{10}p$	33.1	34.9	43.2	66.2	75.4
povprečje $\|N\|$	12368	14381	24788	62098	98077

KEGG	21.8.2012	5.7.2013	3.4.2014	14.1.2015	5.2.2016
povprečje KS	0.035	0.034	0.034	0.041	0.040
povprečje $log_{10}p$	82.2	82.2	81.8	93.1	93.0
povprečje $\|N\|$	719052	769897	788936	514272	528714

b) interakcijski prispevek

BioGRID	2.0.37	2.0.61	3.1.85	3.2.109	3.4.133
povprečje KS	0.027	0.027	0.027	0.030	0.029
povprečje $log_{10}p$	7.5	8.8	13.3	25.9	31.5
povprečje $\|N\|$	12619	14680	25307	63367	100133

KEGG	21.8.2012	5.7.2013	3.4.2014	14.1.2015	5.2.2016
povprečje KS	0.016	0.016	0.016	0.014	0.014
povprečje $log_{10}p$	37.1	38.2	39.4	31.0	31.1
povprečje $\|N\|$	719120	769973	789015	514355	528799

3.4 Zaključek

Naši rezultati kažejo, da so izrazni profili genov iz skupin v bazah KEGG in BioGRID bolj podobni kot izrazni profili naključnih genov. To potrjuje rezultate predhodnih raziskav [20–22] in je v nasprotju z nekaterimi prej objavljenimi opažanji [91]. Naša prispevka sta veliko število uporabljenih podatkovnih naborov ter uporaba dodatne metrike podobnosti: interakcijskega prispevka. Z interakcijskim prispevkom smo pokazali, da pozitivne interakcije med geni iz skupin niso pogostejše kot med naključnimi geni. Pokazali smo še, da podobnosti genskih izrazov iz skupin preizkušenih baz ostajajo enake ne glede na verzijo, a jih z večanjem števila skupin v posamezni bazi statistično ocenimo kot bolj značilne.

Kljub temu, da smo lahko zanesljivo opazili razlike med porazdelitvami ocen podobnosti, so bile le-te velikostno gledano majhne. To bi lahko bil eden od razlogov, da se tehnike napovedovanja s predhodno pretvorbo značilk iz prostora genov v prostor skupin genov obnesejo slabše, kot smo pričakovali [15].

4

Sočasna matrična faktorizacija za napovedovanje s skupinami značilk

4.1 Uvod

Matrična faktorizacija razcepi matriko na produkt faktorjev z manjšimi rangi od izvirne matrike tako, da produkt čim bolje, glede na podano kriterijsko funkcijo, aproksimira izvorno matriko. V idealnem primeru, ko struktura izvorne matrike omogoča dobro rekonstrukcijo, s tem lahko odstranimo nekaj šuma. Z zmnožkom razcepnih faktorjev lahko rekonstruiramo elemente, ki jih v izvorni matriki nismo poznali.

Matrično faktorizacijo so v strojnem učenju začeli na široko uporabljati v priporočilnih sistemih. Eden njenih zgodnjih uspehov je bila zmaga na tekmovanju priporočilnih sistemov Netflix prize,[1] kjer so zmagovalci implementirali kompleksno mešanico tehnik, med katerimi je imela matrična faktorizacija velik pomen [23]. Uspešno so jo uporabili tudi na drugih področjih, denimo za iskanje skupnosti v omrežjih [24]. Nedavno so metode, ki temeljijo na matrični faktorizaciji, uspešno uporabili za združevanje več tipov bioloških podatkov [26]. V splošnem si od matrične faktorizacije pri združevanju bioloških podatkov veliko obetamo [25].

Večina trenutnih pristopov za napovedovanje s skupinami značilk za vsako skupino na nek način izbira podmnožico značilk skupine: nekateri izbirajo direktno, drugi pa implicitno, z utežmi posameznih značilk. Postopki, ki bi, nasprotno, genske skupine poskušali povečati, so redki, čeprav je njihova uporaba intuitivno smiselna. Genske skupine namreč odsevajo trenutno znanje na področju molekularne biologije, ki ga raziskovalci stalno nadgrajujejo. Pristop, ki so ga predlagali Van Vliet et al. (2007) [98], tvori genske skupine na podlagi podobnosti med geni v več različnih vzorcih, a je zaradi računske učinkovitosti omejen na iskanje podmnožic v že znanih genskih skupinah.

V tem poglavju matrično faktorizacijo uporabimo za transformacijo iz prostora značilk v prostor skupin značilk. Izvedemo sočasno faktorizacijo dveh matrik, kjer prva matrika vsebuje izvorne ali primarne podatke, denimo podatke o genskih izrazih različnih tkiv, druga pa podatke o skupinah značilk, denimo o genih, ki tvorijo proteine na isti presnovni poti. Pri sočasni matrični faktorizaciji si razcepa obeh izvornih matrik delita skupen latentni faktor. Zaradi skupnega deljenega faktorja izvorna matrika podatkov vpliva na razcep matrike skupin in s tem efektivno vpliva na sestavo skupin (in obratno).

Predlagamo metodo transformacije, ki za transformacijo podatkov v prostor skupin značilk prilagodi uspešen algoritem za zlivanje podatkov DFMF [27]. Predlagano me-

[1] http://www.netflixprize.com/

todo smo preizkusili na velikem številu naborov podatkov. Rezultati kažejo, da lahko zgradimo modele, ki dosegajo primerljivo točnost kot modeli zgrajeni na originalnih (netransformiranih) podatkih.

4.2 Metode

Naj podatkovna množica $\mathbf{X} \in \mathbb{R}^{m \times n}$ vsebuje m primerov $\mathbf{x} \in R^n$ v vrsticah, kjer je vsak primer opisan z vrednostmi n značilk: x_{ij} je vrednost j-te značilke i-tega primera. Brez izgube splošnosti lahko privzamemo, da so vrednosti razredne spremenljivke zapisane v vektorju $\mathbf{y} \in \{0,1\}^m$; y_i je vrednost razreda i-tega primera. Skupine značilk (o skupin) zapišemo v matriki $\mathbf{F} \in \{0,1\}^{n \times o}$, kjer element f_{ij} označuje ali i-ta značilka izvornih podatkov pripada j-ti skupini: vsaka skupina je stolpec matrike $\mathbf{F}$.

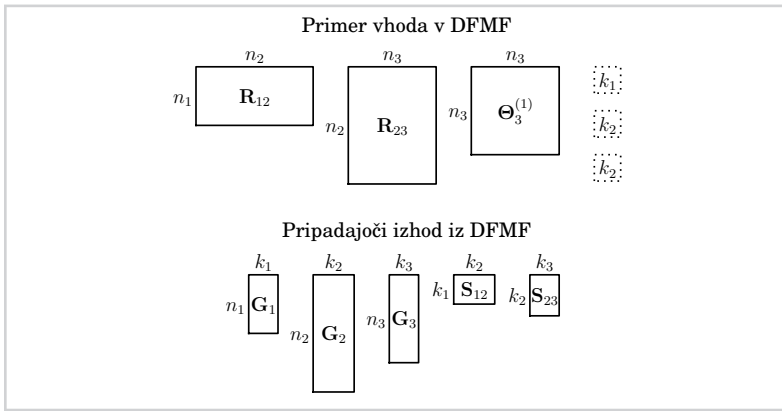
4.2.1 Transformacija v prostor skupin z množenjem matrik

Eden najpreprostejših načinov, da pretvorimo podatke $\mathbf{X}$ iz prostora originalnih značilk v prostor skupin značilk matrike $\mathbf{F}$, je seštevek vrednosti značilk iz skupine. To je ekvivalentno zmnožku matrik $\mathbf{XF}$ in sorodno metodi aritmetične sredine (razdelek 2.2.1).

4.2.2 Zlivanje podatkov z matrično faktorizacijo

Kot del transformacije v prostor skupin smo uporabili algoritem DFMF za sočasno matrično tri-faktorizacijo z omejitvami, ki sta ga razvila Žitnik in Zupan [27] in ga povzemamo v tem razdelku.

Matrična faktorizacija razcepi vhodno matriko $\mathbf{R}$ na faktorje manjših rangov od ranga izvorne matrike, katerih produkt glede na kriterijsko funkcijo dobro aproksimira izvorno matriko. Matriko $\mathbf{R}$ lahko razcepimo na produkt različnega števila faktorjev, denimo dveh $\mathbf{R} \approx \mathbf{FG}$ ali treh $\mathbf{R} \approx \mathbf{FSG}$ [99]. Algoritmi matrične faktorizacije ponavadi obravnavajo eno matriko naenkrat. Algoritem za zlivanje podatkov DFMF [27] pa hkrati, z eno kriterijsko funkcijo, faktorizira več matrik, ki lahko vsebujejo tako relacije med podatki različnih tipov objektov kot tudi relacije med podatki z istim tipom objektov. DFMF na vhodu prejme podatke relacij med različni tipi objektov $\mathbf{R}_{ij}$, ki opisujejo relacije med objekti tipov i in j, ter matrike omejitev $\mathbf{\Theta}_i$, ki opisujejo relacije med objekti tipa i. DFMF vhodne matrike $\mathbf{R}_{ij}$ sočasno razcepi v produkte treh faktorjev $\mathbf{G}_i\mathbf{S}_{ij}\mathbf{G}_j^T$, kjer je $\mathbf{S}_{ij}$ lasten matriki $\mathbf{R}_{ij}$, faktorja $\mathbf{G}_i$ in $\mathbf{G}_j$ pa si delijo vsi razcepi matrik relacij, ki opisujejo relacije med objekti tipa i oziroma tipa j in drugimi tipi objektov. DFMF na izhodu vrne take vrednosti matrik, ki minimizirajo seštevek



Slika 4.1

Primer vhoda in izhoda sočasne faktorizacije z DFMF. Na vhodu so tri matrike, dve matriki z relacijami med različnimi tipi objektov ($\mathbf{R}_{12}$ in $\mathbf{R}_{23}$) in matrika z relacijami znotraj tipa objektov ($\Theta_3^{(1)}$), ter želeni izhodni rangi k_1, k_2 in k_3. Števila objektov tipov 1, 2 in 3 so označena z n_1, n_2 in n_3.

razlik med podanimi matrikami relacij $\mathbf{R}_{ij}$ in njihovimi razcepi $\mathbf{G}_i\mathbf{S}_{ij}\mathbf{G}_j^T$, ob čemer poskušajo čim bolje zadostiti omejitvam v matrikah $\mathbf{\Theta}_i^{(t)}$. Primer uporabe prikazuje slika 4.1.

Algoritem DFMF [27] na vhodu prejme množico $\mathscr{R}$, ki vsebuje matrike relacij med r različnimi tipi objektov $\mathbf{R}_{ij}$, želene range latentnih matrik $k_1, k_2, \ldots, k_r$ ter matrike omejitev $\mathbf{\Theta}_i^{(t)}$ za $t \in 1, 2, \ldots, \max_i t_i$ (vsak tip objektov ima lahko več matrik z omejitvami). Na izhodu vrne matrike $\mathbf{G}_i$ in $\mathbf{S}_{ij}$ za $i, j \in [1, r]$. Naj bo $\mathbf{G}$ bločno diagonalna matrika, kjer si na diagonali sledijo $\mathbf{G}_1, \mathbf{G}_2, \ldots, \mathbf{G}_r$. $\mathbf{R}$ je bločna matrika, katere i,j-ti blok je enak $\mathbf{R}_{ij}$. DFMF najde lokalni minimum kriterijske funkcije

$$\sum_{\mathbf{R}_{ij} \in \mathscr{R}} \|\mathbf{R}_{ij} - \mathbf{G}_i\mathbf{S}_{ij}\mathbf{G}_j^T\|^2 + \sum_{t=1}^{\max_i t_i} \operatorname{tr}(\mathbf{G}^T\mathbf{\Theta}^t\mathbf{G})$$

z naslednjimi koraki:

(a) Nastavi začetno rešitev $\mathbf{G}_i$ za $i = 1, 2, \ldots, r$.

(b) Do konvergence ponavljaj:

- $\mathbf{S} \leftarrow (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{R}\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}$
- $\mathbf{G}_i^{(e)} \leftarrow 0$ in $\mathbf{G}_i^{(d)} \leftarrow 0$ za $i = 1, 2, \ldots, r$

- Za vsako $\mathbf{R}_{ij} \in \mathscr{R}$:

$$\begin{aligned}
\mathbf{G}_i^{(e)} &+= (\mathbf{R}_{ij}\mathbf{G}_j\mathbf{S}_{ij}^T)^+ + \mathbf{G}_i(\mathbf{S}_{ij}\mathbf{G}_j^T\mathbf{G}_j\mathbf{S}_{ij}^T)^- \\
\mathbf{G}_i^{(d)} &+= (\mathbf{R}_{ij}\mathbf{G}_j\mathbf{S}_{ij}^T)^- + \mathbf{G}_i(\mathbf{S}_{ij}\mathbf{G}_j^T\mathbf{G}_j\mathbf{S}_{ij}^T)^+ \\
\mathbf{G}_j^{(e)} &+= (\mathbf{R}_{ij}^T\mathbf{G}_i\mathbf{S}_{ij})^+ + \mathbf{G}_j(\mathbf{S}_{ij}^T\mathbf{G}_i^T\mathbf{G}_i\mathbf{S}_{ij})^- \\
\mathbf{G}_j^{(d)} &+= (\mathbf{R}_{ij}^T\mathbf{G}_i\mathbf{S}_{ij})^- + \mathbf{G}_j(\mathbf{S}_{ij}^T\mathbf{G}_i^T\mathbf{G}_i\mathbf{S}_{ij})^+
\end{aligned}$$

- Za $t = 1, 2, \ldots, \max_i t_i$ in $i = 1, 2, \ldots, r$:

$$\begin{aligned}
\mathbf{G}_i^{(e)} &+= [\mathbf{\Theta}_i^{(t)}]^-\mathbf{G}_i \\
\mathbf{G}_i^{(d)} &+= [\mathbf{\Theta}_i^{(t)}]^+\mathbf{G}_i
\end{aligned}$$

- Za $i = 1, 2, \ldots, r$:

$$\mathbf{G}_i \leftarrow \mathbf{G}_i \circ \sqrt{\frac{\mathbf{G}_i^{(e)}}{\mathbf{G}_i^{(d)}}}.$$

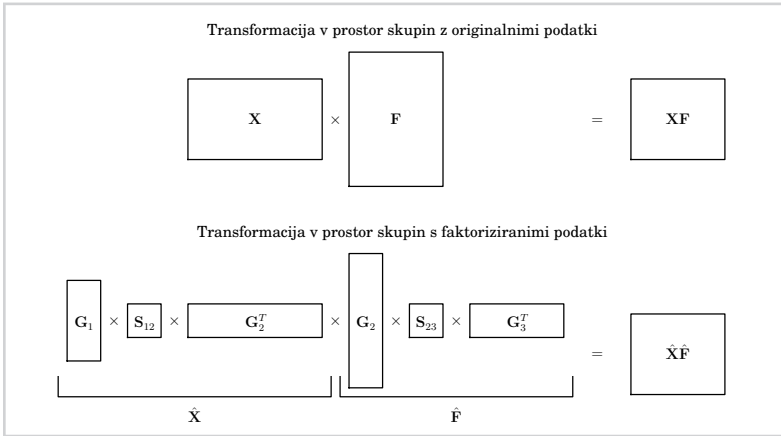
Algoritem DFMF so uspešno uporabili za napovedovanje genov, ki so potrebni, da ameba Dictyostelium še vedno uspeva na gojišču okuženem z Gram-negativnimi bakterijami. Združili so 14 različnih naborov podatkov. Vseh osem najbolj obetavnih kandidatov, ki so jih dobili z analizo razcepov DFMF, so s poskusi v laboratoriju kasneje potrdili [28]. Tudi pri napovedovanju toksičnosti zdravil, kjer sta z algoritmom DFMF zlila kar 29 različnih naborov podatkov, sta Žitnik in Zupan [100] dosegla zelo dobre rezultate.

4.2.3 Transformacija v prostor skupin s sočasno matrično faktorizacijo

Matrična faktorizacija aproksimira matriko s produktom matrik nižjega ranga. V predlaganem postopku uporabljamo tri-faktorizacijo, kjer vhodni matriki, ki predstavljata matriki relacij $\mathbf{X} = \mathbf{R}_{12}$ in $\mathbf{F} = \mathbf{R}_{23}$, aproksimiramo s produktom treh matrik. V splošnem lahko neko matriko $\mathbf{R}$ razcepimo na $\mathbf{R} \approx \mathbf{G}_1\mathbf{S}\mathbf{G}_2$, pri sočasni faktorizaciji dveh matrik pa si razcepa delita latentni faktor. V našem primeru je skupna latentna matrika značilk $\mathbf{G}_2$ (slika 4.2):

$$\mathbf{X} \approx \mathbf{G}_1\mathbf{S}_{12}\mathbf{G}_2^T \text{ in } \mathbf{F} \approx \mathbf{G}_2\mathbf{S}_{23}\mathbf{G}_3^T.$$

Za matrično faktorizacijo želimo uporabiti algoritem, ki nam omogoča sočasno faktorizacijo dveh matrik. Algoritma tri-SPMF [99], ki se nam na prvi pogled zdi ustrezen,



Slika 4.2

Transformacija originalnih podatkov v prostor skupin kot zmnožek originalnih matrik podatkov $\mathbf{X}$ in skupin $\mathbf{F}$ (zgoraj) ali z veriženjem razcepnih faktorjev (spodaj). Ista latentna matrika značilk ($\mathbf{G}_2$) je del rekonstrukcij $\hat{\mathbf{X}}$ in $\hat{\mathbf{F}}$.

ne moremo uporabiti, ker imamo podani le relaciji med prostorom primerov in prostorom značilk (matrika $\mathbf{R}_{12} = \mathbf{X}$) ter med prostorom značilk in prostorom skupin značilk (matrika $\mathbf{R}_{23} = \mathbf{F}$). Da bi zadostili omejitvam algoritma tri-SPMF, bi potrebovali še matriko z relacijami med primeri in skupin značilk (matriko $\mathbf{R}_{13}$), kar je pravzaprav matrika, ki jo želimo ustvariti. Zato smo za faktorizacijo uporabili algoritem za zlivanje podatkov DFMF, ki je posplošitev tri-SPMF [27].

Ker zlivamo le dve vhodni matriki ($\mathbf{R}_{12} = \mathbf{X}$ in $\mathbf{R}_{23} = \mathbf{F}$) brez omejitev, se kriterijska funkcija algoritma za zlivanje podatkov DFMF, katere minimum iščemo, poenostavi v

$$\|\mathbf{X} - \mathbf{G}_1\mathbf{S}_{12}\mathbf{G}_2^T\|^2 + \|\mathbf{F} - \mathbf{G}_2\mathbf{S}_{23}\mathbf{G}_3^T\|^2,$$

kjer v sklopu optimizacijske naloge iščemo matrike $\mathbf{G}_1, \mathbf{G}_2$ in $\mathbf{G}_3$ brez negativnih elementov ter poljubni matriki $\mathbf{S}_{12}$ in $\mathbf{S}_{23}$, ki dajo najmanjšo vrednost kriterijske funkcije. Za pretvorbo v prostor skupin najprej z algoritmom DFMF faktoriziramo matriko originalnih podatkov $\mathbf{X}$ in matriko skupin $\mathbf{F}$. Ker DFMF zagotavlja zgolj, da najde lokalni optimum, ne pa globalnega, ga poženemo večkrat. Postopek je sledeč:

1. Kot vhod algoritmu za zlivanje podatkov DFMF [27] nastavimo $\mathbf{R}_{12} = \mathbf{X}$ in $\mathbf{R}_{23} = \mathbf{F}$.

2. Algoritem DFMF (opisan v podrazdelku 4.2.2) poženemo p-krat, vsakič z drugimi začetnimi faktorji. Dobimo p različnih razcepov matrik $\mathbf{X}$ in $\mathbf{F}$.

3. Med p dobljenimi razcepi izberemo in shranimo tiste $\mathbf{S}_{12}$, $\mathbf{G}_2$, $\mathbf{S}_{23}$, $\mathbf{G}_3$, katerih faktorizacija je dala najmanjšo rekonstrukcijsko napako za $\mathbf{R}_{12} = \mathbf{X}$ ($\mathbf{G}_1$ ne shranimo). Rekonstrukcijske napake na $R_{23} = \mathbf{F}$ ne upoštevamo, ker je matrika $\mathbf{F}$ zelo redka.

Po transformaciji bi lahko originalne podatke transformirali v prostor skupin s $\hat{\mathbf{X}}\hat{\mathbf{F}}$, kjer bi bili $\hat{\mathbf{X}} = \mathbf{G}_1\mathbf{S}_{12}\mathbf{G}_2^T$ in $\hat{\mathbf{F}} = \mathbf{G}_2\mathbf{S}_{23}\mathbf{G}_3^T$ (slika 4.2), vendar želimo omogočiti tudi transformacijo testnih primerov, ki jih ob modeliranju morda še ne poznamo. Edini faktor, ki opiše različne vhodne primere je latentna matrika primerov $\mathbf{G}_1$: njene vrstice predstavljajo vrstice $\mathbf{X}$ v prvem latentnem prostoru. Zato moramo za transformacijo testnih primerov $\check{\mathbf{X}}$ izračunati njihov pripadajoč $\check{\mathbf{G}}_1$.

Latentni faktor testnih primerov $\check{\mathbf{G}}_1$ bi lahko izračunali z metodo najmanjših nenegativnih kvadratov (angl. non-negative least squares, NNLS) [101] kot v Žitnik in Zupan [27]. NNLS za podani matriki X in Z išče matriko D z nenegativnimi elementi, ki minimizira $\|X - ZD\|$. V našem primeru bi iskali D, ki minimizira $\|\check{\mathbf{X}}^T - \mathbf{G}_2\mathbf{S}_{12}^T D\|$ in nato dobili $\check{\mathbf{G}}_1 = D^T$. Ker NNLS optimizacijo izvaja drugače kot algoritem DFMF, se lahko $\check{\mathbf{G}}_1$, ki ga vrne NNLS, precej razlikuje od $\check{\mathbf{G}}_1$, ki ga vrne DFMF. Zato smo $\check{\mathbf{G}}_1$ računali z istimi multiplikativnimi pravili, kot jih uporablja algoritem DFMF, le da smo prej izračunani matriki $\mathbf{G}_2$ in $\mathbf{S}_{12}$ uporabili kot konstantni skozi celoten izračun. Naj $\check{\mathbf{X}}$ označuje originalne podatke, ki so lahko učni ali testni. Za pretvorbo $\check{\mathbf{X}}$ v prostor skupin predlagamo naslednji postopek:

1. Nastavimo $\mathbf{R}_{12} = \check{\mathbf{X}}$ in uporabimo $\mathbf{G}_2$ in $\mathbf{S}_{12}$, ki smo ju dobili kot rezultat zlivanja na učnih podatkih.

2. Uporabimo prilagojena multiplikativna pravila algoritma DFMF [27] za $\mathbf{G}_1$, kjer vrednosti $\mathbf{G}_2$ in $\mathbf{S}_{12}$ skozi celotno izvajanje ne spreminjamo:

   (a) Nastavi začetno rešitev $\check{\mathbf{G}}_1$.

   (b) Do konvergence ponavljaj:

   - Ker ostalih latentnih faktorjev ne spreminjamo, računamo le:

$$\check{\mathbf{G}}_1^{(e)} \mathrel{+}= (\check{\mathbf{X}}\mathbf{G}_2\mathbf{S}_{12}^T)^+ + \mathbf{G}_1(\mathbf{S}_{12}\mathbf{G}_2^T\mathbf{G}_2\mathbf{S}_{12}^T)^-$$
$$\check{\mathbf{G}}_1^{(d)} \mathrel{+}= (\check{\mathbf{X}}\mathbf{G}_2\mathbf{S}_{12}^T)^- + \mathbf{G}_1(\mathbf{S}_{12}\mathbf{G}_2^T\mathbf{G}_2\mathbf{S}_{12}^T)^+$$

- Posodobimo trenutni približek:

$$\check{\mathbf{G}}_1 \leftarrow \check{\mathbf{G}}_1 \circ \sqrt{\frac{\check{\mathbf{G}}_1^{(e)}}{\check{\mathbf{G}}_1^{(d)}}}.$$

3. V prostor skupin značilk transformiran $\check{\mathbf{X}}$ je $\check{\mathbf{T}} = \check{\mathbf{G}}_1\mathbf{S}_{12}\mathbf{G}_2^T\mathbf{G}_2\mathbf{S}_3\mathbf{G}_3^T$, kjer so $\mathbf{S}_{12}, \mathbf{G}_2, \mathbf{S}_{23}, \mathbf{G}_3$ rezultati predhodnega izračuna z algoritmom DFMF na učnih podatkih.

4.2.4 *Napovedovanje s transformiranimi podatki*

Predstavljena metoda transformira podatke iz prostora izvirnih značilk podatkovne množice $\mathbf{X}$ v prostor skupin podatkovne množice $\mathbf{F}$. Po transformaciji lahko na transformiranih podatkih uporabimo običajne metode za gradnjo napovednih modelov za klasifikacijo ali regresijo. Celoten postopek transformacije in napovedovanja prikazuje slika 4.3.

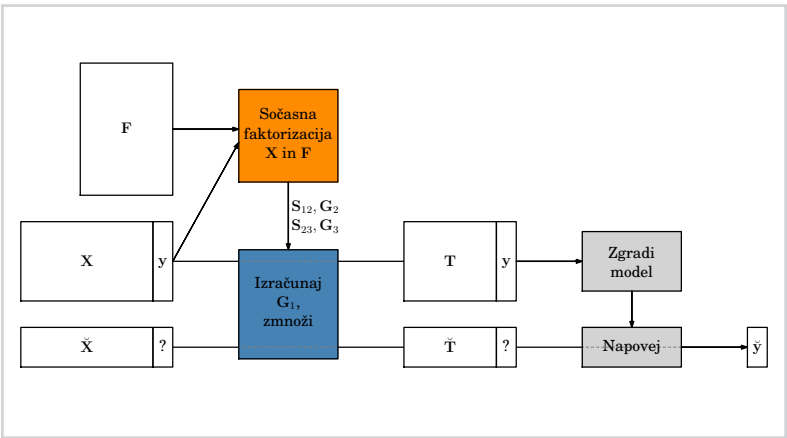
4.3 *Eksperimenti*

4.3.1 *Metode za primerjavo*

Predlagano metodo smo primerjali z izbranimi tehnikami transformacije značilk iz razdelka 2.2. Od opisanih smo izbrali tehnike, ki so se bodisi obnesle dobro v naših predhodnih poskusih bodisi so metodološko relevantne.

Zmnožek originalnih matrik $\mathbf{X}$ *in* $\mathbf{F}$ Vsako skupino značilk lahko predstavimo s povprečjem vrednosti pripadajočih značilk. Prvi poskusi transformacije iz prostora genov v prostor genskih skupin so uporabljali aritmetično sredino ali mediano [14]. Če aritmetično sredino značilk pomnožimo s številom značilk v skupini, dobimo enako vrednost kot z zmnožkom matrik $\mathbf{XF}$. Modeli, ki pred učenjem standardizirajo značilke, bodo zato vračali enake rezultate za transformaciji s povprečji in z zmnožkom $\mathbf{XF}$. Zmnožek $\mathbf{XF}$ zato predstavlja naravno osnovo, ki jo želimo izboljšati.

Analiza glavnih komponent Guo et al. [14] so predlagali uporabo analize glavnih komponent (angl. principal component analysis, PCA). Vrednost, ki za nek primer opiše skupino, je deviacija tistega primera v smeri, v kateri se primeri najbolj razlikujejo, če pri računanju opazujemo le značilke iz skupine. Vrednost značilke, ki opisuje skupino, je za nek primer linearna kombinacija originalnih značilk skupine.



Slika 4.3

Napovedovanje s transformacijo podatkovnega nabora $\mathbf{X}$ v prostor, ki ga definirajo skupine značilk iz $\mathbf{F}$. Transformirati moramo tudi testno množico $\check{\mathbf{X}}$. Sočasno faktoriziramo matriki $\mathbf{X}$ in $\mathbf{F}$ in shranimo faktorje razcepa $\mathbf{S}_{12}$, $\mathbf{G}_2$, $\mathbf{S}_{23}$, $\mathbf{G}_3$. Da transformiramo potencialno novo množico primerov $\check{\mathbf{X}}$ v prostor skupin, izračunamo predstavitev novih primerov v prvem latentnem prostoru $\mathbf{G}_1$ in primere transformiramo kot $\check{\mathbf{T}} = \check{\mathbf{G}}_1\mathbf{S}_{12}\mathbf{G}_2^T\mathbf{G}_2\mathbf{S}_3\mathbf{G}_3^T$. Na transformiranem naboru podatkov lahko uporabimo katerokoli metodo za gradnjo napovednih modelov.

CORG Metoda CORG [33] izračuna povprečje podmnožice značilk iz skupine, katerih vrednosti so najbolj povezane z razredi. Povezanost značilke z razredom merijo s statistiko t in za vsako skupino požrešno dodajajo značilke v izbor (v vrstnem redu glede na povezanost značilke z razredom), dokler se statistika t povprečja značilk izbora ne neha izboljševati.

SetSig Metoda SetSig [15] oceni, ali je primer (glede na značilke iz skupine) bolj podoben prvemu ali drugemu razredu: izračuna statistiko t med Pearsonovimi koeficienti korelacije do skupin primerov iz obeh razredov.

Ločena faktorizacija vhodnih matrik Matriki $\mathbf{X}$ in $\mathbf{F}$ smo faktorizirali na enak način kot pri predlagani sočasni faktorizaciji, z algoritmom DFMF, vendar kot ločena problema. Za vsako inicializacijo začetnih faktorjev tokrat algoritem DFMF poženemo dvakrat: enkrat kot edini vhod nastavimo $\mathbf{R}_{12} = \mathbf{X}$ in dobimo razcep $\mathbf{X} \approx \mathbf{G}_1\mathbf{S}_{12}\mathbf{G}_{2,1}^T$, drugič pa vhod nastavimo $\mathbf{R}_{23} = \mathbf{F}$ in dobimo razcep $\mathbf{F} \approx \mathbf{G}_{2,2}\mathbf{F}_{23}\mathbf{G}_3^T$. Pri pretvorbi v prostor skupin $\check{\mathbf{G}}_1$ izračunamo kot pri sočasni faktorizaciji, le da je tokrat $\check{\mathbf{T}} = \check{\mathbf{G}}_1\mathbf{S}_{12}\mathbf{G}_{2,1}^T\mathbf{G}_{2,2}\mathbf{S}_3\mathbf{G}_3^T$. Da smo dobili približno podobno skupno število elementov v faktorjih razcepa, smo latentno dimenzijo matrik $\mathbf{G}_{2,1}$ in $\mathbf{G}_{2,2}$ nastavili na polovico dimenzije $\mathbf{G}_2$.

4.3.2 Simulirani podatki

Glavni vir Po vzoru podatkov o genskih izrazih smo ustvarili podatkovne nabore z $n = 50$ primeri in $m = 10000$ značilkami. Prvih 25 primerov je predstavljalo en razred, naslednjih 25 pa drugega. Značilke so bile treh vrst:

- Z različnimi porazdelitvami med razredi ($N = 100$). Vrednosti za prvi razred smo vzorčili iz normalne porazdelitve $N(-0.25, 1)$ in $N(0.25, 1)$ za drugi razred. Razlika med razredoma je bila torej 0.5.
- Korelirane značilke ($N = 100$). Vzorčili smo jih iz $N(-0.25f, 1)$ za prvi razred in $N(0.25f, 1)$ za drugega. Možne vrednosti parametra f so bile $-1 \leqslant f \leqslant 1$. Parameter f je predstavljal stopnjo korelacije z značilkami iz prejšnje skupine.
- Naključne značilke ($N = 9800$). Vrednosti smo ne glede na razred vzorčili iz normalne porazdelitve $N(0, 1)$.

Za vsako vrednost parametra f koreliranih značilk smo generirali 20 podatkovnih naborov z različnimi semeni generatorja naključnih števil.

Viri skupin značilk Ustvarili smo 100 skupin s 100 značilkami dveh tipov:

- Skupine z značilkami z različnimi porazdelitvami med razredi ($N = 10$). Vsaka skupina vsebuje 10 značilk z različnimi porazdelitvami med razredi, 90 naključnih značilk in nobene korelirane značilke.
- Skupine značilk brez različnih porazdelitev med razredi ($N = 90$). Vsaka skupina vsebuje 100 naključnih značilk ter nobene značilke iz skupine značilk z različnimi porazdelitvami med razredi ali skupine koreliranih značilk.

4.3.3 Resnični podatki

Podatki o genskih izrazih Uporabili smo 42 naborov podatkov o človekovih genskih izrazih, ki smo jih opisali v razdelku 2.3.1.

Viri skupin značilk Skupine značilk smo prebrali iz baze MSigDB [6]. Uporabili smo skupine genov iz baze MSigDB verzije 5.1, ki so jih pridobili iz baze KEGG (oznaka C2.CP.KEGG). Pri primerjavi z naključnimi skupinami genov smo uporabili še večji vir skupin: skupine z oznakami C2.CP, C5.MF in C5.BP iz baze MSigDB 3.0.

4.3.4 *Potek eksperimentov*

Učne metode Izvorne podatke in podatke transformirane v prostor značilk smo klasificirali z regularizirano logistično regresijo in naključnimi gozdovi, kot smo opisali v razdelku 2.3.2. Na simuliranih podatkih regularizacijskega parametra logistične regresije nismo nastavljali znotraj prečnega preverjanja, ampak smo uporabili $C = 1$.

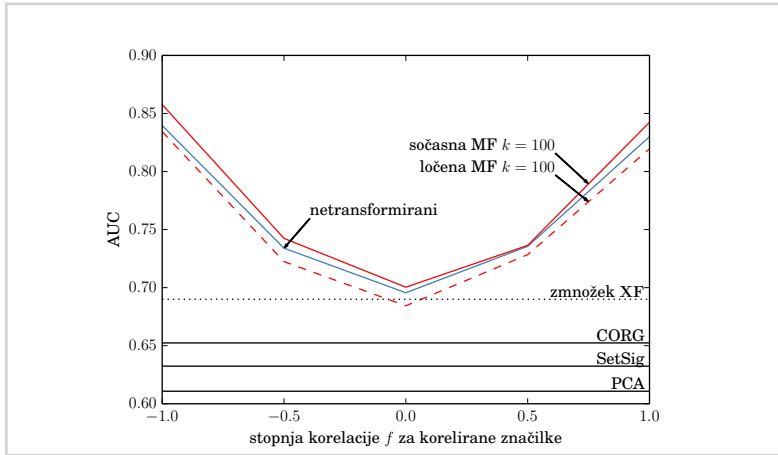
Parametri faktorizacije Latentne dimenzije vseh izhodnih matrik sočasne faktorizacije so bile znotraj enega poskusa vse enake k. Ker s poskusi na sintetičnih podatkih zgolj ilustriramo delovanje metode, smo zanje k nastavili na vrednost 100, ki je na pravih podatkih v povprečju dala najboljše rezultate pri klasifikaciji z logistično regresijo. Za vsake vhodne podatke smo naredili $p = 3$ ponovitev faktorizacije z DFMF s 100 iteracijami. Za inicializacijo začetnih približkov matrik $\mathbf{G}_1$, $\mathbf{G}_2$ in $\mathbf{G}_3$ smo uporabili algoritem RandomC [102]. Pri poskusih na resničnih podatkih smo k z notranjim 5-kratnim prečnim preverjanjem izbirali iz množice $\{10, 20, 30, \ldots, 150\}$.

Testiranje Testirali smo s 5-kratnim prečnim preverjanjem, ki smo ga pognali dvakrat. Popularne metode stremenca (angl. bootstrap) nismo uporabili, ker morajo biti za pravilno izvedeno notranje prečno preverjanje vsi učni primeri različni (metoda stremenca dovoli, da se primeri ponovijo).

4.4 *Rezultati in diskusija*

4.4.1 *Sočasna faktorizacija uporabi značilke, ki jih ni v skupinah*

Slika 4.4 prikazuje površino pod krivuljo ROC (angl. area under ROC curve, AUC) [81] za klasifikatorje zgrajene z logistično regresijo na transformiranih in originalnih sintetičnih podatkih. Rezultati so povprečja čez 20 ponovitev generiranja podatkov. Osnovna transformacija, ki zmnoži originalni matriki $\mathbf{X}$ in $\mathbf{F}$, se obnese podobno kot predlagana sočasna faktorizacija, če značilk, ki so korelirane z razredno spremenljivko, a jih ni v skupinah značilk $\mathbf{F}$, ni v podatkih (pri $f = 0$). Ker skupine značilk ne vsebujejo nobene korelirane značilke, jih množenje originalnih matrik ne upošteva in vrača ne glede na vrednost f vedno enake rezultate. Tudi druge obstoječe metode ne uporabljajo signala v koreliranih značilkah. Če se signal v koreliranih značilkah okrepi, se izboljšajo le rezultati metod, ki temeljijo na matrični faktorizaciji, saj ostale metode signala v koreliranih značilkah ne morejo uporabiti, ker korelirane značilke niso vključene v skupine.

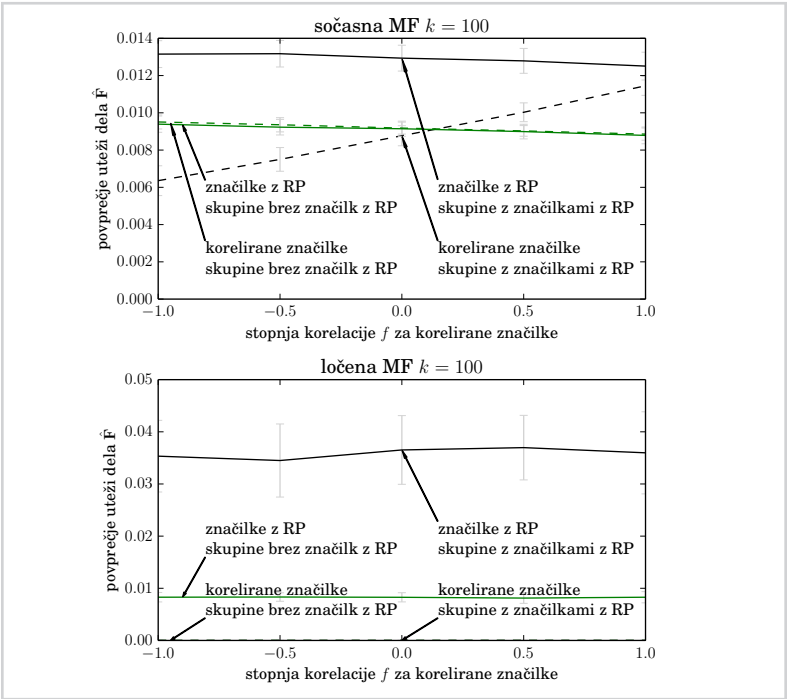


Slika 4.4

Točnost napovednih modelov logistične regresije, ki jih zgradimo na originalnih in transformiranih sintetičnih podatkih v odvisnosti od faktorja f, ki določa stopnjo informacije o razredih v koreliranih značilkah (le-teh ni v nobeni skupini značilk). Rezultati metod, ki temeljijo na matrični faktorizaciji, se izboljšajo, če se signal v značilkah, ki jih ni v skupinah **F**, poveča. Druge metode ne uspejo izrabiti signala v koreliranih značilkah.

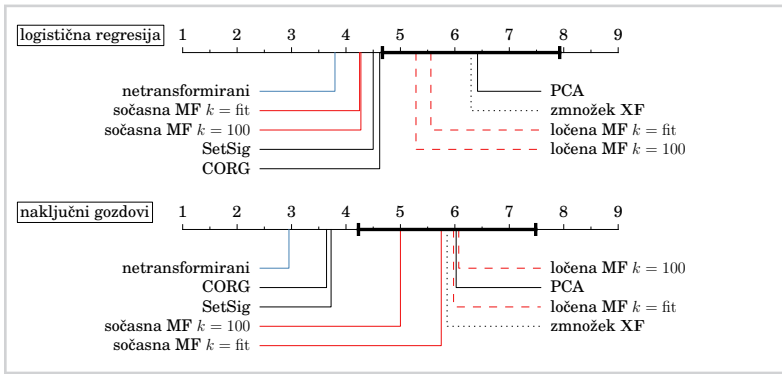
4.4.2 *Značilke, ki jih ni v skupinah, vplivajo na rekonstrukcijo matrike skupin*

Tako sočasna kot ločena matrična faktorizacija sta uspeli uporabiti informacije iz skupine koreliranih genov (slika 4.4). Ker so korelirane značilke del izvorne matrike $\mathbf{X}$, gotovo vplivajo na njeno rekonstrukcijo $\hat{\mathbf{X}}$. Slika 4.5 kaže, da pri sočasni faktorizaciji korelirane značilke iz $\mathbf{X}$ vplivajo tudi na rekonstrukcijo matrike $\hat{\mathbf{F}}$, saj je latentni faktor $\mathbf{G}_2$ del obeh rekonstruiranih matrik $\hat{\mathbf{X}} = \mathbf{G}_1\mathbf{S}_{12}\mathbf{G}_2^T$ in $\hat{\mathbf{F}} = \mathbf{G}_2\mathbf{S}_{23}\mathbf{G}_3^T$. Ker v optimizacijskem problemu nismo omejili zaloge vrednosti rekonstruirane matrike, so vrednosti $\hat{\mathbf{F}}$ poljubna realna števila. Pri ločeni faktorizaciji so uteži, ki opisujejo uvrščenost koreliranih značilk v skupine, ostale enake 0. Nasprotno so pri sočasni faktorizaciji vrednosti, ki opisujejo uvrščenost korelirane značilke v skupine z različnimi porazdelitvami med razredi, večje od 0 (slika 4.5). S tem smo skupine efektivno razširili z značilkami, ki so podobne značilkam v skupini. Učinek je odvisen od faktorja f. Za negativne vrednosti f so uteži koreliranih značilk v skupinah z značilkami z različnimi porazdelitvami med razredi v povprečju nižje kot uteži značilk z različnimi vrednostmi med razredi v skupinah z zgolj naključnimi značilkami. Slednje smo pričakovali, ker za negativne f korelirane značilke na napoved vplivajo v obratni smeri glede na značilke z različnimi porazdelitvami med razredi.



Slika 4.5

Povprečje skupine uteži v rekonstruirani matriki skupin $\hat{\mathbf{F}}$ v odvisnosti od faktorja korelacije f. Sočasna faktorizacija (zgoraj) koreliranim značilkam (povezane z razredno spremenljivko, a jih ni v skupinah $\mathbf{F}$; črtkane črte) v rekonstruirani matriki $\hat{\mathbf{F}}$ nastavi uteži o pripadnosti genskim skupinam. Uteži v $\hat{\mathbf{F}}$ so odvisne od faktorja korelacije f. Pri ločeni faktorizaciji (spodaj) so uteži za korelirane značilke v rekonstrukciji $\hat{\mathbf{F}}$ ostale enake nič. Črna barva označuje skupine z značilkami z različnimi porazdelitvami med razredi (RP), zelena skupine brez.



Slika 4.6

Točnost klasifikatorjev logistične regresije in naključnih gozdov, ki smo jih zgradili na originalnih podatkih in podatkih, ki smo jih transformirali z različnimi metodami. Slika prikazuje povprečne uvrstitve AUC čez $N = 42$ naborov podatkov. Okrog zmnožka **XF** z debelo črto prikazujemo interval, na katerem se metode od zmnožka ne razlikujejo značilno (test Bonferonni-Dunn, $\alpha = 0.05$).

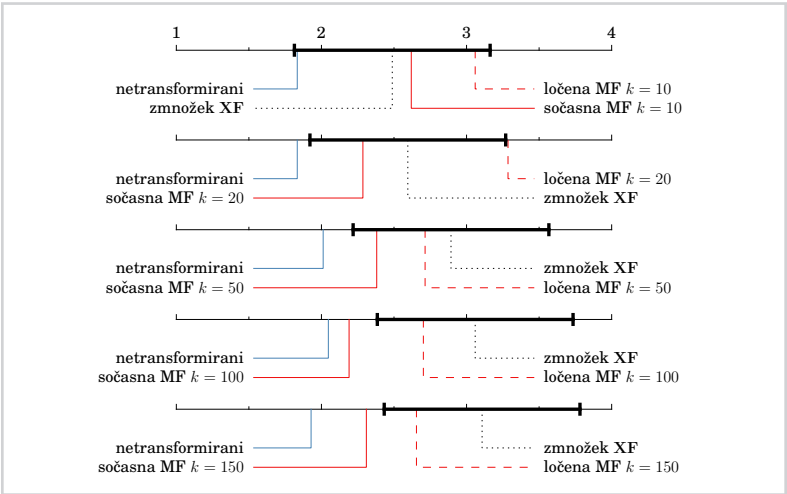
4.4.3 *Razlike med metodami na resničnih naborih podatkov niso velike*

Na resničnih podatkih med transformacijskimi metodami ni velikih razlik (slika 4.6). Če s testom Bonferonni-Dunn [103] primerjamo vse prikazane metode z zmnožkom izvornih matrik **XF**, je pri klasifikaciji z logistično regresijo povprečna uvrstitev sočasne faktorizacije značilno boljša od zmnožka **XF** ($\alpha = 0.05$), uvrstitev ločene faktorizacije pa ne. Pri klasifikaciji z naključnimi gozdovi se sočasna faktorizacija sicer še vedno obnese bolje od ločene, a nobena ni značilno boljša od zmnožka **XF**. Značilno bolje od njega se obneseta metodi CORG in SetSig. Iz rezultatov v tabelah 4.1 in 4.2 razberemo, da so povprečni AUC čez vse nabore podatkov za posamezno metodo pri klasifikaciji z logično regresijo vedno višji od povprečnih AUC z naključnimi gozdovi, ob čemer se najbolj razlikujejo prav metode matrične faktorizacije (za najmanj 0.035), najmanj pa se spremenita rezultata CORG in SetSig (za največ 0.007).

Glede na naše rezultate ni nobena transformacijska metoda presegla rezultatov na netransformiranih podatkih, so pa rezultati primerljivi, kar potrjuje rezultate predhodnih primerjalnih študij metod za transformacijo v prostor genskih skupin [16, 18, 104].

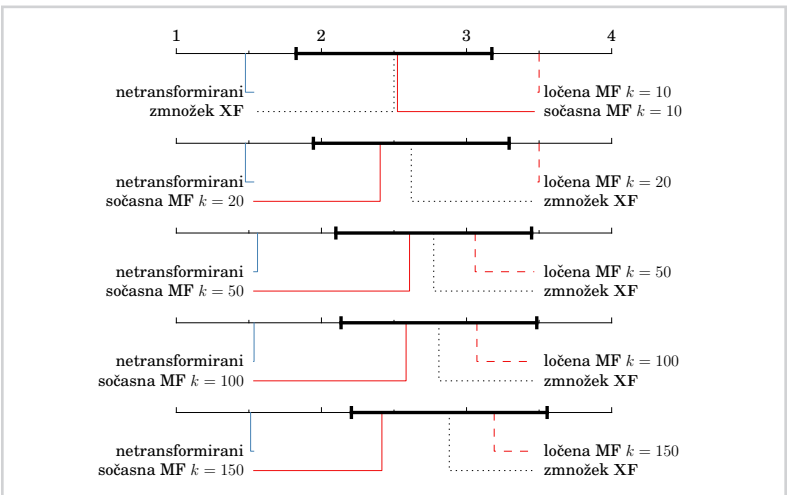
4.4.4 *Vpliv ranga latentnih faktorjev na rezultate*

Pri klasifikaciji z logistično regresijo smo v povprečju najboljše rezultate dobili z rangom latentnih faktorjev $k = 100$ (za vse matrike smo uporabili enak rang). Slika 4.7 prikazuje, da se pri logistični regresiji AUC povečuje s povečevanjem k do $k = 100$ in nato pri $k = 150$ pade. Pri klasifikaciji z naključnimi gozdovi (slika 4.8) s $k = 10$ tudi s soča-



Slika 4.7

Povprečni rangi AUC čez ($N = 42$) naborov podatkov za različno število latentnih komponent k, če smo za klasifikacijo uporabili logistično regresijo.



Slika 4.8

Povprečni rangi AUC čez ($N = 42$) naborov podatkov za različno število latentnih komponent k, če smo za klasifikacijo uporabili naključne gozdove.

Tabela 4.1

Rezultati AUC primerjanih metod na resničnih podatkih o genskih izrazih, kjer skupine definirajo genske poti iz baze KEGG . Za klasifikacijo smo uporabili logistično regresijo.

nabor podatkov	primerov	značilk	skupin	netransformirani	sočasna MF k = fit	sočasna MF $k = 100$	SetSig	CORG	PCA	zmnožek **XF**	ločena MF k = fit	ločena MF $k = 100$
DLBCL	77	6219	180	.979	.991	.993	.997	.996	.948	.973	.989	.991
GDS232	23	932	93	.608	.717	.733	.583	.556	.567	.675	.685	.731
GDS531	173	9459	181	.758	.750	.766	.750	.709	.700	.725	.733	.754
GDS806	60	20007	182	.709	.684	.683	.694	.680	.700	.632	.661	.711
GDS971	23	9695	182	.825	1.000	.975	1.000	.967	.908	.900	1.000	.950
GDS1059	53	6200	180	.710	.709	.737	.665	.667	.611	.685	.676	.681
GDS1062	27	14903	182	.608	.675	.646	.754	.727	.676	.854	.671	.633
GDS1209	54	14903	182	.983	.983	.983	.963	.971	.967	.979	.979	.983
GDS1210	30	6277	181	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1220	54	14903	182	.894	.900	.900	.911	.900	.889	.894	.900	.900
GDS1221	28	9697	182	.528	.586	.495	.639	.613	.389	.480	.529	.531
GDS1282	32	14903	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1329	43	14902	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1375	63	14903	182	.974	.989	.989	.993	.993	.993	.996	.967	.963
GDS1390	20	14903	182	.861	.833	.806	.806	.750	.639	.917	.778	.833
GDS1562	28	1394	62	1.000	.989	.989	.989	.989	.950	.900	.978	.978
GDS1618	20	14903	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1650	39	9697	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1667	36	34700	181	.990	.948	.990	1.000	1.000	.973	.992	.940	.957
GDS1714	28	16246	182	.806	.783	.833	.778	.856	.717	.617	.750	.717
GDS1887	46	9697	182	.476	.326	.405	.325	.469	.417	.550	.458	.421
GDS2113	75	14903	182	.652	.645	.605	.584	.532	.511	.579	.661	.586
GDS2201	37	14903	182	1.000	1.000	1.000	.950	.958	.975	.992	1.000	1.000
GDS2250	38	34700	181	.988	.994	.994	1.000	1.000	.988	.969	.994	.994
GDS2415	59	14345	183	.761	.696	.720	.719	.736	.648	.653	.690	.742
GDS2489	44	6278	181	.990	1.000	1.000	.980	1.000	1.000	.994	1.000	1.000
GDS2520	44	9697	182	.978	.956	.956	.982	.955	.941	.930	.940	.925
GDS2547	122	10629	151	.779	.777	.767	.747	.748	.742	.695	.775	.767
GDS2609	22	34700	181	.867	.842	.842	.917	.950	.942	.892	.842	.842
GDS2735	46	18022	181	.672	.610	.605	.629	.643	.695	.651	.569	.530
GDS2771	187	14902	182	.801	.762	.759	.755	.759	.749	.727	.758	.758
GDS2785	43	9697	182	.832	.879	.748	.839	.806	.749	.745	.789	.744
GDS2842	36	9697	182	.625	.552	.637	.628	.472	.395	.476	.547	.577
GDS3268	202	29391	181	.925	.880	.879	.864	.866	.841	.787	.859	.877
GDS3929	183	19136	180	.486	.412	.357	.401	.530	.540	.446	.395	.387
GDS3952	88	31623	181	.456	.472	.461	.409	.479	.433	.393	.449	.499
GDS4228	166	3779	161	.846	.758	.765	.742	.763	.728	.682	.724	.708
GDS4228_agent	125	3779	161	.881	.865	.864	.854	.838	.833	.827	.855	.868
GSE412	110	6776	180	.968	.958	.958	.980	.918	.923	.902	.951	.924
GSE3726	52	14166	182	.940	.945	.946	.958	.945	.941	.929	.934	.932
leukemia	72	4680	177	.996	.986	.992	.998	.974	.994	.996	.988	.988
prostata	102	9582	182	.968	.952	.959	.937	.970	.919	.933	.953	.914
povprečje	66	13686	174	.836	.829	.827	.827	.826	.798	.809	.818	.817

Tabela 4.2

Rezultati AUC primerjanih metod na resničnih podatkih o genskih izrazih, kjer skupine definirajo genske poti iz baze KEGG. Za klasifikacijo smo uporabili naključne gozdove.

nabor podatkov	primerov	značilk	skupin	netransformirani	sočasna MF k = fit	sočasna MF k = 100	SetSig	CORG	PCA	zmnožek **XF**	ločena MF k = fit	ločena MF k = 100
DLBCL	77	6219	180	.960	.926	.917	.990	.978	.913	.934	.941	.923
GDS232	23	932	93	.491	.658	.692	.602	.667	.625	.528	.679	.769
GDS531	173	9459	181	.735	.677	.704	.736	.680	.607	.672	.698	.707
GDS806	60	20007	182	.748	.722	.725	.710	.723	.700	.685	.784	.708
GDS971	23	9695	182	1.000	.925	.950	1.000	.975	.917	.875	.796	.875
GDS1059	53	6200	180	.677	.676	.689	.645	.681	.687	.710	.664	.687
GDS1062	27	14903	182	.746	.700	.717	.754	.692	.677	.697	.681	.725
GDS1209	54	14903	182	.996	.988	.983	.963	.983	.979	1.000	1.000	.988
GDS1210	30	6277	181	1.000	1.000	.988	1.000	1.000	1.000	1.000	1.000	1.000
GDS1220	54	14903	182	.928	.905	.916	.911	.928	.880	.883	.940	.919
GDS1221	28	9697	182	.492	.435	.389	.551	.486	.383	.365	.325	.443
GDS1282	32	14903	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1329	43	14902	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	.990
GDS1375	63	14903	182	.996	.996	.996	.996	.996	.993	.994	.981	.996
GDS1390	20	14903	182	.861	.806	.833	.861	.833	.812	.844	.812	.778
GDS1562	28	1394	62	.978	.978	.967	1.000	.961	.933	.883	.981	.961
GDS1618	20	14903	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1650	39	9697	182	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
GDS1667	36	34700	181	1.000	.940	.915	1.000	1.000	.925	.980	.929	.910
GDS1714	28	16246	182	.822	.706	.759	.783	.817	.722	.744	.707	.600
GDS1887	46	9697	182	.390	.446	.498	.345	.402	.499	.488	.448	.476
GDS2113	75	14903	182	.610	.556	.581	.570	.560	.572	.580	.514	.570
GDS2201	37	14903	182	1.000	.855	.831	.933	.975	.762	.762	.781	.807
GDS2250	38	34700	181	1.000	.825	.835	1.000	1.000	1.000	.844	.795	.833
GDS2415	59	14345	183	.727	.727	.732	.708	.742	.629	.665	.673	.699
GDS2489	44	6278	181	1.000	.964	.974	.989	1.000	.919	.989	.978	.945
GDS2520	44	9697	182	.976	.930	.950	.966	.946	.823	.905	.939	.930
GDS2547	122	10629	151	.766	.760	.755	.757	.735	.725	.678	.741	.759
GDS2609	22	34700	181	.983	.825	.842	.917	.925	.917	.867	.823	.825
GDS2735	46	18022	181	.522	.306	.349	.433	.453	.384	.429	.277	.346
GDS2771	187	14902	182	.748	.670	.680	.792	.749	.718	.697	.659	.674
GDS2785	43	9697	182	.829	.801	.807	.849	.837	.740	.718	.776	.748
GDS2842	36	9697	182	.557	.497	.470	.618	.517	.503	.463	.493	.448
GDS3268	202	29391	181	.834	.756	.745	.801	.808	.734	.747	.729	.731
GDS3929	183	19136	180	.495	.434	.451	.428	.478	.535	.505	.489	.447
GDS3952	88	31623	181	.460	.485	.513	.430	.470	.507	.513	.525	.516
GDS4228	166	3779	161	.720	.710	.704	.754	.739	.743	.708	.715	.706
GDS4228_agent	125	3779	161	.852	.824	.832	.838	.855	.834	.814	.830	.812
GSE412	110	6776	180	.917	.864	.865	.941	.908	.859	.848	.864	.865
GSE3726	52	14166	182	.967	.870	.848	.958	.981	.884	.881	.829	.822
leukemia	72	4680	177	.998	.976	.973	.985	.998	.979	.960	.970	.965
prostata	102	9582	182	.939	.867	.890	.912	.943	.810	.876	.844	.869
povprečje	66	13686	174	.827	.785	.792	.820	.820	.782	.779	.776	.780

sno faktorizacijo dobimo slabše rezultate kot z zmnožkom **XF**, za večje k pa ne vidimo jasnega učinka. V povprečju je sočasna faktorizacija vedno pred ločeno. Spremembe točnosti glede na k po posameznih naborih podatkov prikazujejo slike A.1–A.4.

Izbor ustreznega ranga faktorizacije je precej pomemben. Menimo, da je z boljšim izborom ranga latentnih faktorjev mogoče rezultate še izboljšati. Avtorjem algoritma DFMF se je obneslo, če so range latentnih faktorjev nastavljali v sorazmerju s številom objektov tistega tipa [26].

4.4.5 Razširjene skupine so smiselne

Na resničnih podatkih smo iz vsake genske skupine odstranili 10% naključnih genov. Za vsak nabor primarnih podatkov posebej smo s sočasno matrično faktorizacijo (z rangom 100) faktorizirali vhodne podatke in matriko skupin značilk. Nato smo v rekonstruirani matriki $\hat{\mathbf{F}}$ opazovali, kakšne uteži imajo odstranjeni geni. Za vsako skupino, kjer smo odstranili vsaj 3 gene, smo ocenili, kako se uteži odstranjenih genov uvrstijo glede na ostale gene v skupini. Uporabili smo mero AUC: uteži skupine smo uredili glede na njihovo absolutno vrednost, odstranjenim genom pripisali vlogo pozitivnega razreda, ostalim negativnega. Za vsak nabor podatkov smo vrednosti AUC, ki opisujejo kvaliteto uvrstitve genov, povprečili čez vse skupine.

V povprečju (po 5 ponovitvah poskusa in 42 naborih podatkov) smo dobili AUC 0.733. Najmanjši AUC na posameznem naboru podatkov je bil 0.673, največji pa 0.853. Ker so rezultati boljši od tistega, ki bi ga dobili, če bi bile uteži odstranjenih genov naključne (dobili bi 0.5), sklepamo, da sočasna matrična faktorizacija skupine interno razširja na smiseln način.

4.4.6 Vpliv naključnih genskih skupin

Naključne genske skupine smo ustvarili tako, da smo premešali gene med pravimi genskimi skupinami: nek gen smo v vseh genskih skupinah zamenjali z istim naključnim genom, pri čemer so genske skupine ohranile svoje velikosti. Ustvarili smo 20 različic naključnih genskih skupin, vsako s svojim semenom generatorja naključnih števil. Opazovali smo, kakšen AUC dobimo s pravimi genskimi skupinami v primerjavi z naključnimi skupinami (pri klasifikaciji z logistično regresijo). Za nabor podatkov d smo izračunali p-vrednost p_d kot delež rezultatov, kjer so bili AUC pravih genskih skupin boljši kot rezultati na naključnih genskih skupinah.

Da bi povzeli rezultate vseh 42 naborov podatkov, smo delež naborov podatkov s

Tabela 4.3

Stolpci D_α prikazujejo deleže naborov podatkov, na katerih je p_d (delež rezultatov, kjer so naključne skupine boljše od pravih) manjši od α, stolpci $p(D_{0.05})$ pa s permutacijskim testom ocenjene p-vrednosti tega rezultata.

a) Manj skupin (C2.CP.KEGG)

	$D_{0.05}$	$p(D_{0.05})$	$D_{0.10}$	$p(D_{0.10})$	$D_{0.15}$	$p(D_{0.15})$	$D_{0.20}$	$p(D_{0.20})$
sočasna MF $k = 100$	.071	.136	.119	.083	.143	.185	.167	.253
zmnožek **XF**	.071	.208	.119	.239	.167	.219	.214	.174
CORG	.071	.148	.119	.107	.190	.022	.214	.052
PCA	.048	.423	.048	.833	.095	.690	.143	.588

b) Več skupin (C2 + C5.BP + C5.MF)

	$D_{0.05}$	$p(D_{0.05})$	$D_{0.10}$	$p(D_{0.10})$	$D_{0.15}$	$p(D_{0.15})$	$D_{0.20}$	$p(D_{0.20})$
sočasna MF $k = 100$	.119	.007	.167	.011	.190	.024	.286	.002
zmnožek **XF**	.048	.437	.048	.770	.048	.923	.071	.922
CORG	.024	.701	.095	.221	.143	.165	.143	.412
PCA	.000	1.000	.024	.932	.024	.982	.119	.636

p_d manjšo od α označili z D_α in jo ovrednotili s permutacijskim testom. V permutacijskem testu smo 100.000-krat za vsak nabor podatkov naključno premešali prave in naključne AUC. Za vsako od 100.000 ponovitev smo izračunali $\hat{D}_\alpha$. Končna p-vrednost $p(D_\alpha)$ je delež $\hat{D}_\alpha$ večjih od prave D_α.

Rezultate permutacijskega testa prikazuje tabela 4.3. Kadar smo uporabili manjši vir skupin, v povprečju s 174 uporabljenimi skupinami na nabor podatkov, nobena metoda ni dosegla posebno nizkih p-vrednosti. Na podatkih z več, v povprečju s 1850 uporabljenimi skupinami, vidimo, da sočasna matrična faktorizacija na pravih genskih skupinah vrača opazno boljše rezultate kot na naključnih genskih skupinah (vsi $p(D)$ so manjši od 0.05). Preostale metode (zmnožek **XF**, CORG in PCA) se na naključnih skupinah ne obnesejo slabše kot na pravih, kar so opazili tudi v drugih študijah [18, 104]. Iz tega lahko sklepamo, da transformacija s sočasno matrično faktorizacijo strukturi skupin daje večjo težo kot preostale preizkušene metode.

4.5 Zaključek

Predlagana metoda za pretvorbo v prostor značilk s sočasno matrično faktorizacijo deluje z logistično regresijo primerljivo dobro kot ostale metode, ki so jih predlagali za napovedovanje podatkov o genskih izrazih s predznanjem o genskih skupinah, z naključnimi gozdovi pa ne vrača dobrih rezultatov. V nasprotju z nekaterimi metodami, kot sta na primer CORG ali SetSig, predlagana transformacija ne uporablja vrednosti razredne spremenljivke, zato deluje tudi na podatkih, kjer razredna spremenljivka ni le binarna. Lahko jo uporabimo tako za razvrščanje v skupine, kot tudi za klasifikacijo in regresijo.

Rezultati transformacije s sočasno faktorizacijo so bili boljši kot z uporabo ločene transformacije. Pokazali smo, da sočasna faktorizacija uporabi tudi značilke, ki so podobne značilkam v skupini, a ji ne pripadajo. Ker v praksi pogosto ne poznamo vseh značilk, ki določeni skupini pripadajo, nam interno razširjanje skupin lahko koristi. Na resničnih podatkih pokažemo, da so razširitve, ki jih transformacija s sočasno matrično faktorizacijo uporablja, smiselne.

5

Napovedovanje s skladanjem transformiranih vrednosti

5.1 Uvod

Pri napovedovanju z vnaprejšnjo transformacijo iz prostora originalnih značilk v prostor skupin značilk moramo nastaviti parametre transformacijskih modelov. Pri najpreprostejši transformaciji, ki izračuna povprečno vrednost značilk v skupini, so parametri transformacije značilke, ki pripadajo skupini. Nastavimo jih lahko brez pregleda originalnih podatkov: zadostujejo podatki o skupinah. Kompleksnejša transformacija, ki vsako skupino oceni z razdaljo v smeri prve glavne komponente vrednosti značilk iz skupine, zahteva izračun prvega lastnega vektorja. Za transformacijo z metodo glavnih komponent za nastavitev parametrov internega modela torej potrebujemo učne podatke, ne pa tudi ciljne spremenljivke.

Nekatere transformacije pri izračunu parametrov internega modela transformacije uporabijo tudi ciljno spremenljivko [15, 30–33, 35, 36]: značilke, ki opisujejo skupine osnovnih značilk, se vsaj deloma prilagodijo ciljni spremenljivki. Če interne parametre transformacije nastavimo na celotnih učnih podatkih, učne podatke z njimi transformiramo in na transformiranih učnih podatkih zgradimo napovedni model, se lahko zgodi, da napovedni model za nove, še nevidene primere, ne bo dobro deloval. Učni algoritem, ki gradi napovedni model, po transformaciji celotnih učnih podatkov namreč ne more razločiti, ali neka transformirana značilka dobro opisuje razred ali pa so se parametri transformacije pretirano prilagodili razredu. V obeh primerih je korelacija transformirane značilke in razreda visoka. Zato bo napovedni model značilkam, ki so preveč prilagojene razredu, pripisal prevelik pomen.

Transformacijske metode, ki ciljne spremenljivke ne uporabljajo, te težave nimajo. Precej obstoječih raziskav, ki se ukvarja s transformacijskimi metodami z uporabo ciljne spremenljivke, težave s preveliko prilagoditvijo ne poskuša reševati, niti je ne komentira [15, 16, 18, 34, 37, 79]. Druge raziskave problem rešujejo z uporabo različnih delov učne množice podatkov za nastavitev parametrov transformacije in izbor transformiranih značilk: ali z razbitjem učnih podatkov na dve neprekrivajoči množici [33, 35] ali s prečnim preverjanjem [31]. Oba pristopa odstranita značilke, ki opisujejo skupine, katerih transformacije dobro opišejo le podmnožico, ki smo jo uporabili za nastavitev parametrov transformacije. Tako v učni množici, na kateri gradimo napovedni model, ni preveč prilagojenih značilk skupin. A popolna odstranitev morda ni najboljša rešitev. Ker lahko preveč prilagojena značilka vendarle nosi nekaj koristne informacije o ciljni spremenljivki, bi nam pri napovedovanju lahko pomagala, če bi jo smiselno,

morda z nižjo utežjo, vključili v končni napovedni model.

Skladanje (angl. stacking) uporabljamo za združevanje napovedi modelov, ki smo jih na isti učni množici zgradili z različnimi algoritmi [38]. Če bi denimo napovedi združevali tako, da bi izračunali povprečno napoved vseh napovednih modelov, bi na končno napoved vsi modeli vplivali v enaki meri. Ker pa za vse učne primere vsi napovedni modeli niso enako dobri, skladanje poskuša upoštevati njihovo kakovost tako, da se nauči modela za združevanje napovedi različnih algoritmov. Ker želimo doseči dobro napovedno točnost za nove, še nevidene primere, skladanje model za združevanje algoritmov gradi na napovedih primerov, ki jih pri gradnji osnovnih napovednih modelov nismo uporabili.

V tem poglavju predlagamo metodo, ki rešuje problem ciljni spremenljivki preveč prilagojenih transformacij značilk. Predlagana metoda deluje na podlagi skladanja klasifikatorjev. V nasprotju z obstoječimi predlogi za uporabo transformiranih značilk, vse transformirane značilke obdrži vse do gradnje končnega napovednega modela.

5.2 Načini transformacije v značilke skupin

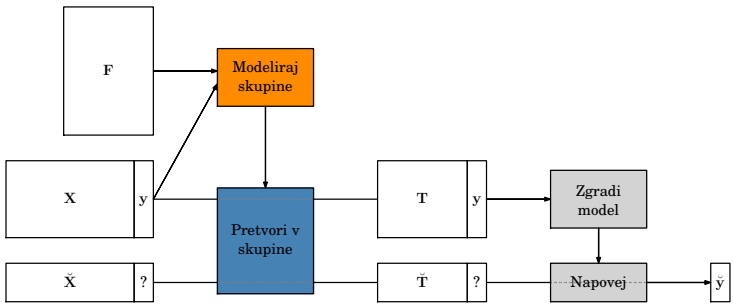
5.2.1 Uporaba vseh podatkov za transformacijo

Najpreprosteje je interne parametre transformacije nastaviti na celotni učni množici, z njimi transformirati učno množico ter na transformirani množici zgraditi napovedni model. Pri napovedovanju ciljne spremenljivke testnemu primeru z na učni množici naučenimi parametri transformacije transformiramo testni primer. Nato z že zgrajenim napovednim modelom napovemo ciljno spremenljivko novemu primeru (slika 5.1).

Kot smo že omenili, je slabost takega pristopa potencialno preveliko prilagajanje transformacije vrednostim ciljne spremenljivke, a ga kljub temu precej študij vseeno uporablja [15, 16, 18, 34, 37, 79].

5.2.2 Obstoječe rešitve pretirane prilagoditve značilk skupin

Lee et al. [33] so predlagali postopek, kjer so 2/3 učnih primerov uporabili za izračun parametrov transformacije v skupine, preostalo 1/3 pa za izbiro značilk, ki so opisovale skupine. Značilke skupin so izbirali tako, da so značilke uredili po p-vrednosti Studentovega t-testa in v urejenem zaporedju značilke dodajali logistični regresiji, dokler so se rezultati izboljševali. Po vzoru Lee et al. [33] so enako obravnavo transformiranih značilk uporabili še Su et al. [35].



Slika 5.1

Napovedovanje z značilkami skupin, kjer skupine iz **F** modeliramo na celotni množici učnih podatkov **X**. Zgrajene modele skupin (za vsako skupino enega) uporabimo na celotni množici učnih in testnih podatkov. Napovedni model zgradimo na transformirani učni množici.

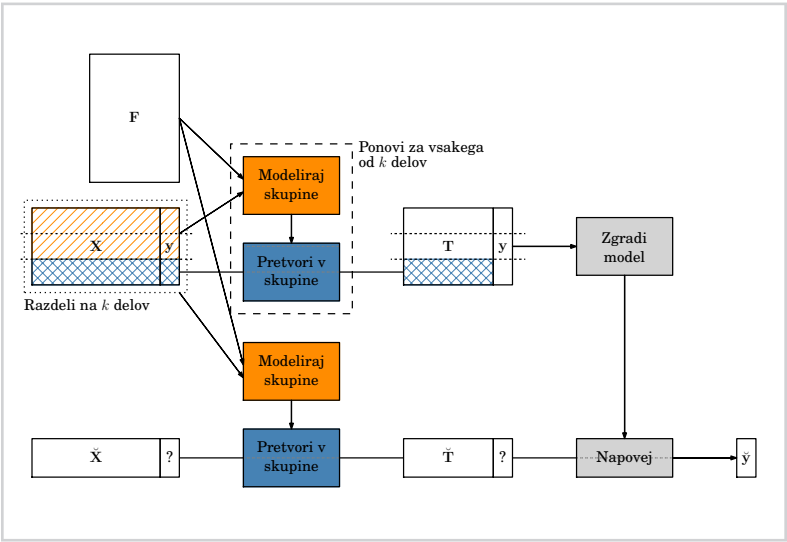
Liu et al. [31] so predlagali izbor najboljših značilk s prečnim preverjanjem. Ocenili so napovedno točnost vseh transformiranih značilk (vsako posebej) ter izbrali zgolj tiste, katerih napovedna točnost je presegla vnaprej nastavljen prag.

Naš predlagan pristop smo primerjali zgolj z metodo Lee et al. [33], ki ji ni treba nastavljati nobenih parametrov.

5.2.3 *Skladanje transformiranih vrednosti*

Skladanje (angl. stacking) združuje več napovednih algoritmov [38]. Recimo, da združujemo napovedne algoritme učnih metod $L_1, L_2, \ldots, L_l$ (metode osnovnega nivoja oziroma nivo 0) z napovednim algoritmom za združevanje L_Z (nivo 1). Skladanje poteka v treh fazah:

1. Gradnja modela za združevanje. Za gradnjo tabele za učenje modela nivoja 1, ki združuje napovedi osnovnih algoritmov, učne podatke $\mathbf{X}$ razdelimo na k disjunktnih podmnožic približno enake velikosti $\mathbf{X}_1, \ldots, \mathbf{X}_k$. Tabelo za združevanje napovedi gradimo s prečnim preverjanjem: za vsak $i \in \{1, \ldots, k\}$ generiramo učno množico kot unijo $\mathbf{X}_i^{\text{train}} = \cup_{j \in \{1,\ldots,k\}-\{i\}} \mathbf{X}_j$, vseh l metod osnovnega nivoja naučimo na $\mathbf{X}_i^{\text{train}}$, z njimi napovemo primere iz $\mathbf{X}_i$ in napovedi shranimo v tabelo. Tabeli za združevanje primerov ciljno spremenljivko nastavimo na pravo vrednost ciljne spremenljivke tistega primera in na njem zgradimo model, ki združuje napovedi M_Z.



Slika 5.2

Napovedovanje s skladanjem transformiranih vrednosti. Podatke za učenje napovednega modela ustvarimo tako, da učne podatke razdelimo na k delov in za vsak del ponovimo: izbrani del vzamemo iz množice podatkov, na preostanku modeliramo skupine iz **F**, nato z zgrajenimi modeli skupin transformiramo izbran del podatkov. Na združenih transformiranih delih podatkov zgradimo napovedni model. Transformacije za napovedovanje novih primerov generiramo na celotni učni množici.

2. Gradnja modelov nivoja 0 za napovedovanje. Vseh l osnovnih metod naučimo na celotni učni množici $\mathbf{X}$. Dobimo M modelov $M_1, M_2, \ldots, M_l$.

3. Napovedovanje s skladanjem. Nov primer napovemo z na celotni učni množici zgrajenimi napovednimi modeli $M_1, M_2, \ldots, M_l$, končno napoved pa dobimo z združevanjem napovedi s prej zgrajenim modelom M_Z.

Skladanje spada med meta učenje (angl. meta-learning), ker nam transformacija učne množice poda informacije o napovedih osnovnih napovednih modelov [105]. Kot oblika meta učenja je skladanje precej omejeno, ker so vse uporabljene napovedne metode fiksno določene in imajo torej fiksno določen prostor hipotez, v katerem iščejo ciljno hipotezo oziroma napovedni model [105].

Če si metode za transformacijo značilk v skupino predstavljamo kot metode, ki gradijo napovedne modele, lahko skladanju podobno shemo uporabimo tudi za transformacijo podatkov v prostor skupin. Podatke za učenje končnega napovednega modela ustvarimo tako, da učne podatke razdelimo na k delov in za vsak del ponovimo: izbrani del vzamemo iz množice podatkov, na preostanku modeliramo skupine iz **F**, nato pa z zgrajenimi modeli skupin transformiramo izbran del podatkov. Na združenih

transformiranih delih podatkov zgradimo končni napovedni model. Transformacije za napovedovanje novih primerov generiramo na celotni učni množici (slika 5.2).

Par učnih množic, na katerih za nek nabor podatkov znotraj transformacije s skladanjem nastavljamo parametre transformacij posameznih skupin značilk, ima delež skupnih primerov enak $\frac{k-2}{k}$. Če je metoda transformacije takšna, da različni učni primeri njenega rezultata ne spremenijo bistveno oziroma ima majhno varianco, skladanje ne bo bistveno spremenilo rezultatov. Pri transformacijskih metodah z veliko varianco pa za različne učne množice dobimo drugačne rezultate transformacij, kar seveda vpliva na končni rezultat skladanja.

Da metodo transformacije vrednosti značilk v vrednost, ki pripada skupini, uporabimo s skladanjem, moramo metodo transformacije pognati $(k+1)$-krat: za vsakega od delov razdeljene učne množice ter še enkrat, da dobimo parametre transformacije za nove primere. Ker metode poganjamo na delu učne množice, ki je skoraj tako velik kot celotna učna množica, bomo tako za transformacije porabili skoraj $(k+1)$-krat toliko časa. Če bi skladanje izvedli analogno metodi testiranja "izpusti enega" (k je število učnih primerov m), bi za transformacijo s skladanjem porabili $(m+1)$-krat toliko časa, kot ga potrebujemo za transformacijo brez skladanja.

5.3 Metoda SetSig s skladanjem

V posebnih primerih lahko transformacijo značilk s skladanjem občutno pospešimo.

Metoda SetSig [15] za primer $\mathbf{x}$, ki ga želimo transformirati v vrednosti skupine značilk G, izračuna Pearsonove koeficiente korelacije do primerov iz obeh razredov (0 in 1), ob čemer upošteva le značilke iz skupine G:

$$R_0(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=0}\} \text{ in}$$

$$R_1(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=1}\},$$

kjer sta $\mathbf{x}^{(G)}$ in $\mathbf{x}'^{(G)}$ primera $\mathbf{x}$ in $\mathbf{x}'$ opisana zgolj z vrednostmi značilk iz skupine G. Transformirana vrednost primera $\mathbf{x}$ za skupino G je nato s statistiko t ocenjena razlika med množicama R_0 in R_1:

$$t_G^{\text{SetSig}}(\mathbf{x}) = t\text{-statistika}(R_0(\mathbf{x}; G), R_1(\mathbf{x}; G)). \tag{5.1}$$

Če formuli za $R_0(\mathbf{x}; G)$ in $R_1(\mathbf{x}; G)$ spremenimo tako, da pri izračunu korelacij izpustita primer $\mathbf{x}$, torej v

$$R_0(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=0} \wedge \mathbf{x}' \neq \mathbf{x}\} \text{ in}$$

$$R_1(\mathbf{x}; G) = \{r(\mathbf{x}^{(G)}, \mathbf{x}'^{(G)}) : \mathbf{x}' \in \mathbf{X}^{y=1} \wedge \mathbf{x}' \neq \mathbf{x}\},$$

dobimo natanko transformacijo z metodo SetSig s skladanjem za $k = m$, če je m število primerov učne množice. Z enim pogojem, ki ga lahko implementiramo kot iskanje v razpršeni tabeli, smo se izognili zamudnemu prečnemu preverjanju. S tako predelano metodo SetSig pri skladanju, ki ga izvedemo analogno metodi "izpusti enega", porabimo enako časa kot brez skladanja.

5.4 Potek eksperimentov

5.4.1 Simulirani podatki

Glavni vir. Po vzoru podatkov o genskih izrazih smo ustvarili podatkovne nabore z $m = 50$ primeri in $n = 10000$ značilkami. Prvih 25 primerov je predstavljalo en razred, naslednjih 25 drugega. Značilke so bile dveh vrst:

- Z različnimi porazdelitvami med razredi ($N = 100$). Vrednosti za prvi razred so bile vzorčene iz normalne porazdelitve $\mathcal{N}(-d/2, 1)$ za prvi razred in $\mathcal{N}(d/2, 1)$ za drugi razred. Konstanto d, ki predstavlja razliko med razredoma, smo nastavljali na vrednosti iz intervala $[0, 1]$.
- Naključne značilke ($N = 9900$). Vrednosti značilke smo ne glede na razred vzorčili iz normalne porazdelitve $\mathcal{N}(0, 1)$.

Pri podatkih z večjo razliko med razredi d je laže odkriti razliko med značilkami z različnimi porazdelitvami med razredi in naključnimi značilkami. Za vsako vrednost d smo generirali 20 podatkovnih naborov, kjer smo vsakič uporabili drugo seme za generator naključnih števil.

Viri skupin značilk. Ustvarili smo 100 skupin značilk s po 100 geni dveh tipov:

- Skupine z značilkami z različnimi porazdelitvami med razredi ($N = 10$). Vsaka skupina vsebuje 20 značilk z različnimi porazdelitvami med razredi in 80 naključnih značilk.

- Skupine značilk brez različnih porazdelitev med razredi ($N = 90$). Vsaka skupina vsebuje 100 naključnih značilk.

5.4.2 Resnični podatki

Glavni vir. Poleg 42 naborov podatkov o človekovih genskih izrazih, ki smo jih opisali v razdelku 2.3.1 in uporabljali v prejšnjih poglavjih, smo uporabili še 19 dodatnih naborov podatkov iz baze Gene Expression Omnibus (GEO) [80]. Vsi uporabljeni nabori so našteti v tabeli 5.1.

Skupine značilk. Skupine značilk smo prebrali iz baze MSigDB [6]. Uporabili smo skupine genov iz baze MSigDB verzije 5.1, ki so jih pridobili iz baze KEGG (oznaka C2.CP.KEGG). Za primerjavo smo uporabili še večje skupine značilk izbranih iz MSigDB verzije 3.0 z oznakami C2, C5.BP in C5.MF, ki so morale vsebovati vsaj 100 genov testiranega nabora podatkov.

5.4.3 Metode transformacije

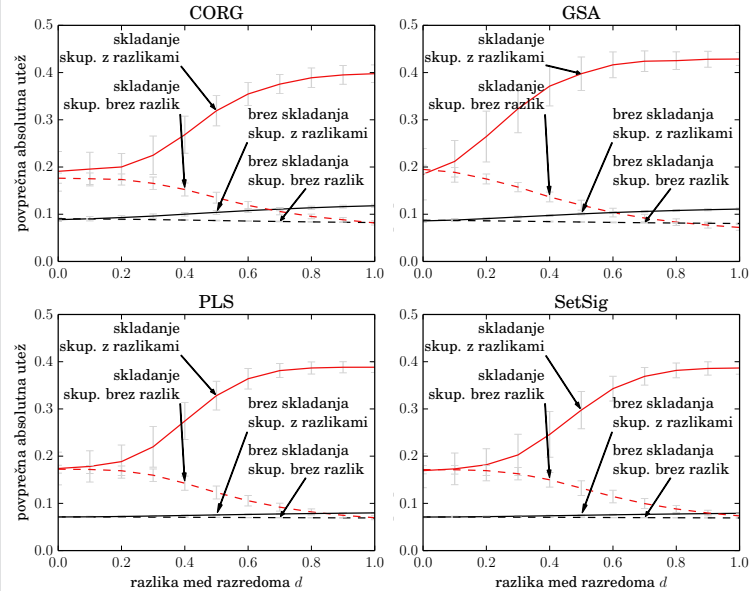
Za preizkus skladanja smo uporabili štiri metode transformacije, ki pri transformaciji uporabijo tudi informacijo o ciljni spremenljivki učnih primerov: CORG, GSA, PLS in SetSig.

5.4.4 Potek eksperimentov

Učne metode. Lotevali smo se zgolj klasifikacijskih problemov. Izvorne podatke in podatke transformirane v prostor značilk smo klasificirali z regularizirano logistično regresijo in naključnimi gozdovi, kot smo opisali v razdelku 2.3.2. Na simuliranih podatkih regularizacijskega parametra logistične regresije nismo nastavljali znotraj prečnega preverjanja, ampak smo uporabili $C = 1$.

Na resničnih podatkih smo pri transformacijskih metodah CORG, GSA, PLS skladali z delitvijo na $k = 5$ delov, metodo SetSig pa smo skladali analogno metodi "izpusti enega", torej s $k = m$, kar je zaradi implementacije iz razdelka 5.3 petkrat hitreje kot s $k = 5$, glede na preliminarne rezultate pa nismo opazili bistvenih razlik.

Pri skladanju sta končni napovedni model, ki združuje napovedi, in predstavitev značilk, ki opisujejo osnovne napovedne modele, zelo pomembna [106, 107]. Kljub temu smo v tem poglavju kot končni napovedni model uporabili isto metodo kot pri netransformiranih podatkih in podatkih transformiranih brez skladanja, da bi bila primerjava z drugimi načini napovedovanja s transformiranimi značilkami neposrednejša.



Slika 5.3

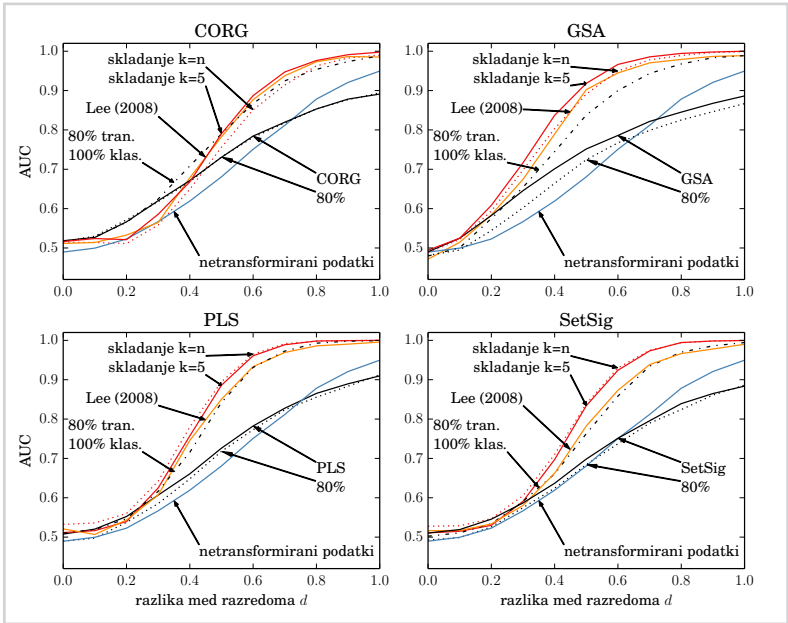
Povprečne uteži logistične regresije za obe vrsti skupin značilk: skupine z od razreda odvisnimi značilkami (polna črta) in skupine brez njih (črtkano). Pri vseh metodah transformacije opazimo, da je skladanje (rdeče črte) logistični regresiji pomagalo laže razločiti med skupinami značilk kot transformacija z uporabo vseh podatkov (črne črte).

Testiranje. Ker metoda stremena (angl. bootstrap) učno množico gradi z vzorčenjem z vračanjem, lahko učna množica vsebuje ponovljene primere. Če bi na takšni učni množici nato izvajali še skladanje, bi se v razdelitvi primerov znotraj skladanja (prečno preverjanje) primer iz učne množice lahko ponovil še v testni množici. Ker bi to favoriziralo metode, ki se preveč prilagodijo podatkom, smo testirali s 5-kratnim prečnim preverjanjem, ki smo ga pognali štirikrat. Vsak rezultat AUC, ki ga poročamo, je zato povprečje 20 rezultatov na testni množici. Na simuliranih podatkih smo uporabljali 5-kratno prečno preverjanje.

5.5 Rezultati in diskusija

5.5.1 Skladanje omogoči klasifikatorju, da oceni kvaliteto značilk

Slika 5.3 prikazuje absolutne uteži logistične regresije za obe vrsti skupin (povprečje čez 20 ponovitev generiranja podatkov). Vidimo, da se pri skladanju poveča razlika med skupinami, ki vsebujejo od razreda odvisne značilke, in tistimi, ki ga ne. Razlike se s



Slika 5.4

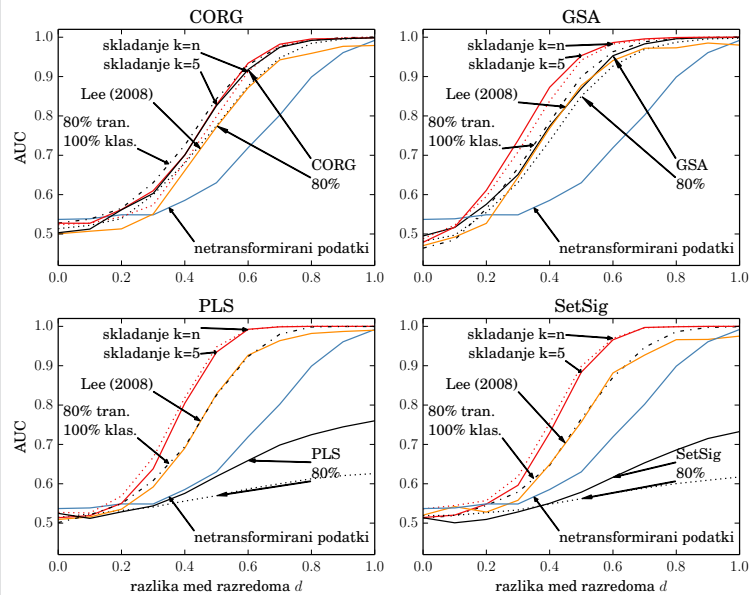
Kvaliteta končnih klasifikatorjev (AUC) v odvisnosti od razlike med povprečji razredov d. Klasifikatorje smo gradili z logistično regresijo s parametrom $C = 1$.

povečevanjem razlike med razredi d povečujejo tako pri skladanju kot pri transformaciji z uporabo vseh podatkov, a je pri skladanju razlika večja.

Kljub temu, da skupine brez od razreda odvisnih značilk ne nosijo nobene koristne informacije o razredu, logistični regresiji po transformaciji z direktno uporabo celotne učne množice ni uspelo dobro ločevati med obema vrstama skupin značilk. To kaže na pretirano uporabo skupin brez od razreda odvisnih značilk pri končni klasifikaciji. Po transformaciji s skladanjem logistična regresija oceni prave skupine kot veliko pomembnejše.

5.5.2 *Skladanje izboljša rezultate na simuliranih podatkih*

Sliki 5.4 (logistična regresija) in 5.5 (naključni gozdovi) prikazujeta, kako se kvaliteta končnih rezultatov na sintetičnih podatkih spreminja v odvisnosti od razlike med razredi d na simuliranih naborih podatkov. Pri gradnji končnih modelov z logistično regresijo (slika 5.4) opazimo, da so za majhne razlike med razredi d (d manjši od približno



Slika 5.5

Kvaliteta končnih klasifikatorjev (AUC) v odvisnosti od razlike med povprečji razredov d. Kot končne klasifikatorje smo uporabili naključne gozdove s 1000 drevesi.

0.6, odvisno od metode) klasifikatorji zgrajeni na skupinah značilk (po transformaciji) boljši od klasifikatorjev zgrajenih na originalnih značilkah. Pri d okrog 0.7 (odvisno od metode transformacije) pa opazimo, da se klasifikatorji na izvornih značilkah obnesejo bolje kot transformacija z uporabo vseh podatkov. Vendar, če transformiramo s skladanjem ali z metodo po Lee et al. [33], s transformiranimi značilkami še vedno dobimo veliko boljše rezultate kot z učenjem na netransformiranih podatkih. Skladanje se obnese še bolje kot metoda po Lee et al. Med variantami skladanja, kjer smo primere razdelili na $k = 5$ množic in kjer je bil k enak številu primerov učne množice m, nismo opazili bistvenih razlik.

Po pričakovanjih smo s transformacijo in nadaljnjim učenjem končnih klasifikatorjev na 80% učne množice vsakič dobili slabše rezultate kot s transformacijo na celotnem naboru podatkov. Če pa smo za nastavitev parametrov transformacij v prostor skupin uporabili 80% učne množice, za gradnjo odločitvenega modela pa celotno transformirano učno množico, so rezultati skoraj tako dobri kot tisti z skladanjem: že 20%

dodatnih primerov pri gradnji končnih klasifikatorjev logistični regresiji pomaga ločevati med skupinami z in brez različnih porazdelitev med razredi.

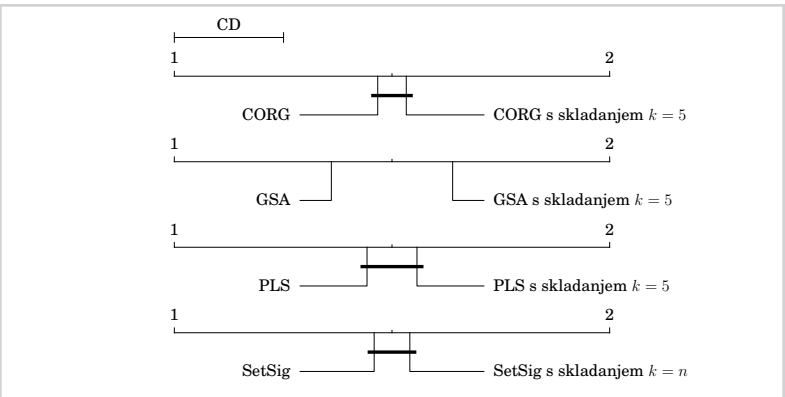
Pri vseh štirih preizkušenih metodah transformacije opazimo podobne izboljšave s skladanjem. Skladanje najmanj izboljša rezultate metode CORG, kar bi lahko razložili s tem, da metoda CORG za izračun ocene skupine uporabi preprosto statistiko (povprečje) nad majhnim številom izbranih značilk: pri skupinah brez značilk z razlikami med razredi je v povprečju uporabila 7.01 od 100 značilk v skupini. Dovolj malo izbranih značilk metodi CORG že v osnovi preprečuje preveliko prilagoditev razredni spremenljivki. Metoda CORG je bila tudi edina izmed preizkušenih transformacijskih metod, kjer je skladanje na sintetičnih podatkih za neke vrednosti d (za $d < 0.4$) poslabšalo rezultate.

Naključni gozdovi so z originalnimi značilkami dosegli boljše rezultate kot logistična regresija s $C = 1$ (slika 5.5). Pri naključnih gozdovih glede na rezultate opazimo dve skupini transformacijskih metod: v prvi sta metoda CORG in GSA, v drugi PLS in SetSig. Obe metodi iz prve skupine najprej izvedeta izbor značilk, nato pa za vsako skupino izračunata povprečne vrednosti posameznih značilk. Metoda CORG, kjer skladanje pri naključnih gozdovih ne izboljša rezultatov, je v povprečju za skupine brez razlik izbrala 7.0 od 100 značilk. Metoda GSA, kjer opazimo majhne izboljšave, izbere pozitivno ali negativno prispevajočo skupino značilk, kar pomeni, da bo na naključnih podatkih izbrala približno polovico značilk: v povprečju je izbrala 53 od 100 značilk. Pri metodah PLS in SetSig opazimo celo večje razlike med skladanjem in transformacijo z uporabo vseh podatkov kot pri logistični regresiji.

5.5.3 Rezultati na podatkih o izrazih genov

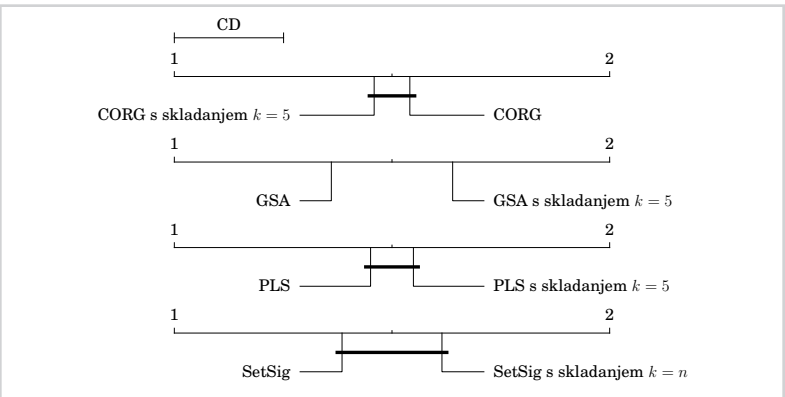
Če primerjamo transformacijo s skladanjem in transformacijo z uporabo celotne učne množice za vsako transformacijsko metodo posebej, med povprečnimi uvrstitvami AUC na 61 naborih podatkov ne opazimo bistvenih razlik (sliki 5.6 in 5.7). Le za metodo GSA so razlike značilne. Razlike smo ocenili z izboljšanim Friedmanovim in Nemenyi-vem testom [103]. Edina razlika med rezultati z logistično regresijo in naključnimi gozdovi je obrnjena uvrstitev transformacije s skladanjem in brez pri metodi CORG.

Tabeli 5.1 (logistična regresija) in 5.3 (naključni gozdovi) prikazujeta rezultate skladanja za vsak nabor podatkov posebej. Skladanje najmanj vpliva na rezultate transformacijske metode CORG pri logistični regresiji in GSA pri naključnih gozdovih.



Slika 5.6

Povprečne uvrstitve napovedne točnosti čez 61 podatkovnih naborov. Končne klasifikatorje smo gradili z logistično regresijo, kjer smo C nastavili z notranjim prečnim preverjanjem. CD označuje kritično razdaljo, znotraj katere razlike med povprečnimi uvrstitvami končnih klasifikatorjev niso značilne glede na Nemenyi-ev test (α = 0.05).



Slika 5.7

Povprečne uvrstitve napovedne točnosti čez 61 podatkovnih naborov. Končne klasifikatorje smo gradili z naključnimi gozdovi. CD označuje kritično razdaljo, znotraj katere razlike med povprečnimi uvrstitvami končnih klasifikatorjev niso značilne glede na Nemenyi-ev test (α = 0.05).

Tabela 5.1

Skladanje na resničnih podatkih z logistično regresijo in manjšimi skupinami značilk. Stolpec Orig prikazuje AUC na originalnih značilkah. Stolpci T prikazujejo razliko med AUC transformacije na celotni učni množici in AUC na originalnih podatkih, stolpci S-T pa razliko med AUC transformacije s skladanjem in AUC transformacije na celotni učni množici. Tabela je urejena po povprečni vrednosti S-T.

					CORG		GSA		PLS		SetSig		
Nabor podatkov	n	m	l	Orig	T	S-T	T	S-T	T	S-T	T	S-T	$\overline{\text{S-T}}$
GDS3929	183	19136	180	.479	+.035	+.050	−.066	+.034	+.027	+.091	−.093	+.211	+.097
GDS1887	46	9697	182	.455	−.010	+.048	−.023	−.056	−.082	+.102	−.086	+.198	+.073
GDS1390	20	14903	182	.855	−.118	+.013	−.092	+.112	−.092	+.053	−.039	−.039	+.035
GDS3952	88	31623	181	.477	+.026	+.010	−.058	+.024	−.014	+.002	−.031	+.079	+.029
GDS2547	122	10629	151	.777	−.040	+.010	−.058	+.017	−.040	+.013	−.032	+.014	+.014
GDS2609	22	34700	181	.908	+.067	0	+.025	0	+.017	+.025	+.025	+.025	+.012
GDS5037_con_sev	58	29905	181	.873	+.028	+.019	−.024	+.007	+.013	+.012	+.045	+.004	+.010
GDS3875_hea_typ1	105	16847	167	.938	−.068	−.003	−.110	+.042	−.068	+.002	−.088	−.001	+.010
GDS1667	36	34700	181	.977	+.023	0	−.009	+.010	+.013	+.009	+.013	+.009	+.007
GDS1209	54	14903	182	.989	−.013	+.002	−.006	−.002	−.017	+.006	−.017	+.011	+.004
GDS2489	44	6278	181	.980	+.020	0	+.005	−.003	+.012	+.007	+.010	+.010	+.004
GSE3726	52	14166	182	.933	+.030	−.005	−.033	+.010	+.005	+.005	+.020	+.006	+.004
leukemia	72	4680	177	.996	−.017	+.005	+.003	0	−.004	+.007	+.003	+.001	+.003
GDS4549_ph_pfi	77	26594	180	1.000	0	0	−.004	+.003	−.004	+.004	−.001	+.001	+.002
GDS4318_gc_gg	87	26594	180	.547	−.062	+.001	−.045	−.033	−.086	+.047	−.074	−.008	+.002
GDS1562	28	1394	62	.994	−.006	0	−.061	−.006	−.019	0	−.031	+.011	+.001
GDS2785	43	9697	182	.822	+.012	+.013	−.044	−.020	−.022	+.005	+.027	+.004	+.001
GDS1615_cn	101	14093	182	.998	−.005	−.001	−.028	−.002	−.008	−.002	−.008	+.005	0
GDS1210	30	6277	181	1.000	0	0	0	0	0	0	0	0	0
GDS1282	32	14903	182	1.000	0	0	0	0	0	0	0	0	0
GDS1329	43	14902	182	1.000	0	0	0	0	0	0	0	0	0
GDS1618	20	14903	182	1.000	0	0	0	0	0	0	0	0	0
GDS1650	39	9697	182	1.000	0	0	0	0	0	0	0	0	0
GDS4129	120	31595	181	1.000	0	0	0	0	0	0	0	0	0
GDS4130	104	31595	181	1.000	0	0	0	0	0	0	0	0	0
GDS1375	63	14903	182	.983	+.009	−.002	+.012	+.001	+.014	0	+.012	0	0
GDS1220	54	14903	182	.894	+.002	0	+.003	−.011	0	+.006	+.014	+.003	−.001
GDS2855_norm_juv	40	16847	167	1.000	0	−.003	−.012	0	−.006	0	0	0	−.001
GDS4274	130	31595	181	.999	−.007	+.001	−.006	−.003	−.004	0	−.003	−.002	−.001
GDS2250	38	34700	181	.990	+.006	0	0	−.008	+.003	+.003	+.010	0	−.001
GDS2520	44	9697	182	.986	−.018	+.005	−.057	+.010	−.012	−.008	+.001	−.012	−.001
GDS1615_un	68	14093	182	.992	−.027	−.002	−.013	−.018	−.027	+.003	−.019	+.009	−.002
GDS531	173	9459	181	.767	−.056	+.002	−.007	+.001	0	−.009	−.018	−.002	−.002
GSE412	110	6776	180	.955	−.035	−.020	−.059	+.002	−.036	+.001	+.016	+.001	−.004
DLBCL	77	6219	180	.982	+.012	−.005	+.001	−.021	+.006	+.004	+.012	+.003	−.005
prostata	102	9582	182	.964	+.009	−.016	−.060	−.008	−.047	−.002	−.024	+.003	−.006
GDS2201	37	14903	182	1.000	−.037	+.004	0	−.017	−.017	−.017	−.025	−.008	−.009
GDS2735	46	18022	181	.678	−.017	+.055	−.029	−.143	−.066	+.040	−.042	+.004	−.011
GDS806	60	20007	182	.701	−.037	+.041	−.028	−.011	0	−.004	+.013	−.069	−.011
GDS1615_cu	85	14093	182	.988	−.022	−.026	−.068	−.023	−.062	−.015	−.040	+.012	−.013
GDS4228	166	3779	161	.841	−.097	−.002	−.114	−.011	−.101	−.025	−.099	−.017	−.014
GDS971	23	9695	182	.871	+.071	+.017	−.004	−.042	+.096	−.008	+.129	−.025	−.015
GDS2855_norm_eme	42	16847	167	.974	−.074	−.012	−.091	−.014	−.086	−.025	−.087	−.009	−.015
GDS5037_con_mild	70	29905	181	.741	−.037	−.016	−.062	−.004	−.016	−.039	+.028	−.004	−.016
GDS2771	187	14902	182	.789	−.050	−.016	−.078	−.019	−.052	−.029	−.039	−.002	−.017
GDS2519_park_norm	72	14093	182	.620	+.075	+.070	+.020	−.046	+.037	−.048	+.039	−.070	−.024
GDS4228_agent	125	3779	161	.891	−.053	−.018	−.057	−.011	−.052	−.038	−.042	−.047	−.029
GDS1059	53	6200	180	.690	−.023	+.034	−.015	−.045	−.049	−.067	−.035	−.042	−.030
GDS4318_gc_cc	76	26594	180	.520	−.051	−.009	−.052	+.038	−.030	−.116	−.020	−.036	−.031
GDS2415	59	14345	183	.750	−.040	0	+.009	−.044	−.030	−.054	−.019	−.037	−.034
GDS2842	36	9697	182	.642	−.147	+.034	−.200	+.050	−.143	−.003	−.014	−.229	−.037
GDS2519_park_neur	83	14093	182	.640	−.084	+.024	+.007	−.083	−.041	−.052	−.036	−.058	−.042
GDS2113	75	14903	182	.645	−.129	−.003	−.115	+.023	−.076	−.062	−.047	−.138	−.045
GDS4318_gg_cc	53	26594	180	.591	−.015	−.002	−.059	−.007	−.083	−.068	−.021	−.136	−.053
GDS1062	27	14903	182	.715	+.035	−.048	+.015	−.045	+.068	−.090	+.068	−.033	−.054
GDS1714	28	16246	182	.814	−.019	−.050	−.056	−.069	−.072	−.019	−.039	−.078	−.054
GDS3268	202	29391	181	.925	−.059	−.068	−.100	−.098	−.051	−.069	−.071	−.043	−.070
GDS3874_hea_typ1	105	14093	182	.929	−.053	−.061	−.074	−.047	−.008	−.126	−.020	−.065	−.074
GDS232	23	932	93	.629	−.069	−.144	−.011	−.077	+.004	−.149	−.031	+.006	−.091
GDS1221	28	9697	182	.529	+.064	−.056	−.079	+.013	+.046	−.114	+.078	−.273	−.107
GDS2519_neur_norm	55	14093	182	.649	−.054	−.080	−.048	−.093	−.049	−.216	−.026	−.235	−.156

Tabela 5.2

Skladanje na resničnih podatkih z logistično regresijo in večjimi skupinami značilk. Stolpec Orig prikazuje AUC na originalnih značilkah. Stolpci T prikazujejo razliko med AUC transformacije na celotni učni množici in AUC na originalnih podatkih, stolpci S-T pa razliko med AUC transformacije s skladanjem in AUC transformacije na celotni učni množici. Tabela je urejena po povprečni vrednosti S-T.

					CORG		GSA		PLS		SetSig		
Nabor podatkov	n	m	l	Orig	T	S-T	T	S-T	T	S-T	T	S-T	$\overline{\text{S-T}}$
GDS1887	46	9697	238	.455	−.055	+.067	+.004	−.039	−.053	+.217	−.151	+.410	+.164
GDS3929	183	19136	304	.479	−.026	+.059	−.110	+.065	−.078	+.110	−.086	+.148	+.096
GDS2735	46	18022	294	.678	−.034	+.072	−.134	+.001	−.154	+.107	−.187	+.158	+.084
GDS971	23	9695	238	.871	+.050	+.025	−.169	+.171	+.021	+.004	+.117	+.012	+.053
GDS2489	44	6278	162	.980	+.015	+.005	−.003	−.003	−.055	+.050	−.020	+.033	+.021
GDS2609	22	34700	306	.908	+.079	+.012	−.042	−.013	+.004	+.025	+.029	+.037	+.016
GDS806	60	20007	281	.701	−.003	−.040	−.021	−.013	−.003	+.034	−.010	+.081	+.016
GDS3952	88	31623	308	.477	+.053	−.022	−.055	+.055	−.002	+.001	−.004	+.005	+.010
GDS2520	44	9697	238	.986	−.006	0	−.067	+.010	−.033	+.007	−.018	+.008	+.006
GDS5037_con_sev	58	29905	312	.873	+.038	+.004	−.025	−.016	+.005	+.019	+.011	+.008	+.004
leukemia	72	4680	122	.996	−.008	−.002	+.002	+.002	−.012	+.012	−.001	+.003	+.004
prostata	102	9582	233	.964	−.028	+.013	−.102	−.003	−.083	+.005	−.021	−.002	+.003
GDS2201	37	14903	284	1.000	0	−.008	−.012	+.008	−.008	+.003	−.008	+.008	+.003
GDS2785	43	9697	238	.822	+.042	+.011	−.029	−.006	−.005	−.003	+.042	+.007	+.002
GDS4228	166	3779	66	.841	−.099	+.008	−.133	+.010	−.104	−.017	−.098	+.007	+.002
GDS1562	28	1394	6	.994	−.022	0	−.033	0	−.042	0	−.047	+.008	+.002
GDS1209	54	14903	284	.989	−.015	0	−.008	+.004	−.017	+.002	−.010	+.002	+.002
GDS1615_cn	101	14093	287	.998	−.004	−.001	−.025	−.004	−.005	+.002	−.015	+.009	+.002
GDS2855_norm_juv	40	16847	79	1.000	0	0	−.003	0	−.003	+.003	−.003	+.003	+.002
GDS1615_un	68	14093	287	.992	−.018	−.004	−.033	−.005	−.008	+.002	−.032	+.012	+.001
GDS4549_ph_pfi	77	26594	304	1.000	0	0	−.003	+.003	−.001	+.001	0	0	+.001
GDS1375	63	14903	284	.983	+.007	+.003	+.011	0	+.014	0	+.014	0	+.001
GDS1220	54	14903	284	.894	+.014	−.008	0	0	−.003	+.009	+.019	0	0
GDS1210	30	6277	162	1.000	0	0	0	0	0	0	0	0	0
GDS1329	43	14902	284	1.000	0	0	0	0	0	0	0	0	0
GDS1618	20	14903	284	1.000	0	0	0	0	0	0	0	0	0
GDS1650	39	9697	238	1.000	0	0	0	0	0	0	0	0	0
GDS4129	120	31595	307	1.000	0	0	0	0	0	0	0	0	0
GDS4130	104	31595	307	1.000	0	0	0	0	0	0	0	0	0
GSE412	110	6776	159	.955	−.071	−.008	−.094	+.003	−.088	+.007	+.024	−.002	0
GDS2250	38	34700	306	.990	+.003	+.007	−.004	−.011	+.003	+.003	+.010	0	0
GDS1282	32	14903	284	1.000	0	0	0	−.006	0	0	0	0	−.001
GDS4228_agent	125	3779	66	.891	−.089	0	−.109	−.015	−.101	−.008	−.102	+.004	−.005
GDS4274	130	31595	307	.999	−.001	−.002	−.008	−.021	−.004	−.002	−.009	−.003	−.007
GDS531	173	9459	247	.767	−.034	−.014	−.016	−.008	−.001	+.001	−.004	−.007	−.007
GDS3875_hea_typ1	105	16847	79	.938	−.075	−.006	−.134	−.020	−.045	−.030	−.082	+.027	−.007
GSE3726	52	14166	272	.933	−.025	+.030	−.069	−.032	−.034	−.021	+.022	−.008	−.008
DLBCL	77	6219	158	.982	−.032	−.010	−.007	−.018	+.003	−.003	+.009	+.001	−.008
GDS1667	36	34700	306	.977	+.023	0	+.013	−.041	+.018	+.005	+.018	+.004	−.008
GDS2415	59	14345	195	.750	−.029	−.020	−.014	+.019	−.032	−.024	−.030	−.011	−.009
GDS1390	20	14903	284	.855	−.026	−.092	−.079	+.053	−.053	+.013	−.039	−.013	−.010
GDS2547	122	10629	35	.777	−.067	−.019	−.049	−.024	−.017	−.013	−.013	−.004	−.015
GDS2855_norm_eme	42	16847	79	.974	−.062	−.047	−.076	+.012	−.050	−.014	−.028	−.022	−.018
GDS5037_con_mild	70	29905	312	.741	−.021	−.036	−.055	−.047	+.005	−.021	+.039	−.001	−.027
GDS3874_hea_typ1	105	14093	287	.929	−.072	−.020	−.036	−.038	−.066	−.020	−.030	−.041	−.030
GDS2771	187	14902	284	.789	−.024	−.056	−.032	−.040	−.027	−.042	−.015	−.024	−.041
GDS2519_park_norm	72	14093	287	.620	+.030	−.008	−.049	−.062	+.001	−.054	−.027	−.059	−.046
GDS4318_gg_cc	53	26594	304	.591	−.020	−.119	−.033	0	−.028	−.059	+.005	−.009	−.046
GDS232	23	932	11	.629	−.033	−.064	+.064	−.026	−.055	−.174	−.162	+.076	−.047
GDS4318_gc_gg	87	26594	304	.547	−.035	−.068	−.027	−.056	−.032	−.043	−.003	−.025	−.048
GDS1062	27	14903	284	.715	+.005	−.066	−.007	+.002	−.030	+.011	+.061	−.144	−.049
GDS1615_cu	85	14093	287	.988	−.012	−.050	−.038	−.066	−.032	−.069	−.044	−.014	−.050
GDS1059	53	6200	145	.690	−.001	−.050	+.011	−.014	−.016	−.116	−.009	−.025	−.051
GDS1714	28	16246	266	.814	+.008	−.003	−.131	−.039	−.058	−.086	−.044	−.092	−.055
GDS1221	28	9697	238	.529	+.053	+.111	+.097	−.110	+.062	−.101	+.070	−.135	−.059
GDS3268	202	29391	310	.925	−.055	−.090	−.086	−.106	−.060	−.035	−.124	−.019	−.063
GDS4318_gc_cc	76	26594	304	.520	+.031	−.009	+.103	+.009	+.054	−.140	+.047	−.117	−.064
GDS2842	36	9697	238	.642	−.053	−.067	−.119	−.005	−.102	−.034	−.011	−.200	−.077
GDS2519_park_neur	83	14093	287	.640	−.076	−.055	+.018	−.078	−.008	−.072	+.025	−.103	−.077
GDS2113	75	14903	284	.645	−.122	−.098	−.063	−.047	−.069	−.115	−.088	−.159	−.105
GDS2519_neur_norm	55	14093	287	.649	−.115	−.119	−.033	−.051	−.015	−.237	+.019	−.192	−.150

Tabela 5.3

Skladanje na resničnih podatkih z naključnimi gozdovi in manjšimi skupinami značilk. Stolpec Orig prikazuje AUC na originalnih značilkah. Stolpci T prikazujejo razliko med AUC transformacije na celotni učni množici in AUC na originalnih podatkih, stolpci S-T pa razliko med AUC transformacije s skladanjem in AUC transformacije na celotni učni množici. Tabela je urejena po povprečni vrednosti S-T.

					CORG		GSA		PLS		SetSig		
Nabor podatkov	n	m	l	Orig	T	S-T	T	S-T	T	S-T	T	S-T	$\overline{\text{S-T}}$
GDS1887	46	9697	182	.415	+.025	+.133	+.064	+.068	−.017	+.174	−.052	+.233	+.152
GDS3929	183	19136	180	.480	+.015	+.047	−.007	+.010	−.020	+.074	−.055	+.143	+.068
GDS2735	46	18022	181	.593	−.077	+.146	−.165	−.083	−.139	+.081	−.106	+.125	+.067
GDS806	60	20007	182	.770	−.040	+.038	−.054	+.003	−.067	+.067	−.048	+.070	+.045
GDS3952	88	31623	181	.485	+.023	+.018	−.025	+.007	+.009	−.009	−.019	+.075	+.023
GDS2609	22	34700	181	.979	−.017	+.021	−.042	0	−.067	+.025	−.046	+.025	+.018
GDS3875_hea_typ1	105	16847	167	.886	−.027	+.017	−.055	+.008	−.047	+.030	−.036	+.003	+.014
GDS1390	20	14903	182	.868	−.079	+.026	−.063	+.010	−.066	+.013	−.039	0	+.012
GDS2785	43	9697	182	.850	−.027	+.037	−.107	+.005	−.065	+.038	−.001	−.043	+.009
GDS1667	36	34700	181	.996	+.004	0	−.024	+.002	−.022	+.018	−.018	+.013	+.008
GDS2855_norm_eme	42	16847	167	.917	−.037	+.021	−.032	+.009	−.048	+.009	−.017	−.010	+.007
GDS2489	44	6278	181	1.000	0	0	−.005	0	−.007	+.007	−.010	+.010	+.004
leukemia	72	4680	177	.998	−.001	+.001	−.004	0	−.007	+.006	−.013	+.009	+.004
GDS2250	38	34700	181	1.000	−.008	+.004	−.040	+.005	−.004	+.004	0	0	+.003
GDS1209	54	14903	182	.996	−.008	+.002	0	0	−.019	+.011	−.023	−.001	+.003
GDS1220	54	14903	182	.928	−.007	+.010	−.045	−.007	−.038	+.012	−.015	−.003	+.003
GDS1615_cn	101	14093	182	.983	+.004	0	−.043	−.003	−.028	0	−.021	+.008	+.001
GDS2520	44	9697	182	.982	−.026	+.003	−.046	+.004	−.018	+.003	−.007	−.008	+.001
GDS4549_ph_pfi	77	26594	180	1.000	0	0	−.043	−.007	−.028	+.004	−.017	+.005	0
GDS4274	130	31595	181	.998	−.002	0	−.004	−.003	−.004	+.002	−.006	+.002	0
GDS1210	30	6277	181	1.000	0	0	0	0	0	0	0	0	0
GDS1282	32	14903	182	1.000	0	0	0	0	0	0	0	0	0
GDS1329	43	14902	182	1.000	0	0	0	0	0	0	0	0	0
GDS1618	20	14903	182	1.000	0	0	0	0	0	0	0	0	0
GDS1650	39	9697	182	1.000	0	0	0	0	0	0	0	0	0
GDS4129	120	31595	181	1.000	0	0	0	0	0	0	0	0	0
GDS4130	104	31595	181	1.000	0	0	0	0	0	0	0	0	0
GDS1615_un	68	14093	182	.988	−.024	−.005	−.070	−.001	−.051	+.006	−.036	−.001	0
GDS1375	63	14903	182	.997	−.001	0	0	−.002	0	0	−.003	0	−.001
GDS2855_norm_juv	40	16847	167	1.000	0	0	−.006	−.003	−.006	0	−.003	0	−.001
GDS971	23	9695	182	.988	−.013	−.004	−.123	+.003	−.067	−.008	0	0	−.002
GSE3726	52	14166	182	.971	+.005	0	−.090	−.005	−.058	−.012	−.017	+.005	−.003
GDS1562	28	1394	62	.975	−.017	+.003	−.042	−.027	−.017	0	−.014	+.008	−.004
DLBCL	77	6219	180	.961	+.013	−.011	−.022	−.016	+.008	+.005	+.027	+.004	−.005
GSE412	110	6776	180	.915	−.012	+.002	−.081	−.001	−.058	−.015	+.030	−.008	−.005
GDS5037_con_sev	58	29905	181	.928	−.007	+.004	−.021	−.019	−.026	0	−.016	−.016	−.007
GDS2547	122	10629	151	.773	−.037	−.011	−.046	−.008	−.027	−.010	−.022	−.005	−.009
prostata	102	9582	182	.939	+.004	−.003	−.057	−.014	−.064	−.012	−.024	−.007	−.009
GDS1615_cu	85	14093	182	.903	−.017	−.021	−.146	−.020	−.084	+.003	−.027	−.001	−.010
GDS1059	53	6200	180	.667	+.012	+.018	+.036	−.033	−.011	−.015	−.003	−.010	−.010
GDS1062	27	14903	182	.783	−.090	+.056	−.013	+.008	+.019	−.064	0	−.048	−.012
GDS2201	37	14903	182	1.000	−.021	−.054	−.050	−.002	−.037	0	−.033	+.008	−.012
GDS4228	166	3779	161	.726	+.008	+.001	+.017	−.009	0	−.014	+.029	−.028	−.013
GDS5037_con_mild	70	29905	181	.779	−.038	−.004	−.042	+.014	−.029	−.024	+.001	−.051	−.016
GDS3874_hea_typ1	105	14093	182	.888	−.064	−.024	−.118	−.004	−.071	−.023	−.061	−.019	−.017
GDS4228_agent	125	3779	161	.844	−.009	−.032	−.006	−.016	−.019	−.023	−.017	−.014	−.021
GDS2415	59	14345	183	.712	+.029	−.007	+.032	−.031	−.014	−.028	−.007	−.032	−.024
GDS531	173	9459	181	.745	−.044	−.017	−.008	+.002	+.002	−.028	−.006	−.058	−.025
GDS3268	202	29391	181	.836	−.030	−.024	−.070	−.029	−.038	−.043	−.036	−.016	−.028
GDS2771	187	14902	182	.740	−.007	−.024	−.042	−.009	+.001	−.037	+.027	−.043	−.028
GDS1221	28	9697	182	.485	+.039	−.038	−.040	+.014	−.008	−.027	+.052	−.079	−.033
GDS2519_park_neur	83	14093	182	.626	−.043	+.002	+.018	−.050	0	−.062	−.012	−.023	−.033
GDS2519_park_norm	72	14093	182	.560	+.123	+.028	+.059	−.034	+.073	−.042	+.112	−.104	−.038
GDS1714	28	16246	182	.822	+.014	−.028	−.075	−.072	−.064	−.053	−.053	−.031	−.046
GDS4318_gg_cc	53	26594	180	.552	−.008	+.029	+.033	−.044	−.008	−.120	+.009	−.105	−.060
GDS4318_gc_gg	87	26594	180	.555	−.028	−.084	−.039	−.044	−.044	−.086	−.063	−.064	−.070
GDS2113	75	14903	182	.590	−.038	−.075	−.025	−.037	−.033	−.064	−.012	−.137	−.078
GDS4318_gc_cc	76	26594	180	.447	+.057	−.091	+.047	+.020	+.067	−.127	+.127	−.122	−.080
GDS232	23	932	93	.526	+.114	−.274	+.095	−.100	+.132	−.112	+.048	+.056	−.107
GDS2519_neur_norm	55	14093	182	.544	+.024	−.064	+.085	−.028	+.032	−.146	+.048	−.203	−.110
GDS2842	36	9697	182	.582	−.042	+.034	−.066	−.025	−.040	−.174	+.041	−.279	−.111

Tabela 5.4

Skladanje na resničnih podatkih z naključnimi gozdovi in večjimi skupinami značilk. Stolpec Orig prikazuje AUC na originalnih značilkah. Stolpci T prikazujejo razliko med AUC transformacije na celotni učni množici in AUC na originalnih podatkih, stolpci S-T pa razliko med AUC transformacije s skladanjem in AUC transformacije na celotni učni množici. Tabela je urejena po povprečni vrednosti S-T.

					CORG		GSA		PLS		SetSig		
Nabor podatkov	n	m	l	Orig	T	S-T	T	S-T	T	S-T	T	S-T	$\overline{\text{S-T}}$
GDS1887	46	9697	238	.415	−.005	+.068	+.137	−.019	+.005	+.205	−.098	+.301	+.139
GDS806	60	20007	281	.770	−.044	+.065	−.097	+.057	−.054	+.129	−.051	+.096	+.087
GDS2735	46	18022	294	.593	−.024	+.090	−.156	−.060	−.177	+.109	−.180	+.136	+.069
GDS2489	44	6278	162	1.000	−.010	+.010	−.094	+.033	−.174	+.117	−.110	+.068	+.057
GDS3929	183	19136	304	.480	−.036	+.117	−.060	−.004	−.050	+.006	−.063	+.067	+.046
GDS1221	28	9697	238	.485	+.066	+.009	−.051	+.032	−.006	+.046	+.014	+.002	+.022
GDS1667	36	34700	306	.996	+.004	0	−.070	+.005	−.057	+.048	−.032	+.032	+.021
GDS971	23	9695	238	.988	−.025	0	−.279	+.162	−.079	−.080	0	0	+.021
GDS2520	44	9697	238	.982	−.004	+.012	−.062	+.014	−.036	+.015	−.022	+.001	+.011
GDS1220	54	14903	284	.928	−.008	+.017	−.024	+.007	−.034	+.015	+.006	−.010	+.007
GDS1209	54	14903	284	.996	−.006	+.004	+.004	−.002	−.021	+.013	−.044	+.012	+.007
GDS4549_ph_pfi	77	26594	304	1.000	0	0	−.038	+.002	−.034	+.009	−.009	+.006	+.004
leukemia	72	4680	122	.998	−.004	−.005	−.042	−.003	−.024	+.008	−.013	+.010	+.002
GDS2855_norm_juv	40	16847	79	1.000	0	0	−.012	+.009	−.003	0	−.003	0	+.002
GDS2609	22	34700	306	.979	+.021	−.012	−.092	0	−.067	0	−.062	+.021	+.002
GDS2201	37	14903	284	1.000	−.005	−.003	−.058	−.008	−.021	+.008	−.017	+.008	+.001
GDS4130	104	31595	307	1.000	0	0	−.001	+.001	0	0	0	0	0
DLBCL	77	6219	158	.961	−.020	−.029	−.200	+.010	−.028	+.014	+.020	+.006	0
GDS1210	30	6277	162	1.000	0	0	−.006	0	0	0	0	0	0
GDS1329	43	14902	284	1.000	0	0	0	0	0	0	0	0	0
GDS1375	63	14903	284	.997	−.001	0	+.003	0	0	0	−.001	0	0
GDS1618	20	14903	284	1.000	0	0	−.025	0	0	0	0	0	0
GDS1650	39	9697	238	1.000	0	0	0	0	0	0	0	0	0
GDS4129	120	31595	307	1.000	0	0	0	0	0	0	0	0	0
GDS1615_un	68	14093	287	.988	−.011	−.002	−.066	−.008	−.043	0	−.038	+.009	0
GDS1615_cn	101	14093	287	.983	+.009	+.001	−.046	−.004	−.023	+.001	−.021	+.001	0
GDS2250	38	34700	306	1.000	0	0	−.062	−.018	−.007	+.007	0	0	−.003
GDS5037_con_sev	58	29905	312	.928	+.023	+.002	−.092	−.007	−.031	−.008	−.028	−.002	−.004
GDS1282	32	14903	284	1.000	0	0	0	−.017	0	0	0	0	−.004
GDS1562	28	1394	6	.975	−.011	−.014	−.025	−.006	−.026	0	−.029	+.001	−.005
GDS4228	166	3779	66	.726	+.007	+.025	−.023	−.003	+.017	−.033	+.021	−.009	−.005
GDS2855_norm_eme	42	16847	79	.917	−.006	−.017	−.037	+.006	−.013	−.009	−.005	−.005	−.006
GSE412	110	6776	159	.915	+.006	−.019	−.114	+.005	−.055	−.032	+.022	+.017	−.007
GSE3726	52	14166	272	.971	+.011	−.010	−.243	−.007	−.164	−.025	−.021	+.007	−.009
GDS3875_hea_typ1	105	16847	79	.886	−.045	−.025	−.055	−.014	−.046	+.008	−.007	−.015	−.012
GDS4274	130	31595	307	.998	−.005	−.001	−.008	−.049	−.008	0	−.007	+.003	−.012
GDS4318_gc_cc	76	26594	304	.447	+.135	−.064	+.085	+.099	+.104	−.008	+.162	−.077	−.012
GDS1615_cu	85	14093	287	.903	+.024	−.032	−.129	−.018	−.099	−.004	−.010	−.006	−.015
GDS2785	43	9697	238	.850	+.015	−.013	−.086	−.014	−.056	+.030	−.023	−.069	−.016
GDS1390	20	14903	284	.868	−.053	−.052	−.118	+.039	−.026	−.026	−.026	−.030	−.017
GDS3952	88	31623	308	.485	+.055	−.034	+.010	−.032	+.047	−.050	+.001	+.043	−.019
GDS2547	122	10629	35	.773	−.066	−.039	−.042	−.042	−.026	−.014	−.015	−.008	−.026
prostata	102	9582	233	.939	0	−.005	−.075	−.017	−.079	−.024	−.008	−.062	−.027
GDS2771	187	14902	284	.740	−.013	−.021	−.053	−.013	−.005	−.033	+.006	−.047	−.029
GDS531	173	9459	247	.745	−.007	−.063	−.013	−.030	0	+.005	+.019	−.031	−.030
GDS3268	202	29391	310	.836	−.023	−.022	−.082	−.031	−.083	−.048	−.081	−.027	−.032
GDS5037_con_mild	70	29905	312	.779	−.006	−.027	−.046	−.024	−.018	−.047	+.013	−.039	−.034
GDS4228_agent	125	3779	66	.844	−.031	−.074	−.060	−.016	−.045	−.032	−.026	−.027	−.037
GDS2415	59	14345	195	.712	−.003	−.007	+.017	−.014	+.015	−.052	+.017	−.087	−.040
GDS3874_hea_typ1	105	14093	287	.888	−.060	−.020	−.084	−.058	−.076	−.029	−.017	−.060	−.042
GDS1714	28	16246	266	.822	−.008	−.039	−.150	−.028	−.053	−.072	−.028	−.033	−.043
GDS1062	27	14903	284	.783	−.092	−.019	−.055	−.038	−.056	−.034	−.012	−.092	−.046
GDS2519_park_neur	83	14093	287	.626	−.054	−.020	−.019	0	−.003	−.068	+.022	−.118	−.052
GDS232	23	932	11	.526	+.039	−.024	+.145	−.083	+.110	−.194	−.066	+.059	−.060
GDS2519_neur_norm	55	14093	287	.544	−.037	−.081	+.068	+.014	+.084	−.132	+.098	−.125	−.081
GDS2519_park_norm	72	14093	287	.560	+.074	−.084	+.012	−.064	+.042	−.147	+.034	−.048	−.086
GDS2113	75	14903	284	.590	−.027	−.122	−.015	−.059	−.019	−.050	−.026	−.141	−.093
GDS1059	53	6200	145	.667	+.024	−.035	+.066	−.102	+.018	−.131	+.033	−.131	−.100
GDS4318_gg_cc	53	26594	304	.552	−.003	−.095	+.046	−.050	+.041	−.131	+.101	−.135	−.103
GDS4318_gc_gg	87	26594	304	.555	−.039	−.042	+.011	−.083	+.011	−.162	+.062	−.138	−.106
GDS2842	36	9697	238	.582	+.034	−.105	−.029	−.093	−.014	−.121	+.042	−.275	−.149

Povprečne absolutne razlike med transformacijo s skladanjem in transformacijo na celotni učni množici so pri logistični regresiji 0.018 za CORG, 0.025 za GSA, 0.032 za PLS in 0.039 za SetSig, pri naključnih gozdovih pa 0.026 za CORG, 0.016 za GSA, 0.032 za PLS in 0.039 za SetSig. Majhne razlike pri metodi CORG smo pričakovali, saj metoda CORG izbira majhne množice podznačilk, ki jih uporabi za transformacijo v vrednost skupine. Tudi metoda GSA uporablja podmnožico skupine: glede na rezultate v 5.5.1 v povprečju uporablja polovico značilk skupine. Največje razlike, tako pozitivne kot negativne, opazimo pri metodi SetSig. Opazne izboljšave s skladanjem (razlika AUC večja od +.05) opazimo pri klasifikaciji z logistično regresijo le na dveh naborih podatkov, GDS3929 in GDS1887, pri klasifikaciji z naključnimi gozdovi pa še pri GDS2735.

Skladanje smo preizkusili še na skupinah iz MSigDB z oznakami C2.CP, C5.BP in C5.BP, a le takimi, ki so vsebovale vsaj 100 genov iz uporabljenih primarnih podatkov. Tam se je skladanje obneslo bolje. Pri logistični regresiji (tabela 5.2) so na dveh podatkovnih naborih, kjer je skladanje najbolj izboljšalo točnost z genskimi skupinami iz C2.CP.KEGG, razlike še večje. Pri naključnih gozdovih (tabela 5.4) se je število naborov s spremembo večjo od +.05 povečalo iz 3 na 4 – dva od le-teh sta ista. Že Liu et al. [31] poročajo, da so pri prečnem preverjanju posameznih skupin ugotovili, da so filtrirali več večjih kot manjših skupin.

Iz rezultatov vidimo, da skladanje pri nekaterih naborih podatkov lahko pomaga, a žal ne znamo vnaprej napovedati, na katerih podatkih bo v resnici pomagalo. Podobno tudi idealne metode za gradnjo napovednih modelov ne poznamo [108] in če narave podatkov ne poznamo dobro, nam ne preostane drugega, kot da preizkusimo več metod in med njimi izberemo najustreznejšo. Zato predlagamo, da uporabniki za posamezni nabor podatkov s prečnim preverjanem preizkusijo, ali se skladanje obnese ali ne.

5.6 *Skladanje ocen zanesljivosti*

Skladanje smo kasneje uporabili še na drugi domeni v sodelovanju s farmacevtskim podjetjem AstraZeneca. Tam prav tako nismo skladali metod za napovedovanje ciljne spremenljivke, kot se skladanje tipično uporablja, ampak smo skladali različne metode za ocenjevanje zanesljivosti napovedi [109].

Eden od načinov za določanje toksičnosti kemikalij z manj testiranja na živih organizmih je uporaba tako imenovanih metod za določanje kvantitativnih relacij med

strukturo in aktivnostjo učinkovin (angl. quantitative structure–activity relationships, QSAR). Metode QSAR uporabljajo za napovedovanje lastnosti kemikalij zgolj na podlagi njihove strukture. Pomemben aspekt metod QSAR je zaradi pestrosti kemikalij določitev domene, na kateri nek model deluje dobro, saj bo ponavadi model za kemikalije podobne tistim v učni množici vračal dobre rezultate, medtem ko tega za nabor kemikalij, ki je zelo različen od teh v učnih primerih, ne moremo pričakovati [110]. Za aktualne pristope QSAR, ki temeljijo na strojnem učenju, torej na metodah kot so naključni gozdovi ali metoda podpornih vektorjev, lahko z metodami za ocenjevanje zanesljivosti napovedi ocenimo, če bo model na nekem primeru dobro deloval. V sklopu raziskave smo primerjali znane metode za napovedovanje zanesljivosti, njihovo skladanje ter uporabo le najboljše metode glede na prečno preverjanje [111].

Metode za ocenjevanje zanesljivosti V študiji smo uporabili pristope, ki temeljijo na metodi najbližjih sosedov, kot jih uporabljajo v modeliranju QSAR [112, 113] in pristope, ki izhajajo iz strojnega učenja [114]. Vse metode zanesljivosti na vhodu potrebujejo učne podatke, regresijsko metodo za napovedovanje ter primere, za katere napovedujejo zanesljivost.

Meri na podlagi razdalj uporabljata Mahalanobisovo razdaljo, ki normalizira učne podatke z inverzom kovariančne matrike in je posplošitev evklidske razdalje. Seštevek razdalj testnega primera do k najbližjih sosedov [113] smo označili z `MN`, razdaljo do centra učne množice pa z `MC`.

Analiza občutljivosti oceni spremembe v napovedih, ki bi jih povzročile majhne spremembe učnih podatkov. Za primer, ki mu napovedujemo zanesljivost, v učno množico dodamo enak učni primer z vrednostjo napovedi ciljne spremenljivke na učnih podatkih, ki smo jo povečali ali zmanjšali za konstanto iz nabora možnih vrednosti. Spremembe lahko ocenimo na tri načine: `Var` (povprečje razlik med napovedmi na spremenjenih učnih množicah, kjer smo vrednost razreda za isto konstanto povečali ali zmanjšali), `BiasSign` (povprečje razlik med napovedmi na množici z dodanim primerom z za isto konstanto povečano ali zmanjšano vrednostjo ciljne spremenljivke in napovedmi na originalni učni množici) in `BiasAbs` (absolutna vrednost `BiasSign`) [114].

Lokalno prečno preverjanje (angl. local cross validation, oznaka `LCV`) izračuna z razdaljami med primeri uteženo povprečje napak napovedi k najbližjih sosedov [114].

Varianca napovedi (angl. bootstrap variance, oznaka `Bootstrap`) izračuna varianco napovedi modelov, ki bi jih zgradili na učnih podatkih vzorčenih z metodo stremenca

(angl. bootstrap) [114].

Lokalno modeliranje napake napovedi (angl. local prediction error modeling) je povprečna razlika napovedi testnega primera in napovedi k najbližjih primerov učne množice [114]. Za to mero smo uporabili oznaki LocalSign in LocalAbs (absolutna vrednost LocalSign).

Uporabimo tudi povprečje metod variance napovedi in lokalnega modeliranja napake napovedi (oznaka Boot+Local), saj sta z njim Bosnić in Kononenko [114] dosegla dobre rezultate.

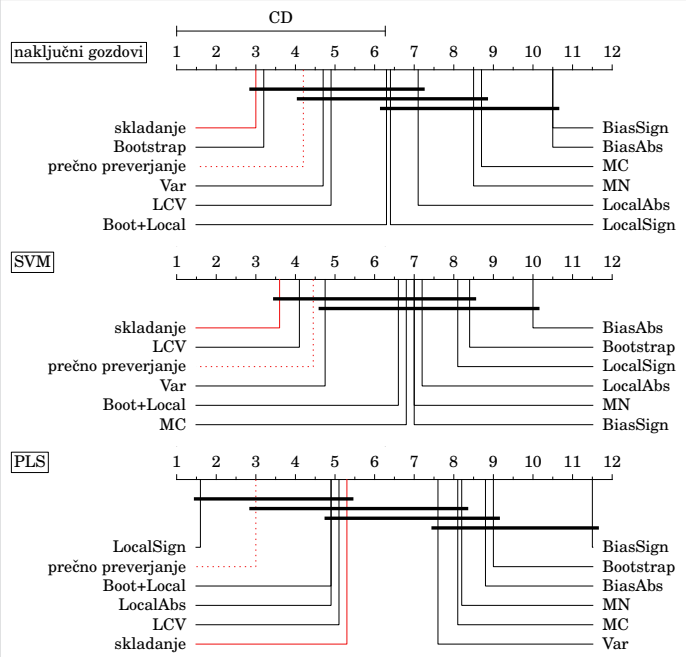
Združevanje metod Za združevanje metod smo uporabili izbiro najboljše metode s prečnim preverjanjem [111] in skladanje (razdelek 5.2.3). Oba načina združevanja uporabljata prečno preverjanje, le da skladanje iz rezultatov prečnega preverjanja še zgradi model, ki združuje rezultate več metod. Pri gradnji modela za združevanje smo za ciljno spremenljivko uporabili razliko med napovedanimi in pravimi vrednostmi ciljne spremenljivke učne množice.

Podatki Uporabili smo deset javno dostopnih regresijskih podatkovnih naborov o sorodnih kemikalijah z oznakami DHFR, COX2, BZR, h-PTP, AMPH1, EDC, ACE, HIVPR, AChE in HIVRT [115]. Nabori podatkov so vsebovali med 101 in 397 učnih primerov. Podatki so bili opisani z značilkami, ki jih vrne orodje RDKit[1]. Učni primeri so bili opisani s približno 300 značilkami.

Regresijske metode V poskusih smo uporabili tri regresijske metode, ki jih pri modeliranju QSAR pogosto uporabljajo: metodo delnih najmanjših kvadratov (angl. partial least squares, PLS) [74], naključne gozdove s 100 drevesi [41] in metodo podpornih vektorjev (SVM) z radialnim jedrom iz knjižnice LibSVM [116]. Parametre C, ν in γ pri SVM smo nastavili s prečnim preverjanjem za vsak nabor podatkov posebej.

Testiranje Kvaliteto metode za oceno zanesljivosti na posameznem naboru podatkov smo ocenili s povprečnim Pearsonovim koeficientom korelacije med ocenami zanesljivosti in razliko med napovedano in pravo vrednostjo ciljne spremenljivke v 10-kratnem prečnem preverjanju. Za metodi z oznakama BiasSign in LocalSign, ki edini vračata tudi smer ocene napake, smo uporabili dejansko razliko med napovedano in pravo vrednostjo, za ostale pa absolutno razliko.

[1] http://www.rdkit.org/



Slika 5.8

Povprečna uvrstitev metod za določanje zanesljivosti napovedi čez 10 podatkovnih naborov o sorodnih kemikalijah. CD označuje kritično razdaljo, znotraj katere razlike med metodami niso značilne glede na Nemenyi-ev test ($\alpha = 0.05$).

Rezultati Povprečno uvrstitev primerjanih metod glede na kvaliteto ocen zanesljivosti prikazujemo na sliki 5.8. Pri gradnji napovednih modelov z naključnimi gozdovi in metodo podpornih vektorjev je skladanje doseglo najboljši rezultat, a razlike v primerjavi z izbiro najboljše metode za oceno zanesljivosti s prečnim preverjanjem niso značilne (Nemenyi test [103], $\alpha = 0.05$). Za modele s PLS se je skladanje obneslo slabše kot izbira najboljše metode s prečnim preverjanjem, a tudi tam razlike niso značilne. Možen vzrok za slabši rezultat skladanja s PLS je velika razlika med najbolje uvrščeno metodo in preostalimi metodami. Tudi Džeroski in Ženko [106] sta na več podatkovnih naborih primerjala različne metode skladanja manjšega števila napovednih metod in nista odkrila značilnih razlik med izbiro najboljše osnovne metode s prečnim preverjanjem in večino pristopov skladanja.

V tem razdelku smo uspešno uporabili skladanje, kjer prav tako nismo skladali napovedi ciljne spremenljivke, ampak smo v nasprotju s preostankom poglavja namesto

transformacij v prostor skupin skladali ocene zanesljivosti modelov. Namesto nekaj sto oziroma nekaj tisoč transformacij, po eno za vsako skupno značilk, smo skladali le 10 transformacijskih metod.

5.7 Zaključek

V pričujočem poglavju smo za transformacijo značilk v vrednosti, ki opisujejo skupine, predlagali skladanje. S skladanjem se izognemo pretirani prilagoditvi parametrov transformacij, ki uporabljajo vrednosti ciljne spremenljivke, učnim podatkom. Postopek skladanja lahko uporabimo s katerokoli obstoječo metodo in to tako za klasifikacijo kot za regresijo.

Na sintetičnih podatkovnih naborih smo pokazali, da skladanje lahko omogoči odločitvenim modelom bolje oceniti kvaliteto značilk in da lahko izboljša rezultate. Na resničnih naborih podatkov, kjer smo skladanje preizkusili, med skladanjem in transformacijo s celotno učno množico v povprečju ne opazimo značilnih razlik, smo pa na nekaterih naborih podatkov opazili velike razlike. Ker skladanje transformacijskih metod na nekaterih podatkovnih naborih lahko izboljša točnost, a ne znamo vnaprej predvideti, na katerih, predlagamo, da se ustreznost uporabe skladanja testira s prečnim preverjanjem. Enak postopek smo uspešno uporabili še na drugi domeni: ocenjevanju zanesljivosti napovedi lastnosti kemikalij.

Predlagali smo tudi prilagoditev metode SetSig, po kateri lahko transformacijo s skladanjem izvedemo v enakem času, kot bi ga potrebovali za transformacijo brez skladanja.

6

Zaključek

Motivacija v disertaciji predstavljenih raziskav so bili raziskovalni problemi s področja biologije in medicine, kjer je zbiranje vzorcev drago in zamudno. Podatkovni nabori s teh področij imajo zato pogosto malo učnih primerov, ki so opisani s tipično nekaj tisoč značilkami, zato je z njimi težko zgraditi napovedne modele z dobro točnostjo na novih podatkih. Še težje je take modele interpretirati – tudi zato, ker je vloga precejšnjega števila značilk (genov) še neznana. Po drugi strani pa imamo o proučevanih objektih na voljo veliko predznanja. Nekaj ga je formaliziranega tudi v obliki grafov ali skupin. V disertaciji smo proučevali transformacijo podatkov iz prostora značilk v prostor skupin značilk in napovedovanje ciljne spremenljivke na tako transformiranih podatkih. Kljub temu, da so razvite metode motivirali problemi pri gradnji napovednih modelov za podatke o genskih izrazih, na katerih smo izsledke tudi preizkušali, razvite metode niso omejene na biološko domeno. Delovale bi lahko s kakršnimikoli podatki, kjer imamo predznanje o značilkah podano v obliki skupin značilk.

V disertaciji smo preizkusili obstoječe metode za transformacijo v prostor skupin značilk, s katerimi v povprečju nismo uspeli preseči točnosti na netransformiranih podatkih, a so razlike med točnostmi modelov na netransformiranih podatkih in podatkih transformiranih z boljšimi metodami majhne.

Predlagali smo novo metodo za transformacijo učnih in testnih podatkov v prostor skupin značilk, ki temelji na sočasni matrični faktorizaciji. Modeli logistične regresije na s predlagano metodo transformiranih podatkih so značilno bolj točni kot na podatkih s transformacijo brez faktorizacije in podobno točni kot modeli na podatkih transformiranih z najboljšimi obstoječimi metodami, ki za razliko od predlagane metode pri transformaciji uporabljajo tudi vrednost ciljne spremenljivke. Predlagana metoda je tako ob primerljivi točnosti splošnejša: lahko jo uporabljamo ne glede na tip ciljne spremenljivke oziroma tudi v vrstah analize podatkov, ko ciljne spremenljivke ne poznamo. Ker si faktorizirani matriki primarnih podatkov in skupin značilk delita skupen latentni faktor, pri faktorizaciji primarna matrika podatkov vpliva na sestavo skupin. Transformacija v vrednost skupine značilk lahko upošteva tudi značilke, ki skupini ne pripadajo, so pa značilkam skupine podobne. Slednje je na področju molekularne biologije intuitivno smiselno, saj skupine odražajo zgolj trenutno znanje, ki ga raziskovalci stalno dopolnjujejo.

Za transformacijske metode, ki vrednosti ciljne spremenljivke uporabljajo pri gradnji transformacijskih modelov, smo predlagali uporabo skladanja, s katerim se izognemo pretirani prilagoditvi parametrov transformacij učnim podatkom. Na nekaterih

podatkovnih naborih je skladanje precej izboljšalo rezultate, a smo opazno izboljšanje dosegli zgolj na majhni množici resničnih podatkovnih naborov.

Metode, ki smo jih preučevali v disertaciji, so omejene na skupine, ki morajo neposredno opisovati značilke primarnih podatkov, zato ne morejo upoštevati drugačnega predznanja, ki je ali strukturirano drugače, recimo v obliki grafov, ali pa je z izvornimi podatki povezano zgolj posredno. Trenutno najzanimivejše metode za analizo podatkov na področju molekularne biologije so metode zlivanja podatkov, ki lahko upoštevajo predznanje različnih tipov in z našimi podatki zgolj posredno povezano predznanje. Z njimi so uspešno zlili tudi več deset različnih podatkovnih virov. Možna alternativa uporabi eksplicitno podanega predznanja bi bila uporaba globokih nevronskih mrež, ki iz sorodnih podatkov same ustvarijo strukture, s katerimi laže napovedujejo. Na področju molekularne biologije je veliko javno dostopnih podatkov, iz katerih bi se učne metode lahko na nek način same "naučile" predznanja. Za razliko od metod zlivanja podatkov tako zgrajeno predznanje ne bi bilo omejeno na vnaprej definirane strukture baz znanja, ampak bi bilo lahko v poljubni, morda nam nerazumljivi, a učnemu algoritmu laže uporabni obliki.

Prednost uporabe metod, ki transformirajo podatke v prostor skupin značilk, pred kompleksnejšimi metodami, kot so metode zlivanja podatkov in nevronske mreže, je prav razumljivost končnih napovednih modelov. Pri analizi podatkov še vedno kdaj zaradi lažje interpretacije uporabimo preproste tehnike, kot so denimo napovedna drevesa, čeprav poznamo tehnike, ki ponavadi zgradijo bolj točne modele. Zaradi istega razloga je lahko tudi uporaba transformacije v prostor skupin značilk smiselna.

6.1 Glavna prispevka k znanosti

Prispevke k znanosti smo opisali v uvodu disertacije (razdelek 1.2), poudarili bi pa:

- Poglobljeno smo preučili tehnike transformacije podatkov iz prostora značilk v prostor vnaprej definiranih skupin značilk. V največji študiji doslej smo pokazali, da z gradnjo napovednih modelov na transformiranih podatkih ne izboljšamo točnosti, še vedno pa dobimo zadovoljivo dobre napovedne modele, da jih je zaradi prednosti pri interpretaciji smiselno uporabljati.

- Predlagali smo metodo za transformacijo vhodnih značilk v značilke, ki opisujejo skupine. Metoda temelji na sočasni matrični faktorizaciji in uporablja vir podatkov, ki opisuje skupine značilk.

6.2 Nadaljnje delo

Ena ključnih omejitev pričujoče raziskave je gradnja napovednih modelov zgolj na transformiranih značilkah. Vse napovedne modele bi namreč lahko gradili tudi v kombinacijami z originalnimi značilkami. Če značilke kombiniramo, redundantnost transformiranih značilk glede na originalne značilke postane pomembna, ker bi v kombinaciji značilk ločevala metode, ki bi se zgolj na transformiranih značilkah podobno obnesle. Pomembno odprto vprašanje je še, kako značilke kombinirati, saj je na podatkih o genskih izrazih transformiranih značilk precej manj kot originalnih.

Skozi celotno disertacijo smo uporabljali zgolj dve klasifikacijski metodi. Za splošnejše zaključke bi jih morali uporabi čim več. Več opisanih poskusov je možno izboljšati. V primerjavi metod v razdelku 2.3 je zmagala metoda SpLin, četudi smo ji nastavili fiksno stopnjo regularizacije za vse skupine značilk. Nastavljanje stopnje regularizacije za vsako skupino posebej s prečnim preverjanjem je zaradi števila skupin nepraktično, lahko bi pa stopnjo regularizacije povezali s kakovostjo skupine ali vsaj z njeno velikostjo. V poglavju o podobnosti genov v genski skupinah (poglavje 3) smo vrednotili le pare genov z dvema metodama; ker so lahko v interakciji tudi večje skupine genov, bi bilo raziskavo smiselno razširiti še nanje. Pri vrednotenju transformacije z matrično faktorizacijo (poglavje 4) bi lahko latentne dimenzije matrik $\mathbf{G}_1$, $\mathbf{G}_2$ in $\mathbf{G}_3$ nastavljali posebej in to za obe vrsti faktorizacije, sočasno in ločeno. Prav tako nismo eksperimentirali s številom iteracij algoritma DFMF. Pri drugem številu iteracij bi se lahko kot najboljše izkazale druge vrednosti latentnih dimenzij. Zanimivo bi bilo tudi poiskati vzrok slabšim točnostim naključnih gozdov na podatkih, ki smo jih transformirali z matrično faktorizacijo.

Metode transformacije smo na resničnih podatkih primerjali glede na povprečne range čez več naborov podatkov in jih s tem ovrednotili. Ko rešujemo problem na konkretnih podatkih, nas zanima le, katera metoda bo na prav teh podatkih dobro delovala. Predvsem za transformacije s skladanjem (poglavje 5) bi bilo smiselno podrobneje raziskati, na kakšnih kombinacijah podatkovnih naborov, skupin značilk in končnih klasifikatorjev delujejo dobro.

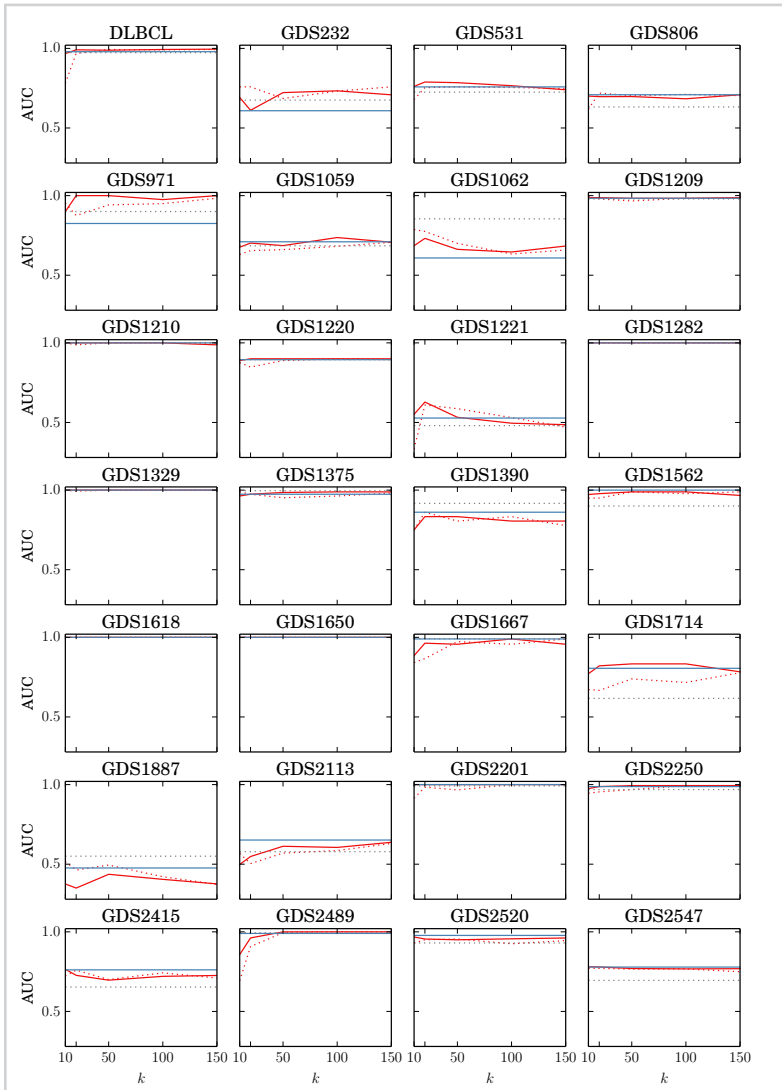
Na kakovost napovednih modelov s transformacijami v prostor skupin gotovo vpliva tudi kakovost predznanja, kar so s primerjavo različnih virov skupin značilk jasno pokazali Holec et al. [117]. Koristno bi bilo razviti mero, ki bi ovrednotila skupine značilk in bi jo lahko uporabljali za filtriranje skupin. Naši dosedanji poskusi, ki so

temeljili na vrednotenju korelacij znotraj skupine, se niso izkazali.

V določenih primerih porazdelitev značilk poznamo. Če genske izraze merimo z visoko-prepustim sekvenciranjem, vrednosti značilk dobro ustrezajo negativni binomski porazdelitvi, čemur so prilagodili metode iskanja diferencialno izraženih genov [118]. Takšnim podatkom bi bilo zanimivo prilagoditi transformacijo v prostor skupin značilk s sočasno matrično faktorizacijo.

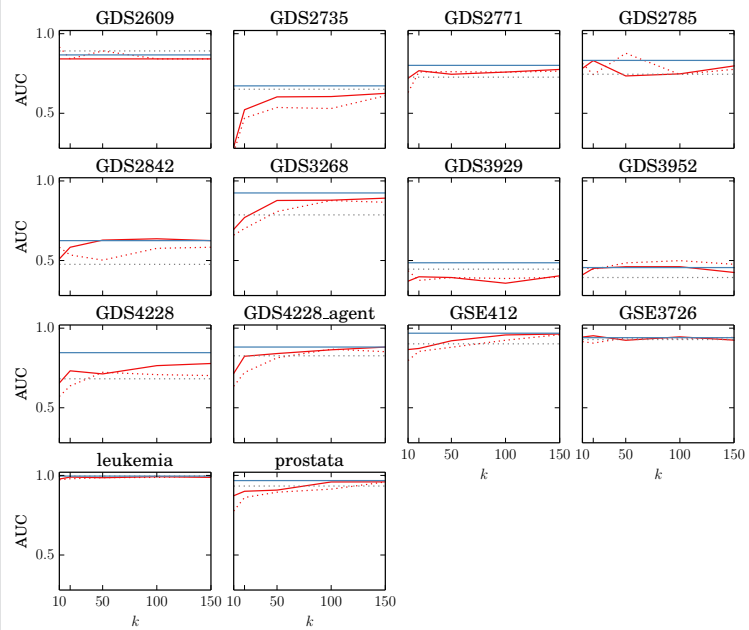
Dodatni rezultati

A



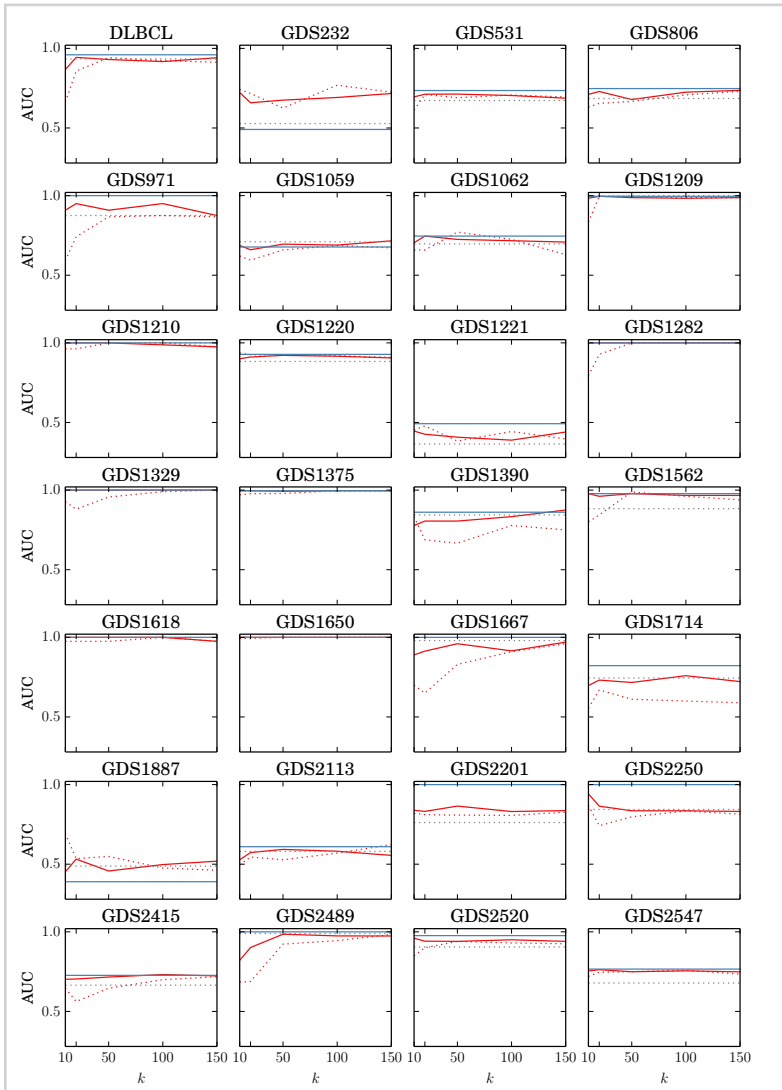
Slika A.1

Dodatni rezultati poglavja 4: vpliv latentnih dimenzij k na napovedno točnost modelov logistične regresije zgrajenih na podatkih transformiranih v prostor skupin značilk s sočasno (rdeče polne črte) in ločeno (rdeče pikčaste črte) matrično faktorizacijo. Modra črta prikazuje točnost na netransformiranih značilkah, pikčasta siva pa točnost zmnožka **XF**. Nadaljevanje sledi na sliki A.2.



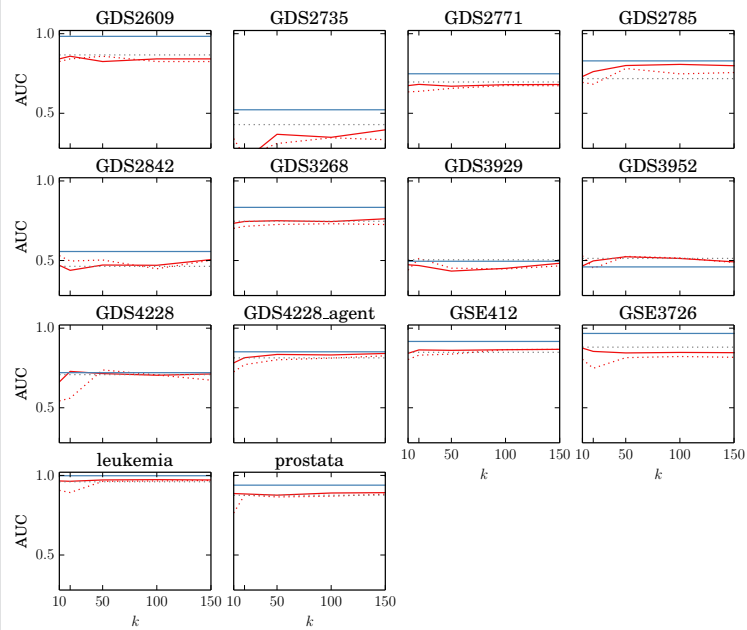
Slika A.2

Nadaljevanje slike A.1.



Slika A.3

Dodatni rezultati poglavja 4: vpliv latentnih dimenzij k na napovedno točnost modelov naključnih gozdov zgrajenih na podatkih transformiranih v prostor skupin značilk s sočasno (rdeče polne črte) in ločeno (rdeče pikčaste črte) matrično faktorizacijo. Modra črta prikazuje točnost na netransformiranih značilkah, pikčasta siva pa točnost zmnožka **XF**. Nadaljevanje sledi na sliki A.4.



Slika A.4

Nadaljevanje slike A.3.

LITERATURA

[1] I. Guyon, A. Saffari, G. Dror, in G. Cawley. Analysis of the ijcnn 2007 agnostic learning vs. prior knowledge challenge. *Neural Networks*, 21(2):544–550, 2008.

[2] David J. Lockhart, Helin Dong, Michael C. Byrne, Maximillian T. Follettie, Michael V. Gallo, Mark S. Chee, Michael Mittmann, Chunwei Wang, Michiko Kobayashi, Heidi Horton, et al. Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature Biotechnology*, 14(13):1675–1680, 1996.

[3] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–7, 1999.

[4] M. F. Barginear, T. Bradley, I. Shapira, in D. R. Budman. Implications of applied research for prognosis and therapy of breast cancer. *Critical Reviews in Oncology/Hematology*, 65(3):223–34, 2008.

[5] Lin Zhang, Wei Zhou, Victor E. Velculescu, Scott E. Kern, Ralph H. Hruban, Stanley R. Hamilton, Bert Vogelstein, in Kenneth W. Kinzler. Gene expression profiles in normal and cancer cells. *Science*, 276 (5316):1268–1272, 1997.

[6] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, in J. P. Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–50, 2005.

[7] S. Draghici, P. Khatri, A. C. Eklund, in Z. Szallasi. Reliability and reproducibility issues in DNA microarray measurements. *Trends in Genetics*, 22(2): 101–109, 2006.

[8] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, in M. Hirakawa. KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Research*, 38(Database issue):D355, 2010.

[9] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, et al. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1): 25–29, 2000.

[10] L. J. Jensen, M. Kuhn, M. Stark, S. Chaffron, C. Creevey, J. Muller, T. Doerks, P. Julien, A. Roth, M. Simonovic, et al. STRING 8——a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, 37(suppl 1): D412–D416, 2009.

[11] Online Mendelian Inheritance in Man, OMIM, 2016. URL http://omim.org.

[12] D. Nam in S. Y. Kim. Gene-set approach for expression pattern analysis. *Briefings in Bioinformatics*, 9(3): 189–97, 2008.

[13] Atul Butte. The use and analysis of microarray data. *Nature reviews drug discovery*, 1(12):951–960, 2002.

[14] Z. Guo, T. Zhang, X. Li, Q. Wang, J. Xu, H. Yu, J. Zhu, H. Wang, C. Wang, E. J. Topol, in S. Rao. Towards precise classification of cancers based on robust gene functional expression profiles. *BMC Bioinformatics*, 6:58, 2005.

[15] Minca Mramor, Marko Toplak, Gregor Leban, Tomaž Curk, Janez Demšar, in Blaž Zupan. On utility of gene set signatures in gene expression-based cancer class prediction. In S. Džeroski, P. Geurts, in J. Pousu, editors, *Machine learning in systems biology : proceedings of the Third International Workshop*, pages 65–73. Helsinki: Department of Computer Science, University, 2009.

[16] G. Abraham, A. Kowalczyk, S. Loi, I. Haviv, in J. Zobel. Prediction of breast cancer prognosis using gene set statistics provides signature stability and biological context. *BMC Bioinformatics*, 11(1):277, 2010.

[17] J. Klema, M. Holec, F. Zelezny, in J. Tolar. Comparative evaluation of set-level techniques in microarray classification. *Bioinformatics Research and Applications*, pages 274–285, 2011.

[18] Christine Staiger, Sidney Cadot, Raul Kooter, Marcus Dittrich, Tobias Müller, Gunnar W Klau, in Lodewyk FA Wessels. A critical evaluation of network and pathway-based classifiers for outcome prediction in breast cancer. *PloS One*, 7(4):e34796, 2012.

[19] Aleks Jakulin in Ivan Bratko. Analyzing attribute dependencies. In *PKDD 2003, volume 2838 of LNAI*, pages 229–240. Springer-Verlag, 2003.

[20] N. Bhardwaj in H. Lu. Correlation between gene expression profiles and protein–protein interactions within and across genomes. *Bioinformatics*, 21(11): 2730, 2005.

[21] H. B. Fraser, A. E. Hirsh, D. P. Wall, in M. B. Eisen. Coevolution of gene expression among interacting proteins. *Proceedings of the National Academy of Sciences*, 101(24):9033, 2004.

[22] R. Jansen, D. Greenbaum, in M. Gerstein. Relating whole-genome expression data with protein-protein interactions. *Genome research*, 12(1):37, 2002.

[23] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 426–434. ACM, 2008.

[24] Jaewon Yang in Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM, 2013.

[25] Vladimir Gligorijević in Nataša Pržulj. Methods for biological data integration: perspectives and challenges. *Journal of The Royal Society Interface*, 12(112): 20150571, 2015.

[26] Marinka Žitnik. *Učenje z zlivanjem heterogenih podatkov*. PhD thesis, Univerza v Ljubljani, 2015.

[27] Marinka Žitnik in Blaž Zupan. Data Fusion by Matrix Factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):41–53, 2015.

[28] Marinka Žitnik, Edward A Nam, Christopher Dinh, Adam Kuspa, Gad Shaulsky, in Blaž Zupan. Gene prioritization by compressive data fusion and chaining. *PLoS Computational Biology*, 11(10):e1004552, 2015.

[29] E. Bair, T. Hastie, D. Paul, in R. Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101:119–137, March 2006.

[30] X. Chen, L. Wang, J. D. Smith, in B. Zhang. Supervised principal component analysis for gene set enrichment of microarray data with continuous or survival outcomes. *Bioinformatics*, 24(21):2474–81, 2008.

[31] J. Liu, J. M. Hughes-Oliver, in Jr. Menius, J. A. Domain-enhanced analysis of microarray data using GO annotations. *Bioinformatics*, 23(10):1225–34, 2007.

[32] Bradley Efron in Robert Tibshirani. On testing the significance of sets of genes. *The Annals of Applied Statistics*, 1(1):107–29, 2007.

[33] E. Lee, H. Y. Chuang, J. W. Kim, et al. Inferring pathway activity toward precise disease classification. *PLoS Computational Biology*, 4(11):e1000217, 11 2008.

[34] M. C. Wu, L. Zhang, Z. Wang, D. C. Christiani, in X. Lin. Sparse linear discriminant analysis for simultaneous testing for the significance of a gene set/pathway and gene selection. *Bioinformatics*, 25(9): 1145–1151, 2009.

[35] Junjie Su, Byung-Jun Yoon, in Edward R Dougherty. Accurate and reliable cancer classification based on probabilistic inference of pathway activity. *PLoS One*, 4(12):e8161, 2009.

[36] E. Edelman, A. Porrello, J. Guinney, B. Balakumaran, A. Bild, P. G. Febbo, in S. Mukherjee. Analysis of sample set enrichment scores: assaying the enrichment of sets of genes for individual samples in genome-wide expression profiles. *Bioinformatics*, 22 (14):e108–16, 2006.

[37] L. Li. Exploiting predictor domain information in sufficient dimension reduction. *Computational Statistics & Data Analysis*, 53(7):2665–2672, 2009.

[38] David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

[39] T. M. Cover in P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[40] C. Cortes in V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[41] Leo Breiman. Random forests. *Machine Learning*, 45 (1):5–32, 2001.

[42] Sašo Džeroski in Nada Lavrač. Rule induction and instance-based learning applied in medical diagnosis. *Technology and Health Care*, 4(2):203–221, 1996.

[43] Yann LeCun, Yoshua Bengio, in Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[44] M. I. Jordan in T. M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349 (6245):255–260, 2015.

[45] Brenden M. Lake, Ruslan Salakhutdinov, in Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350 (6266):1332–1338, 2015.

[46] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[47] Marko Robnik-Šikonja in Igor Kononenko. Theoretical and empirical analysis of relieff and rrelieff. *Machine learning*, 53(1-2):23–69, 2003.

[48] Ron Kohavi in George H. John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[49] Blaž Zupan, Ivan Bratko, Marko Bohanec, in Janez Demšar. Induction of concept hierarchies from noisy data. In *Seventeenth International Conference on Machine Learning*, pages 1199–1206, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[50] Yupeng Cun in Holger Fröhlich. Biomarker gene signature discovery integrating network knowledge. *Biology*, 1(1):5–17, 2012.

[51] Paul Pavlidis, Jason Weston, Jinsong Cai, in William Stafford Noble. Learning gene functional classifications from multiple data types. *Journal of Computational Biology*, 9(2):401–411, 2002.

[52] Shiliang Sun. A survey of multi-view machine learning. *Neural Computing and Applications*, 23(7-8): 2031–2038, 2013.

[53] H. Liu in H. Motoda. *Feature extraction, construction and selection: a data mining perspective*. Springer, 1998.

[54] Minca Mramor. *Gradnja napovednih modelov in odkrivanje znanja iz podatkov DNA mikromrež*. PhD thesis, Univerza v Ljubljani, 2010.

[55] Ivan Bratko. *Prolog programming for artificial intelligence*. Addison-Wesley Longman Ltd, 2001.

[56] Einar Ryeng in Bjørn Kåre Alsberg. Microarray data classification using inductive logic programming and gene ontology background information. *Journal of Chemometrics*, 24(5):231–240, 2010.

[57] Igor Trajkovski, Filip Železn, Nada Lavrač, in Jakub Tolar. Learning relational descriptions of differentially expressed gene groups. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(1):16–25, 2008.

[58] Marc Johannes, Jan C. Brase, Holger Fröhlich, Stephan Gade, Mathias Gehrmann, Maria Fälth, Holger Sültmann, in Tim Beißbarth. Integration of pathway knowledge into a reweighted recursive feature elimination approach for risk stratification of cancer patients. *Bioinformatics*, 26(17):2136–2144, 2010.

[59] Isabelle Guyon, Jason Weston, Stephen Barnhill, in Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.

[60] Sergey Brin in Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International Web Conference (WWW98)*, 1998.

[61] Julie L Morrison, Rainer Breitling, Desmond J Higham, in David R Gilbert. Generank: using search engine technology for the analysis of microarray experiments. *BMC Bioinformatics*, 6(1):233, 2005.

[62] Yupeng Cun in Holger Fröhlich. Network and data integration for biomarker signature discovery via network smoothed T-statistics. *PloS One*, 8(9):e73074, 2013.

[63] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, in Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[64] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[65] Caiyan Li in Hongzhe Li. Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175–1182, 2008.

[66] Koh Takeuchi, Yoshinobu Kawahara, in Tomoharu Iwata. Higher order fused regularization for supervised learning with grouped parameters. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 577–593. Springer, 2015.

[67] Yanni Zhu, Xiaotong Shen, in Wei Pan. Network-based support vector machine for classification of microarray samples. *BMC Bioinformatics*, 10(Suppl 1):S21, 2009.

[68] F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, in J.P. Vert. Classification of microarray data using gene networks. *BMC Bioinformatics*, 8(1):35, 2007.

[69] Ofer Lavi, Gideon Dror, in Ron Shamir. Network-induced classification kernels for gene expression profile analysis. *Journal of Computational Biology*, 19 (6):694–709, 2012.

[70] Michael Anděl, Jiří Kléma, in Zdeněk Krejčík. Network-constrained forest for regularized classification of omics data. *Methods*, 2015.

[71] R. S. Michalski. Pattern recognition as rule-guided inductive inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (4):349–361, 1980.

[72] S. Kramer. CN2-MCI: A two-step method for constructive induction. In *Proceedings of ML-COLT '94 Workshop on Constructive Induction and Change of Representation*, 1994.

[73] H. Y. Chuang, E. Lee, Y. T. Liu, D. Lee, in T. Ideker. Network-based classification of breast cancer metastasis. *Molecular Systems Biology*, 3(1), 2007.

[74] M. Gutkin, R. Shamir, in G. Dror. SlimPLS: a method for feature selection in gene expression-based disease classification. *PLoS One*, 4(7):e6416, 2009.

[75] Andrew Y. Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first International Conference on Machine Learning*, page 78. ACM, 2004.

[76] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, in Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[77] Lexin Li. Exploiting predictor domain information in sufficient dimension reduction. *Computational Statistics & Data Analysis*, 53(7):2665–2672, 2009.

[78] R Dennis Cook. Graphics for regressions with a binary response. *Journal of the American Statistical Association*, 91(435):983–992, 1996.

[79] Seungwoo Hwang. Comparison and evaluation of pathway-level aggregation methods of gene expression data. *BMC Genomics*, 13(Suppl 7):S26, 2012.

[80] T. Barrett, D. B. Troup, S. E. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. F. Kim, A. Soboleva, M. Tomashevsky, in R. Edgar. NCBI GEO: mining tens of millions of expression profiles—database and tools update. *Nucleic Acids Research*, 35:760–5, 2007.

[81] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[82] L. I. Kuncheva. A stability index for feature selection. In *Proceedings of the 25th IASTED International Multi-Conference: artificial intelligence and applications*, pages 390–395. ACTA Press, 2007.

[83] Janez Demšar, Tomaž Curk, Aleš Erjavec, Črt Gorup, Tomaž Hočevar, Mitar Milutinovič, Martin Možina, Matija Polajnar, Marko Toplak, Anže Starič, et al. Orange: data mining toolbox in Python. *Journal of Machine Learning Research*, 14(1):2349–2353, 2013.

[84] Christian Münch in J. Wade Harper. Mitochondrial unfolded protein response controls matrix pre-RNA processing and translation. *Nature*, 534(7609): 710–713, 2016.

[85] Sandra Blanco, Roberto Bandiera, Martyna Popis, Shobbir Hussain, Patrick Lombard, Jelena Aleksic, Abdulrahim Sajini, Hinal Tanna, Rosana Cortés-Garrido, Nikoletta Gkatza, et al. Stem cell function and stress response are controlled by protein synthesis. *Nature*, 534(7609), 2016.

[86] Waleed Nasser, Balaji Santhanam, Edward Roshan Miranda, Anup Parikh, Kavina Juneja, Gregor Rot, Chris Dinh, Rui Chen, Blaz Zupan, Gad Shaulsky, et al. Bacterial discrimination by dictyostelid amoebae reveals the complexity of ancient interspecies interactions. *Current Biology*, 23(10):862–872, 2013.

[87] Paulina Fuentes, Fei Zhou, Alexander Erban, Daniel Karcher, Joachim Kopka, in Ralph Bock. A new synthetic biology approach allows transfer of an entire metabolic pathway from a medicinal plant to a biomass crop. *eLife*, 5:e13664, 2016.

[88] A. Bhatt, I. Kaverina, C. Otey, in A. Huttenlocher. Regulation of focal complex composition and disassembly by the calcium-dependent protease calpain. *J Cell Sci*, 115(17):3415–3425, 2002.

[89] Riccardo Bellazzi in Blaž Zupan. Towards knowledge-based gene expression data mining. *Journal of Biomedical Informatics*, 40(6):787–802, 2007.

[90] C. Stark, B.J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, in M. Tyers. BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34(suppl 1):D535, 2006.

[91] M. Jelizarow, V. Guillemot, A. Tenenhaus, K. Strimmer, in A. L. Boulesteix. Over-optimism in bioinformatics: an illustration. *Bioinformatics*, 26(16):1990, 2010.

[92] H. K. Lee, A. K. Hsu, J. Sajdak, J. Qin, in P. Pavlidis. Coexpression analysis of human genes across many microarray data sets. *Genome Research*, 14(6):1085, 2004.

[93] Marko Toplak, Tomaž Curk, Janez Demšar, in Blaž Zupan. Does replication groups scoring reduce false positive rate in SNP interaction discovery? *BMC Genomics*, 11(1):1, 2010.

[94] Marko Toplak, Tomaž Curk, in Blaž Zupan. Similarity of transcription profiles for genes in gene sets. In *International Conference on Adaptive and Natural Computing Algorithms*, pages 393–399. Springer, 2011.

[95] Javier Gayán, Antonio González-Pérez, Fernando Bermudo, María Eugenia Sáez, Jose Luis Royo, Antonio Quintas, Jose Jorge Galan, Francisco Jesús Morón, Reposo Ramirez-Lorca, Luis Miguel Real, et al. A method for detecting epistasis in genome-wide studies using case-control multi-locus association analysis. *BMC Genomics*, 9(1):1, 2008.

[96] D. Sheskin. *Handbook of parametric and nonparametric statistical procedures*. CRC Pr I Llc, 2004.

[97] Dimitris Anastassiou. Computational analysis of the synergy among multiple interacting genes. *Molecular Systems Biology*, 3(83), February 2007. doi: 10.1038/msb4100124.

[98] Martin H. van Vliet, Christiaan N Klijn, Lodewyk F. A. Wessels, in Marcel J. T. Reinders. Module-based outcome prediction using breast cancer compendia. *PLoS One*, 2(10):e1047, 2007.

[99] Fei Wang, Tao Li, in Changshui Zhang. Semi-supervised clustering via matrix factorization. In *SDM*, pages 1–12. SIAM, 2008.

[100] Marinka Žitnik in Blaž Zupan. Matrix factorization-based data fusion for drug-induced liver injury prediction. *Systems Biomedicine*, 2(1):16–22, 2014.

[101] Rasmus Bro in Sijmen De Jong. A fast non-negativity-constrained least squares algorithm. *Journal of Chemometrics*, 11(5):393–401, 1997.

[102] Amy N. Langville, Carl D. Meyer, Russell Albright, James Cox, in David Duling. Algorithms, initializations, and convergence for the nonnegative matrix factorization. *arXiv preprint arXiv:1407.7299*, 2014.

[103] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(jan):1–30, 2006.

[104] Matěj Holec, Jiří Kléma, Filip Železný, in Jakub Tolar. Comparative evaluation of set-level techniques in predictive classification of gene expression samples. *BMC Bioinformatics*, 13(10):1, 2012.

[105] R. Vilalta in Y. Drissi. A perspective view and survey of meta-learning. *Artificial Intelligence Review*, 18(2): 77–95, 2002.

[106] Sašo. Džeroski in Bernard Ženko. Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273, 2004.

[107] Kai Ming Ting in Ian H. Witten. Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289, 1999.

[108] David H. Wolpert in William G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[109] Marko Toplak, Rok Močnik, Matija Polajnar, Zoran Bosnić, Lars Carlsson, Catrin Hasselgren, Janez Demšar, Scott Boyer, Blaž Zupan, in Jonna Stålring. Assessment of machine learning reliability methods for quantifying the applicability domain of QSAR regression models. *Journal of Chemical Information and Modeling*, 54(2):431–441, 2014.

[110] OECD. Guidance document on the validation of (quantitative) structure-activity relationship QSAR models, oecd series on testing and assessment no.69. env/jm/mono. Technical Report February, OECD Series on Testing and Assessment No.69. ENV/JM/MONO, Paris, France, 2007.

[111] Zoran Bosnić in Igor Kononenko. Automatic selection of reliability estimates for individual regression predictions. *Knowledge Engineering Review*, 25(1): 27–47, 2010.

[112] Shane Weaver in M. Paul Gleeson. The importance of the domain of applicability in QSAR modeling. *Journal of Molecular Graphics & Modelling*, 26(8): 1315–26, 2008.

[113] Pierre Bruneau in Nathan R. Mcelroy. logD7.4 modeling using bayesian regularized neural networks: assessment and correction of the errors of prediction. *Journal of Chemical Information and Modeling*, 46: 1379–1387, 2006.

[114] Zoran Bosnić in Igor Kononenko. Comparison of approaches for estimating reliability of individual regression predictions. *Data & Knowledge Engineering*, 67(3):504–516, 2008.

[115] Ruchi R. Mittal, Ross A. McKinnon, in Michael J. Sorich. Comparison data sets for benchmarking QSAR methodologies in lead optimization. *Journal of Chemical Information and Modeling*, 49(7): 1810–1820, 2009.

[116] Chih-Chung Chang in Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[117] Matěj Holec, Ondřej Kuželka, et al. Novel gene sets improve set-level classification of prokaryotic gene expression data. *BMC Bioinformatics*, 16(1):1, 2015.

[118] Thomas J. Hardcastle in Krystyna A. Kelly. baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, 11(1):422, 2010.