Redukcija dimenzija u dubinskoj analizi podataka

Gašparović, Leo

Sveučilište u Rijeci – Odjel za informatiku

Diplomski sveučilišni studij informatike

Modul Poslovna informatika

Leo Gašparović

Redukcija dimenzija u dubinskoj analizi podataka

Diplomski rad

Mentor: prof. dr. sc. Maja Matetić

Rijeka, lipanj 2020.

ODJEL ZA INFORMATIKU
SVEUČILIŠTE U RIJECI

Rijeka, 9. ožujka 2020.

Zadatak za diplomski rad

Pristupnik: Leo Gašparović

Naziv diplomskog rada: **Redukcija dimenzija u dubinskoj analizi podataka**

Naziv diplomskog rada na eng. jeziku: Dimension reduction in data mining

Sadržaj zadatka:

Dimenzija skupa podataka, koja predstavlja broj varijabli u promatranom skupu podataka, često je vrlo velika i mora se smanjiti kako bi algoritmi dubinske analize podataka djelovali učinkovito. Taj je postupak dio faze pripreme za dubinsku analizu podataka i provodi se prije izrade samog modela. Postoji mnogo različitih tehnika redukcije dimenzija, a svima je cilj pronaći reprezentaciju podataka u prostoru nižih dimenzija koja zadržava što više informacija. U ovom radu će biti prikazane neke od mnogobrojnih tehnika redukcije kao što su razni sažetci skupa podataka, agregacija, analiza korelacije, pivot tablice, PCA i ostale. Odabrane tehnike redukcije dimenzije u radu će biti prvo objašnjene teoretskom osnovom, a nakon toga praktičnim primjerom izrađenim u programskom jeziku R.

Mentor:
Prof. dr. sc. Maja Matetić

Voditeljica za diplomske radove:
Izv. prof. dr. sc. Ana Meštrović

Komentor:

Zadatak preuzet: 23. lipnja 2020.

Leo Gašparović

(potpis pristupnika)

Sadržaj

Sažetak

Dimenzija skupa podataka, koja predstavlja broj varijabli u promatranom skupu podataka, često je vrlo velika i mora se smanjiti kako bi algoritmi dubinske analize podataka djelovali učinkovito. Taj je postupak dio faze pripreme za dubinsku analizu podataka i provodi se prije izrade samog modela. Čak i kada je početni broj varijabli mali, skup podataka se brzo proširuje u koraku pripreme podataka, gdje se stvaraju nove izvedene varijable. U takvim je situacijama vjerojatno da su podskupovi varijabli međusobno visoko korelirani. Uključivanje visoko koreliranih varijabli u model klasifikacije ili predviđanja ili uključivanje varijabli koje nisu povezane s traženim rezultatom može dovesti do prekomjernog preklapanja podataka, a točnost i pouzdanost mogu se znatno smanjiti. U implementaciji modela, suvišne varijable mogu čak i povećati troškove zbog prikupljanja i obrade tih varijabli. Upravo stoga je potrebno provesti metode redukcije dimenzije skupa podataka. Postoji mnogo različitih tehnika redukcije dimenzija, a svima je cilj pronaći reprezentaciju podataka u prostoru nižih dimenzija koja zadržava što više informacija. Uglavnom se sve tehnike svode ili na eliminaciju značajki, koja označava potpuno uklanjanje varijabli koje nisu relevantne za daljnju analizu, ili ekstrakciju značajki, koja smanjuje dimenziju skupa podataka stvaranjem nove varijable kombiniranjem postojećih varijabli. U ovom radu će biti prikazane neke od mnogobrojnih tehnika redukcije kao što su razni sažetci skupa podataka, analiza korelacije, PCA i ostale. Odabrane tehnike redukcije dimenzije u radu će biti prvo objašnjene teoretskom osnovom, a nakon toga praktičnim primjerom izrađenim u programskom jeziku R.

Ključne riječi

Dubinska analiza podataka, skup podataka, dimenzija skupa, redukcija dimenzija, sažeci skupa podataka, korelacija, PCA, regresijski modeli

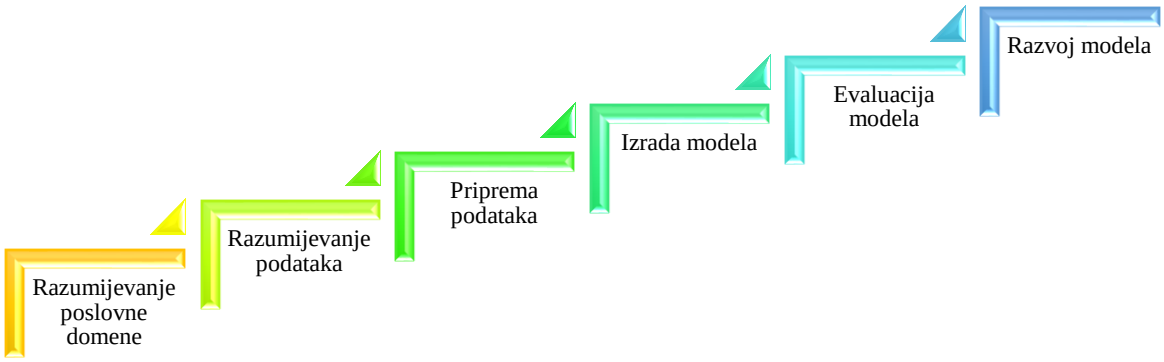
1. Uvod

Dubinska analiza podataka (engl. *Data Mining*) relativno je nova znanstvena disciplina. Predstavlja proces pronalaska anomalija, obrazaca i korelacija unutar velikih skupova podataka za predviđanje određenog ishoda. Pomoću širokog raspona tehnika, informacije dobivene dubinskom analizom podataka mogu se koristiti za povećanje prihoda, smanjenje troškova, poboljšanje odnosa s kupcima, smanjenje rizika i još mnogo različitih namjena[6].

Proces dubinske analize podataka radi otkrivanja skrivenih veza i predviđanja budućih trendova ima dugu povijest. Izraz „dubinska analiza podataka“ nije postojao sve do 1990-ih. Njegovo utemeljenje sadrži tri isprepletene znanstvene discipline: statistiku (numeričko proučavanje odnosa podataka), umjetnu inteligenciju (ljudsku inteligenciju prikazanu softverom i/ili strojevima) i strojno učenje (algoritmi koji mogu naučiti iz podataka da bi predvidjeli) (Slika 1). Tehnologija dubinske analize podataka stalno se razvija kako bi išla u korak s neograničenim potencijalom velikih podataka (engl. *Big Data*) i pristupačnom računalnom snagom. U posljednjem desetljeću napredak u snazi i brzini obrade podataka omogućio je prelazak sa „ručne“, zamorne i dugotrajne prakse na brzu, jednostavnu i automatiziranu analizu podataka. Što su složeniji skupovi podataka, to je veći potencijal za otkrivanje relevantnih uvida.

Slika 1: Utemeljenje dubinske analize podataka

Tema ovog diplomskog rada je redukcija dimenzija skupa podataka. Dimenzija skupa podataka, za koju se može reći da je broj varijabli u promatranom skupu podataka, ne smije biti velika kako bi izrađeni model bio pouzdan. Ovaj je postupak dio faze pripreme za dubinsku analizu podataka i provodi se prije izrade samog modela (Slika 2).



Slika 2: Koraci dubinske analize podataka

Što je broj varijabli u promatranom skupu podataka veći, prostor podataka postaje sve manji, a modeli klasifikacije i predviđanja propadaju, jer dostupni podaci nisu dovoljni za pružanje korisnog modela za toliko mnogo varijabli. Važna je činjenica da se poteškoće nastale dodavanjem varijable povećavaju eksponencijalno dodavanjem svake varijable. Statistički gledano, povećavanje broja varijabli znači da više ništa nije „blizu“ - dodano je previše buke, a obrasci i struktura više se ne prepoznaju. Stoga je jedan od ključnih koraka u pripremi podataka pronalaženje načina za smanjenje dimenzionalnosti uz minimalno smanjenje točnosti[7].

Postoji mnogo različitih tehnika redukcije dimenzija koje pokušavaju pronaći reprezentaciju podataka u prostoru nižih dimenzija koji zadržava što više informacija. Postoje dvije klase tehnika redukcije dimenzija: eliminacija značajki je u potpunosti uklanjanje nekih varijabli ako su suvišne u kombinaciji s nekom drugom varijablom ili ako ne pružaju nikakve nove podatke o skupu podataka. Prednost uklanjanja značajki je ta što je to jednostavno implementirati i čini skup podataka malim, uključujući samo varijable koje su bitne za daljnju analizu. Ali kao nedostatak treba istaknuti da je moguće izgubiti neke podatke iz varijabli koje su odbačene. S druge strane, ekstrakcija značajki je formiranje novih varijabli iz postojećih varijabli. Prednost ekstrakcije značajki je ta što čuva većinu strukture izvornog skupa podataka. No, kao nedostatak ističe se to što novostvorene varijable gube svoju interpretaciju.

U nastavku rada prvo će biti predstavljen skup podataka koji će se koristiti za prikaz praktičnih primjera u ovom radu, a nakon toga bit će kroz zasebna poglavlja prikazano nekoliko najpopularnijih tehnika redukcije dimenzija skupa podataka kao što su:

- uključivanje znanja o domeni kako bi se uklonile ili kombinirale kategorije,
- korištenje sažetaka podataka za otkrivanje preklapanja podataka između varijabli (i uklanjanje ili kombiniranje suvišnih varijabli ili kategorija),
- analiza korelacije,
- pretvorba podataka tehnikom pretvaranja kategoričkih varijabli u numeričke varijable,
- analiza glavnih komponenti (PCA),
- ostali poznatiji algoritmi ukratko (MDS, ICA, t-SNE, UMAP) i
- redukcija dimenzija korištenjem regresijskih modela.

2. Skup podataka „edukacija88“

Za prikaz praktičnih primjera korišten je program *RStudio* koji radi s programskim jezikom *R* specijaliziranim za dubinsku analizu podataka. Odabran je skup podataka „edukacija 88“ koji sadrži 77 instanci i temelji se na podacima sakupljenim u sustavu MudRi u okviru kolegija „Programiranje 2“ akademske godine 2013./2014. Podaci su predstavljeni s 15 atributa čije se pojašnjenje nalazi u Tablici 1[1].

Tablica 1: Atributi skupa podataka "edukacija 88"

ID	Atribut	Opis	Raspon vrijednosti
1	id	Identifikator studenta	[1,77]
2	lectures	Ukupan broj bodova ostvaren putem aktivnosti na predavanjima	[0,12]
3	quizzes	Ukupan broj bodova ostvaren na dva kviza	[0,36]
4	labs	Ukupan broj bodova ostvaren na vježbama	[0,56]
5	videos	Ukupan broj gledanja snimki predavanja	[0,51]
6	selfassesm	Ukupan broj klikova u okviru samoprovjera	[0,749]
7	stog	Student pristupio prezentaciji Stog (ne/da)	0/1
8	forum	Student pristupio Forumu uz demonstrature (ne/da)	0/1
9	red	Student pristupio prezentaciji Red (ne/da)	0/1
10	kruzna	Student pristupio prezentaciji Kružna lista (ne/da)	0/1
11	stabla1	Student pristupio prezentaciji Uvod u stabla – 1. dio (ne/da)	0/1
12	dinamicko	Student pristupio prezentaciji Dinamičko programiranje (ne/da)	0/1
13	stabla2	Student pristupio prezentaciji Uvod u stabla – 2. dio (ne/da)	0/1
14	demons	Student pristupio demonstraturama tijekom izvođenja kolegija (ne/da)	0/1
15	grade	Položen predmet (ne/da)	0/1

Prvi korak je u varijablu *data* učitati skup podataka. Nakon toga je uklonjen prvi stupac koji predstavlja atribut *id* jer nije relevantan za daljnju analizu. U originalnom skupu podataka varijabla *grade* imala je vrijednosti „FAIL“ i „PASS“ što je pretvoreno u 0/1 zbog lakše daljnje analize. Na posljetku je prikazano prvih nekoliko redaka skupa podataka.

```
> data<-read.csv("edukacija88.csv", header = TRUE)
> data<-data[,-1]
> data[,14]<-ifelse(data[,14]=='PASS',1,0)
> head(data)
```

	lectures	quizzes	labs	videos	selfassesm	stog	forum	red	kruzna	stabla1	dinamicko	stabla2	demons	grade
1	0	19.33	32.0	0	116	1	1	1	1	1	1	1	1	1
2	5	22.00	27.0	6	232	0	0	0	0	0	0	0	0	1
3	5	15.00	7.0	10	132	0	1	0	1	1	0	0	1	0
4	6	27.66	27.5	13	451	1	1	1	1	1	1	1	1	1
5	3	28.66	0.0	51	260	0	0	0	1	0	0	0	1	0
6	11	32.66	54.0	20	96	0	0	0	0	0	0	0	0	1

Prvi redak predstavlja prvog studenta koji je ostvario 0 bodova putem aktivnosti na predavanjima, 19.33 boda na dva kviza, 32 boda na vježbama, nije gledao snimke predavanja, pristupio je svim prezentacijama i demonstraturama i u konačnici položio kolegij (*grade* = 1). Na isti način moguće je dalje iščitati podatke za svih 77 studenata koji su uključeni u ovaj skup podataka.

3. Praktična razmatranja skup podataka

Iako se pri dubinskoj analizi podataka preferiraju automatizirane metode, važno je u prvom koraku istraživanja podataka osigurati da su odabrane varijable prilagođene za zadatak koji se obavlja. Stoga je vrlo bitno uvidjeti koje su varijable bitnije od drugih za traženi zadatak kroz raspravu s davateljem podataka (ili korisnikom), što će vjerojatno dovesti do boljih rezultata[4].

Praktična razmatranja uključuju pitanja kao što su:

- Koje su varijable najvažnije za dani zadatak, a koje su najvjerojatnije beskorisne?
- Koje varijable mogu sadržavati veliku pogrešku?
- Koje će varijable biti dostupne za mjerenje (i koliko će koštati mjeriti ih u budućnosti) ako se analiza ponovi?
- Koje se varijable mogu mjeriti prije nego što se ishod dogodi?

Već samim pogledom na skup podataka možemo vidjeti potencijalne kandidate, tj. varijable koje neće biti relevantne za daljnju analizu. U korištenom skupu „edukacija 88“ odmah se može zaključiti da varijabla *id* neće biti relevantna jer ona samo predstavlja redni broj studenta upisanog na kolegij i nema nikakve veze s vrijednostima ostalih varijabli i ni na koji način ne utječe na prolazak kolegija. Također, kada se govori o redukciji dimenzija važna neće biti niti varijabla *grade* jer ona predstavlja ciljnu varijablu čija će se vrijednost predviđati odabranom metodom dubinske analize podataka i iz tog razloga nije ju moguće ukloniti iz skupa. One varijable koje već i zdravorazumskim razmišljanjem možemo izdvojiti kao najvažnije su *quizzes* i *labs* jer upravo one nose najveći broj bodova na kolegiju i o njihovoj vrijednosti će na kraju najviše ovisiti prolaz kolegija, tj. vrijednost varijable *grade*.

4. Sažeci skupa podataka

Nakon što je skup podataka praktički razmotren i uočene su određene varijable koje su kandidati za eliminaciju ili ekstrakciju, važan je idući korak upoznavanja podataka i njihovih karakteristika kroz razne sažetke i grafikone. Što se bolje razumiju podaci, to su bolji i rezultati dobiveni dubinskom analizom podataka. Numerički sažeci i grafikoni vrlo su korisni za redukciju dimenzija podataka. Informacije koje se iz njih dobiju mogu pomoći u kombiniranju kategorija kategoričkih varijabli, u odabiru varijabli koje treba ukloniti, u procjeni podataka koji se preklapaju između varijabli i slično.

R ima nekoliko korisnih funkcija koje pomažu u sažimanju podataka. Funkcija *summary()* daje pregled cijelog skupa varijabli u promatranom skupu podataka i za svaku prikazuje najmanju i najveću vrijednost koju poprima, 1. i 3. kvartil vrijednosti te srednju vrijednost i medijan vrijednosti.

```
> summary(data)
    lectures         quizzes           labs           videos         selfassesm         stog        
 Min.   : 0.000   Min.   : 0.00   Min.   : 0.00   Min.   : 0.000   Min.   :  0.0   Min.   :0.0000  
 1st Qu.: 1.000   1st Qu.:15.33   1st Qu.: 6.50   1st Qu.: 1.000   1st Qu.: 93.0   1st Qu.:0.0000  
 Median : 5.000   Median :22.25   Median :18.00   Median : 7.000   Median :182.0   Median :1.0000  
 Mean   : 4.325   Mean   :20.17   Mean   :19.32   Mean   : 9.558   Mean   :218.9   Mean   :0.6494  
 3rd Qu.: 7.000   3rd Qu.:27.66   3rd Qu.:30.00   3rd Qu.:14.000   3rd Qu.:295.0   3rd Qu.:1.0000  
 Max.   :12.000   Max.   :36.00   Max.   :56.00   Max.   :51.000   Max.   :749.0   Max.   :1.0000  
     forum             red             kruzna           stabla1          dinamicko         stabla2      
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
 Median :1.0000   Median :1.0000   Median :1.0000   Median :1.0000   Median :0.0000   Median :1.0000  
 Mean   :0.5455   Mean   :0.6494   Mean   :0.6234   Mean   :0.6234   Mean   :0.4675   Mean   :0.5195  
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000  
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
     demons           grade       
 Min.   :0.0000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:0.0000  
 Median :1.0000   Median :1.0000  
 Mean   :0.5844   Mean   :0.6753  
 3rd Qu.:1.0000   3rd Qu.:1.0000  
 Max.   :1.0000   Max.   :1.0000
```

Funkcije *mean(), sd(), min(), max(), median()* i *length()* su također vrlo korisne za učenje o karakteristikama svake varijable. Prvo nam daju podatke o razmjeru i vrsti vrijednosti koje varijabla poprima. *Min()* i *max()* funkcije mogu se upotrijebiti za otkrivanje ekstremnih vrijednosti koje bi mogle predstavljati pogreške. *Mean()* i *median()* daju osjećaj središnjih vrijednosti te varijable, a veliko odstupanje između njih također ukazuje na nagib. *Sd()* ili standardno odstupanje daje osjećaj koliko su podaci raspršeni u odnosu na srednju vrijednost.

```
> mean(data$quizzes)
[1] 20.16532
> sd(data$quizzes)
[1] 10.61118
> min(data$quizzes)
[1] 0
> max(data$quizzes)
[1] 36
> median(data$quizzes)
[1] 22.25
> length(data$quizzes)
```

```
[1] 77
> sum(is.na(data$quizzes))
[1] 0
```

Prethodnim kodom izračunata je srednja vrijednost, standardna devijacija, minimalna i maksimalna vrijednost, medijan i duljina te broj nedostajućih vrijednosti varijable *quizzes*. Iz dobivenih rezultata može se vidjeti da je srednja vrijednost koju poprima varijabla *quizzes* 20.17, standardna devijacija iznosi 10.61, minimalna vrijednost je 0, dok je maksimalna 36. Medijan iznosi 22.25 te postoji 77 vrijednosti i nema nedostajućih (*null*) vrijednosti.

Sada će se prikazati prethodno izračunate vrijednosti za svaku varijablu u skupu podataka zasebno, ali spojene u *dataframe* kako bi bilo preglednije.

```
> mean=sapply(data, mean)
> sd=sapply(data, sd)
> min=sapply(data, min)
> max=sapply(data, max)
> median=sapply(data, median)
> length=sapply(data, length)
> miss.val=sapply(data, function(x) sum(length(which(is.na(x)))))
> data.frame(mean, median, sd, min, max, length, miss.val)
                   mean median          sd min max length miss.val
lectures      4.3246753   5.00   3.2381019   0  12     77        0
quizzes      20.1653247  22.25  10.6111798   0  36     77        0
labs         19.3181818  18.00  14.9545164   0  56     77        0
videos        9.5584416   7.00  10.3701159   0  51     77        0
selfassesm  218.9090909 182.00 176.5462651   0 749     77        0
stog          0.6493506   1.00   0.4803024   0   1     77        0
forum         0.5454545   1.00   0.5011947   0   1     77        0
red           0.6493506   1.00   0.4803024   0   1     77        0
kruzna        0.6233766   1.00   0.4877165   0   1     77        0
stabla1       0.6233766   1.00   0.4877165   0   1     77        0
dinamicko     0.4675325   0.00   0.5022165   0   1     77        0
stabla2       0.5194805   1.00   0.5028966   0   1     77        0
demons        0.5844156   1.00   0.4960542   0   1     77        0
grade         0.6753247   1.00   0.4713240   0   1     77        0
```

Iz dobivenog rezultata vidljivo je da različite varijable imaju vrlo različit raspon vrijednosti. Drugo opažanje koje je moguće primijetiti je da je srednja vrijednost varijable *selfassesm* mnogo veća od medijana, što ukazuje na desno nakrivljenje. Nijedna od varijabli nema nedostajućih (*null*) vrijednosti. Čini se da također nema naznaka ekstremnih vrijednosti koje bi mogle biti prouzrokovane pogreškama u tipkanju.

Još jedan vrlo koristan pristup istraživanju podataka je agregacija (združivanje) jedne ili više varijabli. Za združivanje jedne varijable možemo koristiti funkciju *table()*.

```
> table(data$grade)

 0  1
25 52
```

Zaključujemo da su predmet položila 52 studenta, dok je 25 studenata palo kolegij.

Funkcija *agregate()* može se koristiti za objedinjavanje jedne ili više varijabli i izračunavanje niza statistika sažetaka.

```
> aggregate(data, by=list(PROLAZ=data$grade), mean)
  PROLAZ lectures  quizzes     labs   videos selfassesm      stog forum       red    kruzna   stabla1
1      0 1.440000  9.60800  2.94000  5.12000     112.64 0.6400000  0.64 0.6400000 0.7200000 0.5600000
2      1 5.711538 25.24096 27.19231 11.69231     270.00 0.6538462  0.50 0.6538462 0.5769231 0.6538462
  dinamicko   stabla2    demons
1 0.4800000 0.3600000 0.6000000
2 0.4615385 0.5961538 0.5769231
```

U ovom primjeru za varijablu *grade* izračunate su srednje ostvarene vrijednosti svih ostalih varijabli u skupu podataka. Tako su studenti koji su pali kolegij (PROLAZ=0) ostvarili prosječno 1.44 bod u sklopu aktivnosti na predavanju, 9.61 bod na oba kviza, 2.94 boda na vježbama i 5 puta su prosječno gledali video te se može nadalje slično iščitati za svaku varijablu. S druge strane, studenti koji su položili kolegij ostvarili su prosječno 5.71 bodova u sklopu aktivnosti na predavanju, 25.24 bodova na oba kviza, 27.19 bodova na vježbama i 12 puta su prosječno pogledali video te analogno za sve ostale varijable.

Sličan koristan skup funkcija su *melt()* i *cast()* u paketu *reshape* koji omogućuju stvaranje okretnih tablica. *Melt()* uzima skup stupaca i slaže ih u jedan stupac. *Cast()* zatim pojedinačni stupac preoblikuje u više stupaca agregiranjem varijabli po našem izboru. Kako bi isprobali ove funkcije prvo je potrebno instalirati i učitati paket *reshape*. Nakon toga je u varijablu *test* spremljeno prvih 5 zapisa iz skupa podataka kako bi se rezultat ovih funkcija mogao lakše uočiti.

```
> install.packages("reshape")
> library(reshape)
> test<-head(data, n=5)
  lectures quizzes labs videos selfassesm stog forum red kruzna stabla1 dinamicko stabla2 demons grade
1        0   19.33 32.0      0        116    1     1   1      1       1         1       1      1     1
2        5   22.00 27.0      6        232    0     0   0      0       0         0       0      0     1
3        5   15.00  7.0     10        132    0     1   0      1       1         0       0      1     0
4        6   27.66 27.5     13        451    1     1   1      1       1         1       1      1     1
5        3   28.66  0.0     51        260    0     0   0      1       0         0       0      1     0
```

Sada je izrađeni podskup podataka *test* funkcijom *melt()* presložen tako da su stupci *labs* i *quizzes* spojeni u kategoričku varijablu *variable* koja poprima vrijednost *labs* ili *quizzes*.

```
> mlt<-melt(test, measure.vars = c("labs","quizzes"))
   lectures videos selfassesm stog forum red kruzna stabla1 dinamicko stabla2 demons grade variable value
1         0      0        116    1     1   1      1       1         1       1      1     1     labs 32.00
2         5      6        232    0     0   0      0       0         0       0      0     1     labs 27.00
3         5     10        132    0     1   0      1       1         0       0      1     0     labs  7.00
4         6     13        451    1     1   1      1       1         1       1      1     1     labs 27.50
5         3     51        260    0     0   0      1       0         0       0      1     0     labs  0.00
6         0      0        116    1     1   1      1       1         1       1      1     1  quizzes 19.33
7         5      6        232    0     0   0      0       0         0       0      0     1  quizzes 22.00
8         5     10        132    0     1   0      1       1         0       0      1     0  quizzes 15.00
9         6     13        451    1     1   1      1       1         1       1      1     1  quizzes 27.66
10        3     51        260    0     0   0      1       0         0       0      1     0  quizzes 28.66
```

Korištenjem funkcije *cast()* sada će se podskup *mlt* preoblikovati tako da će se po varijabli *grade* sumirati kategorije novonastale varijable *variable*.

```
> cst<-cast(mlt,grade~variable, sum)

  grade labs quizzes
1     0  7.0   43.66
2     1 86.5   68.99
```

Kao rezultat dobivena je tablica u kojoj je vidljivo da su od prvih 5 studenata iz skupa podataka „edukacija 88“ studenti koji su pali kolegij (*grade*=0) ukupno ostvarili 7 bodova iz vježbi i 43.66 boda iz dva kviza, dok su oni studenti od prvih 5 koji su položili kolegij (*grade*=1) ostvarili ukupno 86.5 bodova iz vježbi i 68.99 bodova iz dva kviza.

Kroz ovo poglavlje predstavljene su razne metode dobivanja sažetaka skupa podataka. Kako je i ranije spomenuto, vrlo je važno na ovaj način upoznati skup podataka nad kojim će se raditi daljnja analiza. Već ovim relativno jednostavnim metodama moguće je otkriti one varijable koje će biti beskorisne u daljnjoj analizi čime se povećava točnost dobivenih rezultata.

5. Analiza korelacije

Korelacija ili bivarijatni odnos opisuje odnos (povezanost) između neke dvije varijable. U skupovima podataka s velikim brojem varijabli (koji će vjerojatno poslužiti kao prediktori) obično se dosta preklapaju informacije obuhvaćene skupom varijabli. Jedan jednostavan način pronalaska nepotrebnih varijabli je analiza korelacije. Parovi varijabli koji imaju vrlo jaku (pozitivnu ili negativnu) korelaciju sadrže dosta preklapanja informacija i dobri su kandidati za redukciju podataka uklanjanjem jedne od varijabli. Uklanjanje varijabli koje su u korelaciji s drugima korisno je za izbjegavanje problema s multikolinearnošću[1] koji se mogu pojaviti u raznim modelima. Analiza korelacije također je dobra metoda za otkrivanje duplikacija varijabli u podacima. Ponekad se ista varijabla pojavljuje slučajno više puta u skupu podataka (pod drugim nazivom) jer je skup podataka spojen iz više izvora ili se isti fenomen mjeri u različitim jedinicama i slično. Korelacija se uvijek nalazi na intervalu između -1 i +1. Negativna korelacija označava da kako vrijednost jedne varijable raste, tako druge pada, dok pozitivna korelacija označava da će porastom jedne varijable rasti i druga. Postoje dvije osnovne metode za izračunavanje korelacije između dvije varijable: Pearsonova korelacija (parametarska korelacija) i Spearmanova korelacija (neparametarska korelacija). *R* prema zadanim postavkama korelaciju izračunava pomoću Pearsonove metode.

5.1. Pearsonova korelacija

Pearsonova korelacija je parametarska analiza koja zahtijeva da odnos između varijabli bude linearan i da podaci budu bivarijatno normalno distribuirani[2]. Varijable bi trebale biti numeričke ili u nekom intervalu. Ova vrsta korelacije je osjetljiva na *outliere*[3]. Izračunava se prema formuli:

$$r = \frac{Cov(x, y)}{\sigma_x \sigma_y}$$

gdje je:

- $Cov(x, y)$, kovarijanca[4] između varijabli *x* i *y*,
- $\sigma_x = \sum(x - \bar{x})^2$, standardna devijacija od *x* i
- $\sigma_y = \sum(y - \bar{y})^2$, standardna devijacija od *y*.

[1] Multikolinearnost je prisutnost dva ili više prediktora koji dijele isti linearni odnos s izlaznom varijablom.
[2] Bivarijatna normalna distribucija sastoji se od dvije neovisne slučajne varijable: dvije varijable su u bivarijatnom normalu kada su obje normalno raspodijeljene i imaju normalnu raspodjelu kada se zbroje.
[3] *Outlier* je točka opažanja koja je udaljenija od ostalih točaka; obično nastaje zbog pogreške u mjerenju, a može značajno utjecati na točnost modela.
[4] Kovarijanca je mjera koja pokazuje koliko se dvije varijable mijenjaju zajedno. Kovarijanca postaje sve više pozitivnom za svaki par vrijednosti koji se razlikuje od njihovih srednjih vrijednosti u istom smjeru (zajedničkom pozitivnom ili negativnom odstupanju od svojih aritmetičkih sredina), te postaje više negativna za svaki par vrijednosti koji se razlikuje od njihovih srednjih vrijednosti u suprotnim smjerovima.

Koeficijent korelacije, *r*, može se kretati od +1 do -1, pri čemu je +1 savršena pozitivna korelacija, a -1 savršena negativna korelacija. Ako je koeficijent korelacije jednak 0, to označava da ne postoji nikakva korelacija između dvije varijable.

```
> cor.test( ~ labs+grade, data=data, method = "pearson")
Pearson's product-moment correlation
data:  labs and grade
t = 10.266, df = 75, p-value = 6.051e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6520102 0.8438710
sample estimates:
     cor
0.764364
```

Kao primjer izračunata je korelacija između varijabli *labs* i *grade*. Koeficijent korelacije *r*, u primjeru predstavljen s *cor*, iznosi 0.76 i predstavlja značajnu korelaciju između dvije varijable.

5.2. Spearmanova korelacija

Spearmanova korelacija smatra se neparametarskom analizom. Ova vrsta korelacije razvrstava vrijednosti varijable prema rangu i izračunava razinu sličnosti između rangova. Spearmanova korelacija ima prednost u tome što je robusna na *outliere* i nije povezana s distribucijom podataka. Izračunava se prema formuli:

$$\rho = \frac{Cov(rg_x, rg_y)}{\sigma_{rg_x}\sigma_{rg_y}}$$

Koeficijent korelacije, ρ (*rho)*, može se kretati od +1 do -1, pri čemu je +1 savršena pozitivna korelacija, a –1 je savršena negativna korelacija. ρ od 0 ne predstavlja nikakvu korelaciju.

```
> cor.test( ~ labs+grade, data=data, method = "spearman")

        Spearman's rank correlation rho

data:  labs and grade
S = 15657, p-value < 2.2e-16
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.7941903
```

Kao i u prethodnom primjeru, izračunata je korelacija između varijabli *labs* i *grade*. Koeficijent korelacije *rho*, iznosi 0.79 te slično kao i Pearsonova korelacija, predstavlja značajnu korelaciju između dvije varijable.

5.3. Matrica korelacije

Bivarijatna korelacija je dobar početak upoznavanja odnosa među varijablama, ali s multivarijatnom analizom moguće dobiti širu sliku. Matrica korelacije je matrica koja prikazuje parnu korelaciju svih varijabli u promatranom skupu podataka. *R* funkcija *cor()* vraća matricu korelacije. Jedina razlika s bivarijatnom korelacijom je ta što nije potrebno navesti za koje varijable želimo izračunati korelaciju jer *R* izračunava korelaciju između svih varijabli prema zadanim postavkama. Međutim, treba imati na umu da se korelacija ne može izračunati za faktorsku varijablu i sve takve varijable treba izbaciti iz skupa podataka prije prosljeđivanja funkciji *cor()*. Matrica korelacije je simetrična, što znači da vrijednosti iznad dijagonale imaju iste vrijednosti kao i one ispod te je za preglednost bolje prikazati samo polovicu matrice.

```
> as.dist(round(cor(data),2))
```

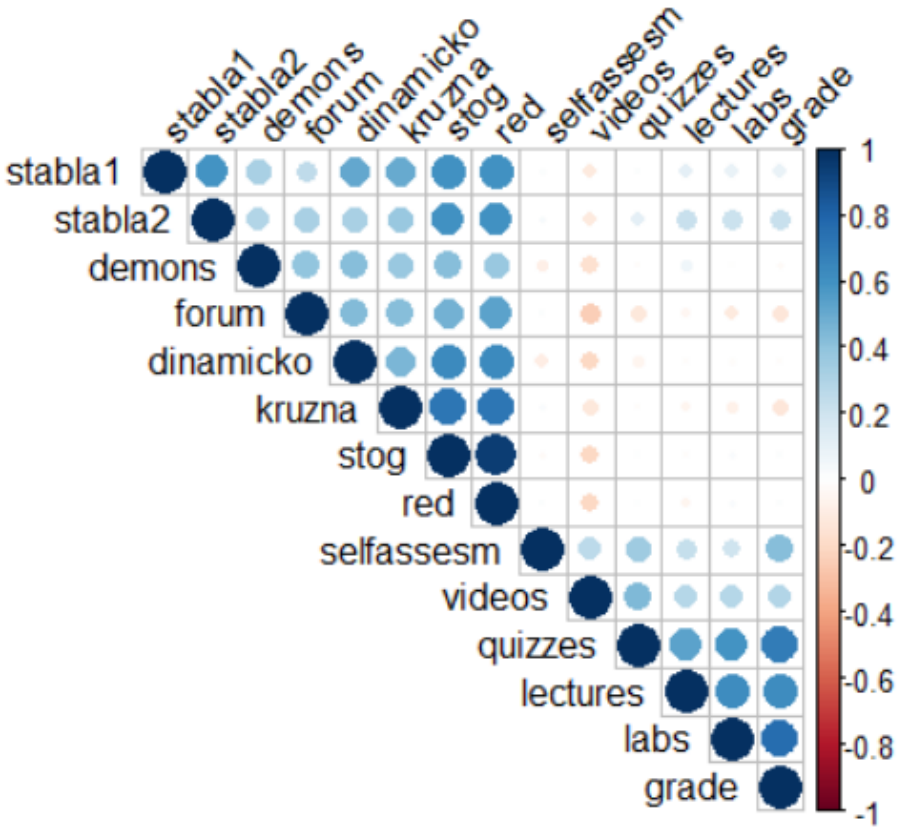
```
           lectures quizzes  labs videos selfassesm  stog forum   red kruzna stabla1 dinamicko stabla2 demons
quizzes        0.53
labs           0.63    0.59
videos         0.28    0.45  0.29
selfassesm     0.24    0.36  0.21   0.26
stog          -0.01    0.01  0.03  -0.20      -0.02
forum         -0.05   -0.13 -0.11  -0.24       0.01  0.48
red           -0.05    0.00  0.02  -0.21       0.02  0.94  0.53
kruzna        -0.05   -0.02 -0.07  -0.13       0.02  0.72  0.42  0.72
stabla1        0.11    0.02  0.10  -0.11       0.02  0.61  0.26  0.61   0.50
dinamicko     -0.01   -0.06 -0.02  -0.20      -0.10  0.63  0.44  0.63   0.46    0.51
stabla2        0.23    0.11  0.21  -0.11       0.03  0.60  0.32  0.60   0.38    0.59      0.33
demons         0.06   -0.01 -0.01  -0.16      -0.08  0.43  0.39  0.37   0.38    0.32      0.42    0.30
grade          0.62    0.69  0.76   0.30       0.42  0.01 -0.13  0.01  -0.14    0.09     -0.02    0.22  -0.02
```

Prethodni dio koda prikazuje matricu korelacije odabranog skupa podataka koja je zbog bolje preglednosti zaokružena na dvije decimale funkcijom *round (,2)* i prikazane su vrijednosti samo ispod glavne dijagonale funkcijom *as.dist()*.

Kako bi grafički prikazali matricu korelacije iz koje će lakše biti uočiti korelacije, moguće je koristiti biblioteku *corrplot*:

```
> install.packages("corrplot")
> library(corrplot)
> corrplot(cor(data), type = "upper", order = "hclust", tl.col = "black", t
l.srt = 45)
```

Nakon instalacije i učitavanja biblioteke, pomoću naredbe *corrplot()* iscrtava se grafički prikaz matrice korelacije prema danim parametrima. Dobiveni graf prikazan je na Slici 3.



Slika 3: Graf matrice korelacije

Iz dobivenih rezultata vidljivo je da je većina korelacija niska te da postoje i negativne i pozitivne korelacije. Kao zaključak može se izvesti da najveći utjecaj na prolazak kolegija, tj. najveću povezanost s varijablom *grade* imaju varijable *quizzes* i *labs*. To je moglo biti zaključeno i detaljnijim logičkim promišljanjem jer broj bodova ostvaren na oba kviza i na vježbama nose najviše bodova na kolegiju pa najviše i utječu na konačnu ocjenu.

Kada se koeficijenti korelacije gledaju kroz domenu redukcije dimenzija, treba uzeti u obzir one varijable koje imaju malu ili gotovo nikakvu korelaciju s ciljnom varijablom, a veliku korelaciju s ostalim varijablama prediktorima. U tom kontekstu veliku korelaciju s nekima od varijabli prediktora, a nikakvu korelaciju s varijablom *grade* imaju varijable *stog* i *red* te varijabla *stog* s varijablama *kruzna* i *dinamicko* kao i varijabla *red* također sa *kruzna* i *dinamicko*. Taj zaključak implicira da se jedna od varijabli *stog* i *red* može ukloniti iz modela predikcije i time će se smanjiti dimenzija skupa podataka, a neće značajnije utjecati na rezultate dubinske analize podataka.

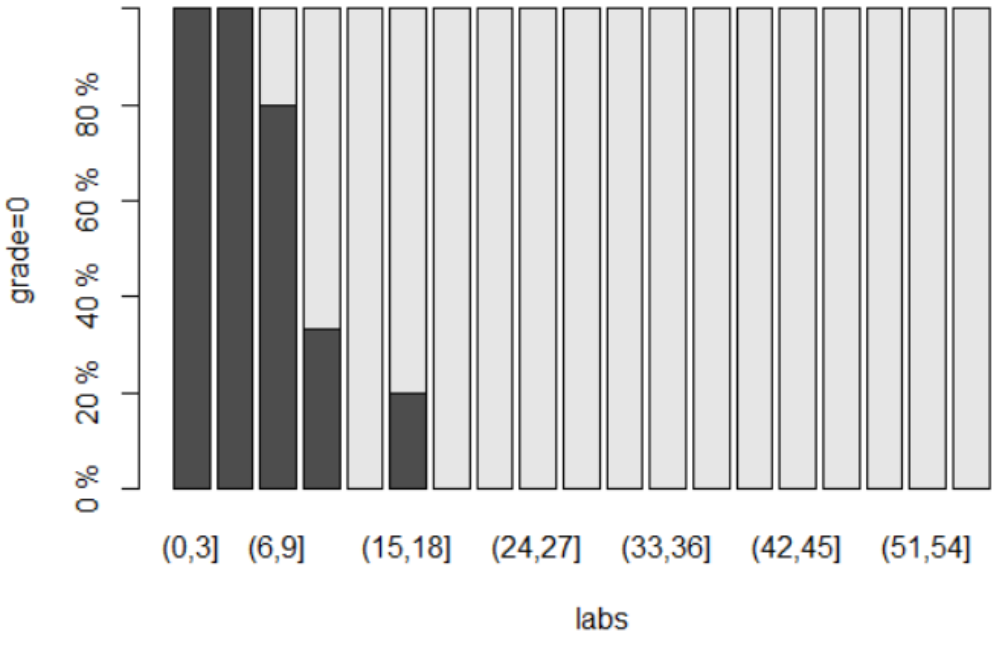
6. Smanjenje broja kategorija u kategoričkim varijablama

Kada kategorička varijabla ima mnogo kategorija, a odabrana je kao prediktor, mnoge metode dubinske analize podataka zahtijevat će njeno pretvaranje u mnoge izvedene (engl. *dummy*) varijable. Konkretno, varijabla s *m* kategorija transformirat će se u *m* ili *m-1* izvedenih varijabli (ovisno o metodi). To znači da čak i ako postoji vrlo malo izvornih kategoričkih varijabli, one mogu uvelike povećati dimenziju skupa podataka. Jedan od načina da se to riješi jest smanjiti broj kategorija kategoričkih varijabli kombiniranjem bliskih ili sličnih kategorija. Kombiniranje kategorija zahtijeva uključivanje stručnog znanja i zdravog razuma.

Okretne tablice (engl. *Pivot tables*) korisne su za ovaj zadatak. Mogu se ispitati veličine različitih kategorija i način ponašanja varijable ishoda u svakoj kategoriji. Općenito, kategorije koje sadrže vrlo malo opažanja dobri su kandidati za kombiniranje s drugim kategorijama. Treba koristiti samo one kategorije koje su najrelevantnije za analizu, a druge označiti kao „ostale“. U zadacima klasifikacije (s kategoričkom varijablom ishoda), okretna tablica razgrađena po razredima ishoda može pomoći u prepoznavanju kategorija koje ne razdvajaju klase. Tada su i te kategorije su kandidati za uključivanje u kategoriju „ostali“.

U sljedećem primjeru prikazana je raspodjela izlazne varijable *grade* s obzirom na varijablu *labs* koja pretvorena u kategoričku tako da je podijeljena u segmente veličine 3. Konkretno, izrađena je pivot tablica koja je zbog lakše interpretacije prikazana grafom.

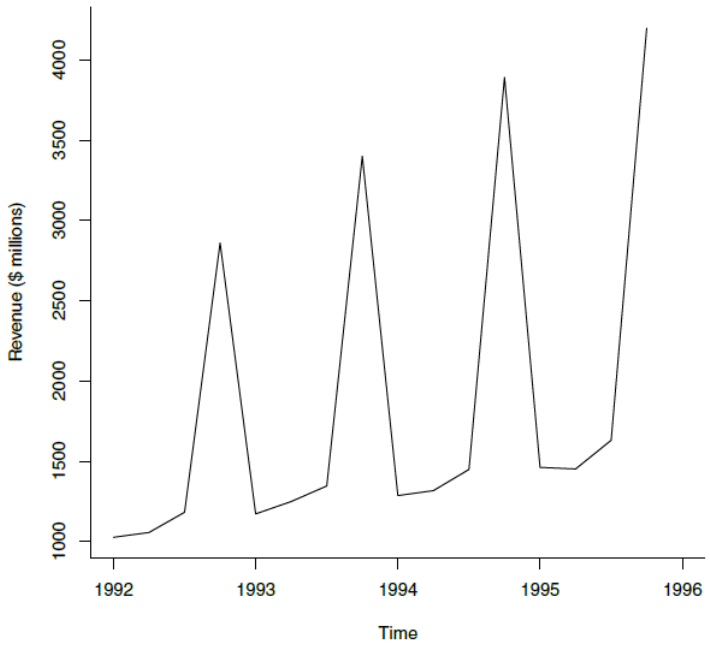
```
> kategoricka_labs<-cut(data$labs, seq(0,60,3))
> tbl <- table(data$grade, kategoricka_labs)
> prop.tbl <- prop.table(tbl, margin=2)
> barplot(prop.tbl, xlab="labs", ylab="grade=0", yaxt="n")
> axis(2, at=(seq(0,1, 0.2)), paste(seq(0,100,20), "%"))
```



Slika 4: Graf raspodjele prolaska kolegija s obzirom na broj bodova ostvaren na vježbama

U grafu na Slici 4 crna linija na „šipci“ označava da je varijabla *grade* = 0, tj. pad kolegija. Može se iščitati da su svi studenti koji su iz vježbi ostvarili manje od 7 bodova, 80% onih koji su ostvarili 7-9 bodova, 35% onih koji su ostvarili 10-12 bodova i 20% onih koji su ostvarili između 16 i 18 bodova pali kolegij. Slične razine linije na „šipci“ za pojedinu kategoriju označavaju sličnu raspodjelu te se takve kategorije tada mogu kombinirati u jednu kategoriju. Na ovom primjeru prigodne za spajanje bile bi sve kategorije koje sadržavaju broj bodova veći od 18. Dakle, u slučaju da je *labs* kategorička varijabla mogli bismo raspon vrijednosti svesti na sedam kategorija: (0,3], (3,6], (6,9], (9,12], (12,15], (15,18] i „ostali“.

U kontekstu vremenske serije u kojoj bismo mogli imati kategoričku varijablu koja označava sezonu (poput mjeseca ili sati u danu) i koja će poslužiti kao prediktor, smanjenje broja kategorija može se izvršiti ispitivanjem grafikona vremenske serije i identificiranjem sličnih razdoblja. Na primjer, vremenska crta na Slici 5 prikazuje godišnje prihode igračke „R“ od 1992. do 1996. godine. Samo se četvrti kvartal razlikuje, pa stoga možemo kombinirati kvartale 1 - 3 u jednu kategoriju.



Slika 5: Godišnji prihodi igračke „R“ od 1992. do 1996. godine[4]

Ponekad kategorije u kategoričkoj varijabli predstavljaju intervale. Česti su primjeri dobne skupine ili dohotka. Ako su vrijednosti intervala poznate (na primjer, kategorija 3 je plaća između 5000 – 7000 kn), kategoričku vrijednost (kategorija 3) može se zamijeniti srednjom vrijednosti intervala (u ovom primjeru 6000 kn). Rezultat će biti numerička varijabla koja više ne zahtijeva razdvajanje na više izvedenih varijabli.

7. Analiza glavnih komponenti (PCA)

Analiza glavnih komponenti (engl. *Principal Components Analysis - PCA*) osmišljena je 1901. godine od strane Karla Pearsona, a predstavlja metodu za redukciju dimenzija skupa podataka, posebno kada je broj varijabli velik. Smanjivanje broja varijabli skupa podataka prirodno donosi smanjenje točnosti, ali ideja redukcije dimenzija je „žrtvovanje“ malo točnosti za jednostavnost budući da je manje varijabli lakše istražiti i vizualizirati i time analizu podataka učiniti mnogo lakšom i bržom za obradu[8]. PCA je posebno značajna kada imamo podskupove mjerenja koja se mjere na istoj skali i visoko su povezana. Ovom metodom dobivaju se ponderirane linearne kombinacije izvornih varijabli koje zadržavaju većinu informacija sadržanih u izvornom skupu[4]. Moguće je očekivati da će 20-ak i više varijabli biti obuhvaćeno s 2 ili 3 glavne komponente koje nisu međusobno u korelaciji. PCA je namijenjena za upotrebu s numeričkim varijablama i nije pogodna za rad s kategoričkim varijablama.

Koraci po kojima se izvodi analiza glavnih komponenti:

1. Standardizacija originalnih podataka
2. Računanje matrice kovarijanci *C*
3. Računanje svojstvenih vrijednosti $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_p$ te odgovarajućih svojstvenih vektora $a_1, a_2, a_3, \ldots, a_p$
4. Eliminacija varijabli koje sadrže mali udio varijacija podataka.

Cilj analize je uzeti *n* varijabli X_1, X_2, ..., X_n i pronaći kombinaciju kako bi se izračunale nove varijable Z_1, Z_2, ..., Z_n koje će opisivati varijacije podataka i koje međusobno nisu u korelaciji. Nove varijable Z predstavljaju glavne komponente:

$$Z_i = a_{i,1}(X_1 - \overline{X}_1) + a_{i,2}(X_2 - \overline{X}_2) + \ \ldots + a_{i,n}(X_1 - \overline{X}_n), \qquad \mathrm{i} = 1, \ldots, \mathrm{n}$$

Varijance[5] novih varijabli su poredane u padajući niz $\mathrm{Var}(Z_1) \geq \mathrm{Var}(Z_2) \geq \ldots \geq \mathrm{Var}(Z_n)$. Prednost glavnih komponenti u odnosu na izvorne podatke jest ta što su nekorelirane (koeficijent korelacije = 0), što znači da ako se regresijski modeli konstruiraju koristeći ove glavne komponente kao prediktore, neće nastati problemi s multikolinearnosti.

7.1. Standardizacija podataka

Prvi korak provedbe PCA jest standardizacija podataka. Cilj je standardizirati raspon kontinuiranih početnih varijabli tako da svaka od njih jednako doprinosi analizi. Preciznije, razlog zašto je presudno izvršiti standardizaciju prije PCA je taj što je potonja prilično osjetljiva na varijance početnih varijabli, tj. ako postoje velike razlike između raspona početnih varijabli,

[5] Varijanca je mjera disperzije mjerenih ili slučajnih veličina; prosječna suma kvadrata odstupanja vrijednosti obilježja (veličine) od aritmetičke sredine. (Hrvatska enciklopedija, www.enciklopedija.hr)

one varijable s većim rasponima dominirat će nad onima s malim rasponima (na primjer, u skupu podataka „edukacija88“ varijabla *selfassesm* koja se kreće između 0 i 749 dominirat će nad varijablom *quizzes* koja se kreće između 0 i 36), što će dovesti do pristranih rezultata. Dakle, transformacija podataka u usporedive ljestvice može spriječiti ovaj problem.

Matematički se za svaku vrijednost svake varijable to može učiniti oduzimanjem srednje vrijednosti i dijeljenjem sa standardnim odstupanjem:

$$z = \frac{x - \mu}{\sigma}$$

gdje je:

- x, pojedinačna vrijednost varijable,
- μ, srednja vrijednost varijable i
- σ, standardna devijacija varijable.

7.2. Izračun matrice kovarijanci

Drugi korak predstavlja izračun matrice kovarijanci, a cilj ovog koraka je razumjeti kako se varijable skupa ulaznih podataka razlikuju od srednje vrijednosti u odnosu jedna na drugu ili drugim riječima kako bi se utvrdilo postoji li međusobna veza. Matrica kovarijanci usko je povezana s matricom korelacije (o čemu je bilo riječi u poglavlju 5.3). Obje matrice mjere odnos i ovisnost dviju varijabli. Kovarijanca označava smjer linearnog odnosa između varijabli, a korelacija mjeri i snagu i smjer linearnog odnosa između dvije varijable. Korelacija je, u biti, funkcija kovarijance. Ono što ih razlikuje jest činjenica da su korelacijske vrijednosti standardizirane, dok vrijednosti kovarijance nisu.

Matrica kovarijanci je $p \times p$ simetrična matrica (gdje je p broj dimenzija) koja prikazuje kovarijance svih parova početnih varijabli. Na primjer, za trodimenzionalni skup podataka s 3 varijable *x*, *y* i *z*, matrica kovarijanci je 3x3 matrica:

$$\begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z) \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

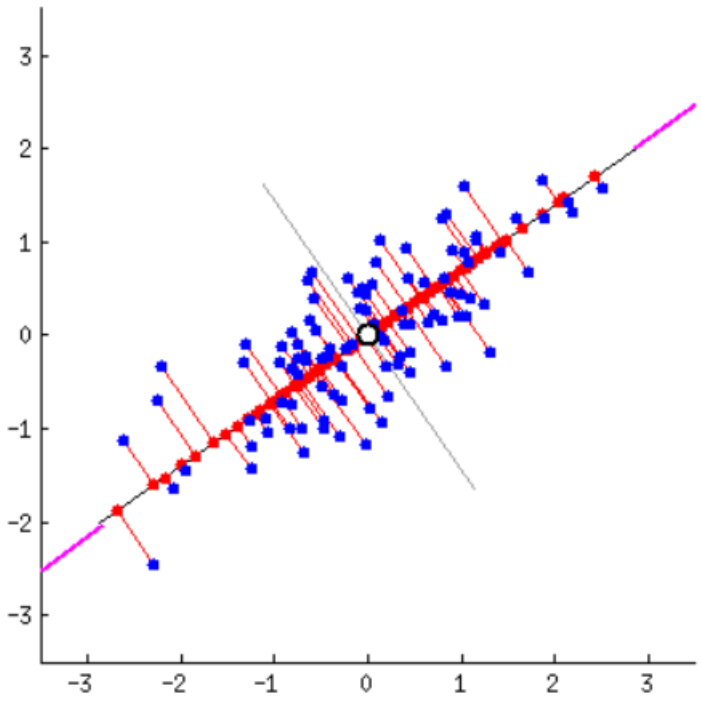
7.3. Izračun svojstvenih vrijednosti i svojstvenih vektora

Svojstvene vrijednosti (engl. *eigenvalues*) i svojstveni vektori (engl. *eigenvectors*) pojmovi su linearne algebre koji se izračunavaju iz matrice kovarijanci da bi se odredile glavne komponente podataka. Glavne komponente su nove varijable koje se grade kao linearne kombinacije početnih varijabli. Te se kombinacije izvode na takav način da su nove varijable, tj. glavne komponente nekorelirane, a većina informacija koje sadrže početne varijable se komprimira u prve komponente. Dakle, ideja je da 10-dimenzionalni podaci daju 10 glavnih komponenti, ali PCA pokušava staviti maksimalno informacija u prvu komponentu, zatim

maksimalno preostale podatke u drugu i tako dalje. Organiziranje podataka u glavnim komponentama na ovaj način omogućit će smanjenje dimenzije bez gubitka puno informacija odbacivanjem komponenti s malo podataka i uzimanjem preostalih komponenti kao novih varijabli. Važno je ovdje shvatiti da su glavne komponente manje interpretabilne i nemaju nikakvo stvarno značenje jer su građene kao linearne kombinacije početnih varijabli.

Geometrijski gledano, glavne komponente predstavljaju pravce podataka koji objašnjavaju maksimalnu količinu varijanci, odnosno linije koje obuhvaćaju većinu podataka. Odnos između varijance i informacija ovdje je da, što je veća varijanca koju nosi linija, veća je disperzija podatkovnih točaka duž nje i nosi više informacija.

Na primjer, pretpostavimo da je shema rasipanja nekog skupa podataka prikazana na Slici 6. Prva glavna komponenta je linija koja se podudara s ljubičastim oznakama jer prolazi kroz ishodište i to je linija u kojoj je projekcija točaka (crvene točke) najviše raširena. Ili matematički gledano, to je linija koja maksimizira varijancu (prosjek kvadrata udaljenosti od projiciranih točaka (crvene točke) do originalnih točaka).



Slika 6: Shema rasipanja skupa podataka[8]

Druga glavna komponenta izračunava se na isti način, uz uvjet da je okomita na prvu glavnu komponentu i da ima sljedeću najveću varijancu. Postupak se nastavlja dok se ne izračuna ukupan broj glavnih komponenti, jednak izvornom broju varijabli.

Svojstveni vektori i svojstvene vrijednosti uvijek dolaze u parovima, tako da svaki svojstveni vektor ima svojstvenu vrijednost, a njihov je broj jednak broju dimenzija podataka.

Na primjer, za trodimenzionalni skup podataka postoje 3 varijable, dakle postoje 3 svojstvena vektora s 3 odgovarajuće svojstvene vrijednosti.

Svojstveni vektori matrice kovarijance zapravo su smjerovi osi na kojima je najviše varijance (najviše informacija) i njih nazivamo glavnim komponentama, a svojstvene vrijednosti jednostavno su koeficijenti svojstvenih vektora, koji označuju količinu varijance koju nosi svaka glavna komponenta. Slaganjem svojstvenih vektora po redoslijedu njihovih vrijednosti od najvećih do najnižih dobivaju se glavne komponente prema značajnosti.

Na primjer, pretpostavimo skup podataka s dvije varijable *x* i *y* te da su svojstveni vektori i svojstvene vrijednosti matrice kovarijance sljedeće:

$$v1 = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix}, \qquad \lambda_1 = 1.284028$$

$$v2 = \begin{bmatrix} -0.7351785 \\ 0.6778736 \end{bmatrix}, \qquad \lambda_2 = 0.049083$$

Ako se svojstvene vrijednosti rangiraju u silaznom redoslijedu, dobit će se $\lambda_1 > \lambda_2$, što znači da je svojstveni vektor koji odgovara prvoj glavnoj komponenti (PC1) *v1*, a onaj koji odgovara drugoj komponenti (PC2) je *v2*.

Nakon određivanja glavnih komponenti, kako bi se izračunao postotak varijance (informacije) koji se izračunava za svaku komponentu, potrebno je podijeliti svojstvenu vrijednost svake komponente sa zbrojem svojstvenih vrijednosti. Primijeni li se ovo na primjer iznad, zaključuje se da PC1 nosi 96%, a PC2 samo 4% varijance podataka.

7.4. Vektor značajki

Izračunavanje svojstvenih vektora i njihovo određivanje po svojstvenim vrijednostima silaznim redoslijedom omogućava pronalaženje glavnih komponenti prema značajnosti. U idućem koraku odlučuje se treba li zadržati sve te komponente ili odbaciti one manjeg značaja (niskih svojstvenih vrijednosti) i s preostalim formirati matricu vektora koja se naziva vektor značajki.

Dakle, vektor značajki je jednostavno matrica koja kao stupce sadrži svojstvene vektore komponenti koje su odlučene da će biti zadržane. To ga čini prvim korakom prema redukciji dimenzija, jer ako se odluči zadržati samo *p* svojstvenih vektora (komponenti) od *n*, konačni skup podataka imat će samo *p* dimenzija.

Nastavno na primjer iz prethodnog koraka, može se stvoriti vektor značajki s oba svojstvena vektora *v1* i *v2*:

$$\begin{bmatrix} 0.6778736 & -0.7351785 \\ 0.7351785 & 0.6778736 \end{bmatrix}$$

Ili odbaciti svojstveni vektor *v2*, koji je manje važan, i oblikovati vektor značajki samo sa svojstvenim vektorom *v1*:

$$v1 = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix}$$

Odbacivanjem svojstvenog vektora *v2* smanjuje se dimenzija za 1 i, posljedično, uzrokuje gubitak informacija u konačnom skupu podataka. Ali s obzirom na to da je *v2* nosio samo 4% informacija, gubitak neće biti značajan i još uvijek će biti dostupno 96% informacija koje prenosi *v1*.

Zaključno na primjer, na osobi koja izvodi dubinsku analizu podataka je da odabere hoćete li zadržati sve komponente ili odbaciti one manjeg značenja, ovisno o tome što pojedini zadatak traži[8].

7.5. Reorganizacija početnog skupa podataka

U prethodnim koracima, osim standardizacije, nisu se radile nikakve promjene na početnim podacima. Samo su odabrane glavne komponente i oblikovan je vektor značajki, ali skup ulaznih podataka ostao je isti u odnosu na izvorne osi, tj. u odnosu na početne varijable.

U posljednjem koraku cilj je koristiti vektor značajki formiran pomoću svojstvenih vektora matrice kovarijance za „preusmjeravanje“ podataka s izvornih osi na one predstavljene glavnim komponentama (otuda i dolazi naziv analiza glavnih komponenti). To se može postići množenjem transponiranog izvornog skupa podataka s transponiranim vektorom značajki.

7.6. Analiza glavnih komponenti na skupu podataka „edukacija88“

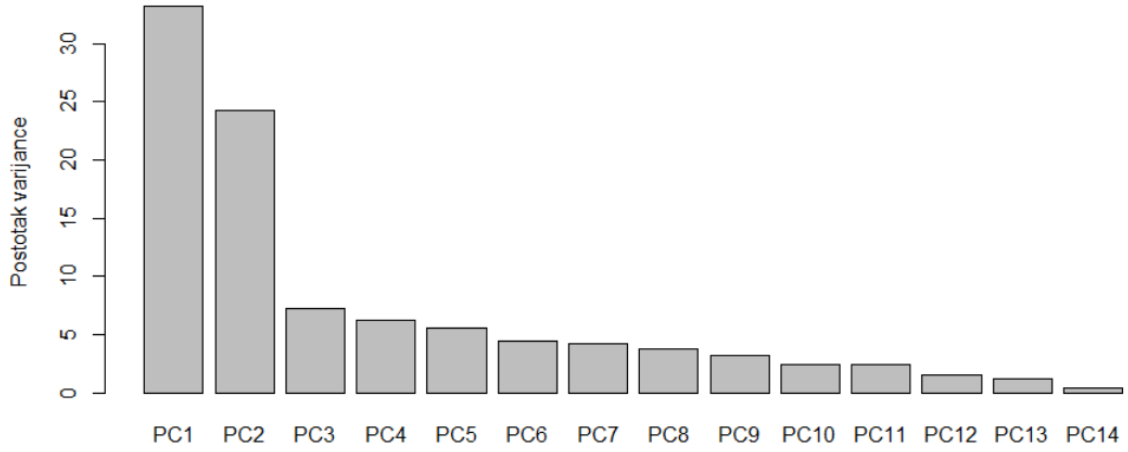
Kako bi se prikazao praktični primjer provođenja analize glavnih komponenti, bit će korišten skup podataka „edukacija88“. Kako je opisano u prethodnim potpoglavljima, prvo je napravljena standardizacija skupa podataka funkcijom *scale()* te su na tom standardiziranom skupu konstruirane glavne komponente funkcijom *prcomp()* i prikazan je sažetak dobivenih rezultata:

```
> data_std <- scale(data)
> pcs <- prcomp(data_std)
> summary(pcs)
```

```
Importance of components:
                          PC1    PC2     PC3     PC4     PC5    PC6     PC7     PC8     PC9    PC10    PC11
Standard deviation     2.1553 1.8436 1.00420 0.93458 0.88520 0.7857 0.76515 0.72993 0.67108 0.58385 0.57560
Proportion of Variance 0.3318 0.2428 0.07203 0.06239 0.05597 0.0441 0.04182 0.03806 0.03217 0.02435 0.02367
Cumulative Proportion  0.3318 0.5746 0.64662 0.70901 0.76498 0.8091 0.85089 0.88895 0.92112 0.94547 0.96913
                          PC12    PC13   PC14
Standard deviation     0.46554 0.40792 0.2214
Proportion of Variance 0.01548 0.01189 0.0035
Cumulative Proportion  0.98461 0.99650 1.0000
```

Dodatno, grafički je prikazan postotak varijance koji nosi pojedina glavna komponenta pomoću *scrre plota* (Slika 7). *Scree plot* pokazuje koliki postotak varijance svaka glavna komponenta bilježi iz podataka. Ova vrsta grafa koristi se za odabir glavnih komponenti koje će se zadržati. Idealna krivulja bi trebala biti strma u početku, a zatim se saviti „u laktu“ - to je točka rezanja i nakon toga se ravna. Odabrane glavne komponente trebale bi moći opisati barem 80% varijance. Ako su za ispunjenje tog uvjeta potrebne više od 3 glavne komponente, PCA možda neće biti najbolji način za redukciju dimenzija odabranog skupa podataka[9].

```
> y <- pcs$sdev^2 / sum(pcs$sdev^2) * 100
> barplot(y, names.arg = c("PC1", "PC2", "PC3", "PC4", "PC5", "PC6", "PC7",
"PC8", "PC9", "PC10", "PC11", "PC12", "PC13", "PC14"),
+         ylab = "Postotak varijance", xlab = "")
```



Slika 7: Scree plot glavnih komponenti

Iz dobivenih komponenti, gledajući postotak varijance koji nosi određena glavna komponenta, zaključuje se da je potrebno šest glavnih komponenti za više od 80% ukupne varijabilnosti. Prve tri glavne komponente čine samo 64% ukupne varijabilnosti, a samim tim smanjivanje broja varijabli na tri značilo bi gubljenje mnogo informacija.

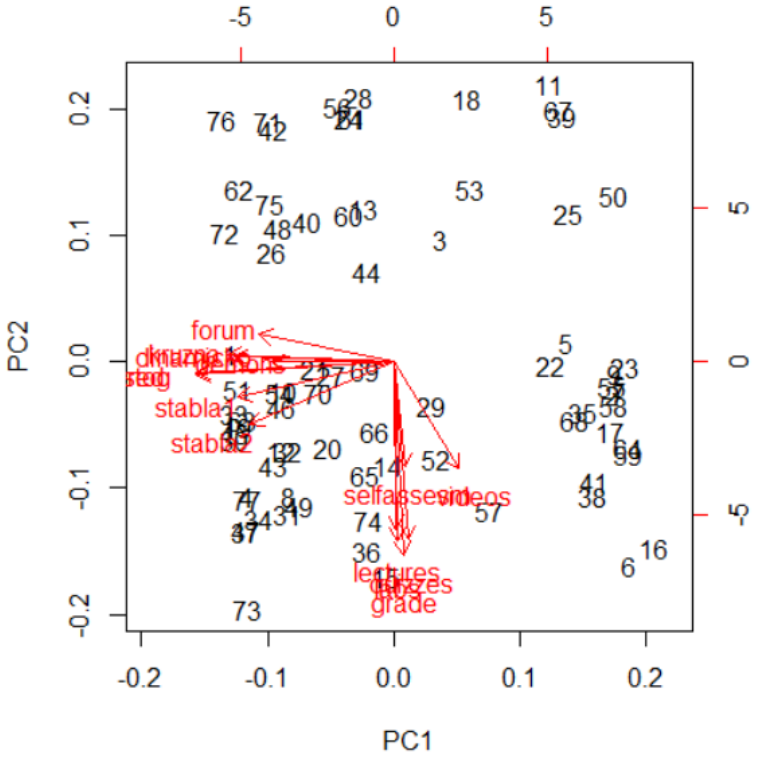
Idućim dijelom koda prikazan je utjecaj pojedinih varijabli na prvih pet glavnih komponenti zaokružen na 10 decimala:

```
> round(pcs$rot[,1:5],10)
                  PC1         PC2         PC3         PC4         PC5
lectures    0.005541316 -0.421070988  0.27383802 -0.14386599  0.009551422
quizzes     0.032000260 -0.447151597 -0.11699514 -0.07868552 -0.177939837
labs        0.008713282 -0.452250124  0.28512281  0.01301031  0.015597418
videos      0.140001471 -0.268194677 -0.40921875  0.07344644 -0.653503512
selfassesm  0.024387104 -0.264837678 -0.66206175 -0.15537414  0.487375446
stog       -0.426148378 -0.032690244 -0.08984858  0.13880281 -0.049685199
forum      -0.292497590  0.068070514 -0.10474804 -0.53350923  0.238410915
red        -0.426585415 -0.027187138 -0.14350862  0.14421988  0.023139245
kruzna     -0.358301425  0.019122262 -0.29132503  0.06305472 -0.198040983
stabla1    -0.338511749 -0.089254072  0.07322481  0.37077897 -0.018495907
dinamicko  -0.344029812  0.012967467  0.07238546 -0.12567242 -0.182332796
```

```
stabla2     -0.312433662 -0.158866968  0.19187309  0.30170803  0.239367963
demons      -0.264956379 -0.000499129  0.20309615 -0.60789167 -0.281947618
grade        0.019962806 -0.485774291  0.10521094 -0.03003640  0.184277976
```

Grafički je prethodno dobivene rezultate moguće prikazati *biplot* grafom. PCA *biplot* prikazuje opterećenja varijabli crvenim vektorima. Što su ti vektori dalje od izvorišta glavne komponente, to imaju veći utjecaj na glavnu komponentu. Međusobni položaj vektora također nagovještava kako varijable koreliraju jedna s drugom: mali kut među njima podrazumijeva pozitivnu korelaciju, veliki sugerira negativnu korelaciju, a kut od 90° ukazuje da nema korelacije između dviju varijabli. Na Slici 8 prikazan je *biplot* graf za prve dvije glavne komponente dobiven funkcijom *biplot()*.

```
> biplot(pcs, choices = c(1,2))
```



Slika 8: Biplot graf za prve dvije glavne komponente

Ispitujući utjecaje dobivene tablično i biplot grafom, zaključuje se da prva glavna komponenta mjeri ravnotežu između dvije količine: broj gledanja videa (*videos*) i broj bodova ostvaren na kvizovima (*quizzes*) (velike pozitivne težine) nasuprot prisustvovanju prezentacijama Red i Stog (velike negativne težine). Prva glavna komponenta opisuje studente koji su mnogo puta gledali snimke predavanja i ostvarili velik broj bodova na kvizovima te vjerojatno nisu pristupili prezentacijama Red i Stog, a prema postotku varijance takvih je 33,18% u skupu podataka. Na drugu glavnu komponentu najviše utječe broj bodova ostvaren na kvizovima (*quizzes*). Na sličan način moguće je tumačiti i ostale glavne komponente kako bi se naučilo o strukturi podataka.

8. Ostali poznatiji algoritmi za redukciju dimenzija

Kao što je u prethodnom poglavlju istaknuto, PCA algoritam koristan je za redukciju dimenzija ako prve tri glavne komponente mogu opisati barem 80% varijance. Kako to vrlo često nije slučaj, potrebno je upotrijebiti druge tehnike koje omogućavaju prikaz podataka u nižim dimenzijama od originalne. Iz tog razloga, u okviru ovog poglavlja teoretskom osnovom će biti predstavljeni neki istaknutiji algoritmi koji pokušavaju postići isti cilj.

8.1. Višedimenzionalno skaliranje (MDS)

Višedimenzionalno skaliranje (engl. *Multidimensional Scaling - MDS*) je algoritam koji daje matricu udaljenosti koja sadrži udaljenosti između svakog para objekata u skupu i postavlja svaki objekt u n-dimenzionalni prostor, gdje *n* predstavlja odabrani broj dimenzija, tako da su udaljenosti između objekata sačuvane što je moguće bolje. MDS se najčešće koristi kao alat za vizualizaciju. Najbolje odgovara problemu koji uključuje stvarne udaljenosti, na primjer udaljenost među gradovima. U grafu dobivenom na temelju ovog algoritma, objekti koji su sličniji međusobno su bliže smješteni od objekata s kojima su manje slični. Višedimenzionalni prostor označava činjenicu da ne postoji ograničenje na dvodimenzionalne grafikone ili podatke. Mogući su trodimenzionalni, četverodimenzionalni i viši grafikoni[5].

O višedimenzionalnom skaliranju se može razmišljati i u kontekstu redukcije dimenzija, tj. pojednostavljenju podataka smještanjem u niže dimenzije. Podaci koji se u nalaze zajedno u nižoj dimenziji zadržavaju slična svojstva. Na primjer, dvije podatkovne točke koje su međusobno bliske u višedimenzionalnom prostoru, također će biti bliske u prostoru nižih dimenzija. U primjeru skupa podataka „edukacija88“, to znači da ako se smanji broj varijabli (dimenzija), odnosi među studentima (objektima) ostat će isti i neće se izgubiti informacije koje originalni skup podataka nosi.

8.2. Analiza nezavisnih komponenti (ICA)

Analiza nezavisnih komponenti (engl. *Independent Component Analysis - ICA*) je još jedan linearni algoritam koji identificira novu osnovu na kojoj će predstavljati izvorne podatke, ali nastoji postići drugačiji cilj od PCA. ICA se pojavio u obradi signala, a problem koji želi riješiti naziva se razdvajanje slijepih izvora. Obično je predstavljen kao problem koktel zabave u kojem određen broj gostiju istovremeno govori u jedan mikrofon tako da on bilježi preklapajuće signale. ICA pokušava razgraditi multivarijantni signal na statistički neovisne signale pretpostavljajući da postoji onoliko različitih mikrofona koliko ima zvučnika[2].

U kontekstu redukcije dimenzija, ICA pretpostavlja da je svaki uzorak podataka mješavina neovisnih komponenti i njegov je cilj pronaći te neovisne komponente. Tako dobivene neovisne komponente mogu biti korištene u procesu dubinske analize podataka ako je njihov broj manji od broja varijabli u originalnom skupu podataka.

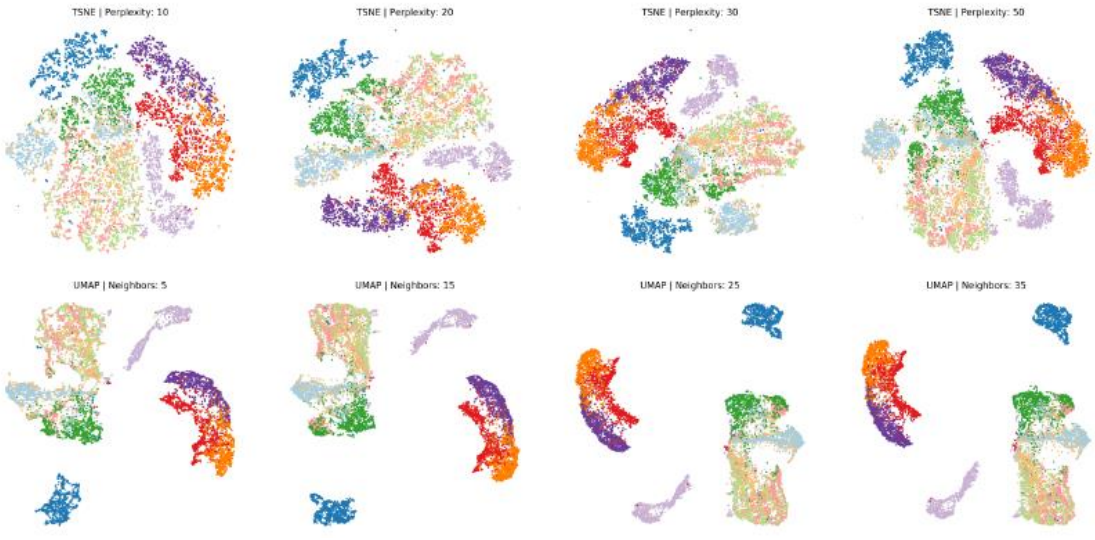
8.3. T-distribuirano stohastičko umetanje u susjedstvo (t-SNE)

T-distribuirano stohastičko umetanje u susjedstvo (engl. *T-distributed Stochastic Neighbor Embedding - t-SNE*) je algoritam za vizualizaciju kojeg su razvili Laurens van der Maaten i Geoffrey Hinton. To je tehnika nelinearne redukcije dimenzija, pogodna za umetanje podataka više dimenzije u niždimenzionalni prostor s dvije ili tri dimenzije. Konkretno, svaki višedimenzionalni objekt modelira se dvodimenzionalnom ili trodimenzionalnom točkom na takav način da slične objekte modeliraju točke u blizini, a različite objekte modeliraju udaljene točke s velikom vjerojatnošću.

T-SNE algoritam odvija se u dvije glavne faze. Prvo, t-SNE konstruira raspodjelu vjerojatnosti za parove viših dimenzija na takav način da slični objekti imaju veliku vjerojatnost odabira, dok različite točke imaju vrlo malu vjerojatnost odabira. Drugo, t-SNE definira sličnu raspodjelu vjerojatnosti nad točkama u niždimenzionalnoj mapi i minimizira divergenciju između dviju distribucija s obzirom na lokacije točaka na mapi. Iako se često čini da t-SNE grafovi često prikazuju klastere, na nakupine podataka u grafu mogu snažno utjecati odabrani parametri, pa je potrebno dobro razumijevanje parametara za t-SNE[4].

8.4. Jedinstvena aproksimacija i projekcija razdjelnika (UMAP)

Jedinstvena aproksimacija i projekcija razdjelnika (engl. *Uniform Manifold Approximation and Projection – UMAP*) noviji je algoritam za vizualizaciju i smanjenje dimenzije skupa podataka. UMAP stvara niždimenzionalni prikaz podataka viših dimenzija koji čuva relevantnu strukturu. Traži projekciju podataka u nižim dimenzijama koja ima najbližu moguću ekvivalentnu topološku strukturu izvornim podacima više dimenzije[2]. UMAP je sličan t-SNE algoritmu, ali čuva više globalne strukture u podacima. Slika 9 u donjem redu prikazuje kako UMAP doista razdvaja različite klastere, dok t-SNE u gornjem redu pruža detaljniji uvid u lokalnu strukturu.



Slika 9: Djelovanje t-SNE i UMAP algoritma[2]

9. Redukcija dimenzija korištenjem regresijskih modela

U prethodnim poglavljima predstavljene su metode za smanjenje broja varijabli pomoću statistike sažetaka, analize korelacije i analize glavnih komponenti. Sve se te metode smatraju istraživačkim. Neke od njih u potpunosti zanemaruju varijablu ishoda (PCA), dok druge metode pokušavaju pronaći odnos prediktora i varijable ishoda. Drugi pristup smanjenju dimenzije skupa podataka, koji izravno uzima u obzir prediktivni ili klasifikacijski zadatak, jest uklapanje regresijskog modela. Za predviđanje koristi se model linearne regresije, a za klasifikaciju model logističke regresije. U oba slučaja moguće je upotrijebiti postupke odabira podskupina koji algoritamski biraju podskup varijabli prediktora među većim skupom[4].

Prilagođeni regresijski modeli također se mogu koristiti za daljnje kombiniranje sličnih kategorija: kategorije koje imaju koeficijente koji nisu statistički značajni, tj. imaju visoku p-vrijednost, mogu se kombinirati s referentnom kategorijom, jer se čini da njihova razdvojenost od referentne kategorije nema značajnog učinka na varijablu ishoda. Štoviše, kategorije koje imaju slične vrijednosti koeficijenta često se mogu kombinirati, jer je njihov utjecaj na varijablu ishoda sličan.

Još jedna metoda za smanjenje broja varijabli i za kombiniranje kategorija kategoričke varijable je primjena klasifikacijskih i regresijskih stabala. Klasifikacijska stabla koriste se za klasifikacijske zadatke, a regresijska stabla za zadatke predviđanja. U oba slučaja algoritam stvara binarne prijelome na prediktorima koji najbolje klasificiraju/predviđaju varijablu ishoda (na primjer, iznad/ispod 30 godina). Rezultirajuće stablo može se koristiti za određivanje važnih prediktora. Prediktori (numerički ili kategorički) koji se ne pojavljuju u stablu mogu se ukloniti. Slično tome, kategorije koje se ne pojavljuju na stablu mogu se kombinirati[3].

10. Zaključak

Skupovi podataka koji se uobičajeno koriste u dubinskoj analizi podataka mogu imati milijune zapisa i tisuće varijabli. Malo je vjerojatno da su sve varijable neovisne i među njima nema korelacije. Znanstvenici koji se bave dubinskom analizom podataka moraju se zaštititi od multikolinearnosti, stanja u kojem su neke od varijabli predviđanja snažno povezane jedna s drugom. Multikolinearnost dovodi do nestabilnosti u prostoru rješenja, što posljedično dovodi do mogućih nekoherentnih rezultata. Čak i ako se izbjegne takva nestabilnost, uključivanje varijabli koje su visoko povezane ima tendenciju preuveličavanja određene komponente modela, jer se ona u biti dvostruko broji.

Upotreba previše varijabli prediktora za modeliranje odnosa s izlaznom varijablom može nepotrebno zakomplicirati interpretaciju analize pa bi trebalo razmotriti zadržavanje broja prediktora na veličini koja bi se lako protumačila. Također, zadržavanje previše varijabli može dovesti do prekomjernog preklapanja podataka (engl. *overfitting*), u kojem se smanjuje općenitost rezultata, jer se novi podaci ne ponašaju isto kao i podaci za treniranje za sve varijable.

Nadalje, analiza samo na razini varijable mogla bi propustiti temeljne odnose prediktora. Nekoliko prediktora može prirodno „pasti“ u jednu grupu koja se bavi jednim aspektom podataka. Na primjer, varijable kao što su: stanje na štednom računu, stanje na tekućem računu, vrijednost kapitala i vrijednost dionica mogu sve zajedno pasti pod jednu komponentu, imovinu. U nekim aplikacijama, poput onih za analizu slike, zadržavanje početne dimenzije skupa podataka učinilo bi većinu problema neizrecivim.

Ljudi prirodno imaju sposobnosti prepoznavanja vizualnih uzoraka koje omogućuju na intuitivan način da na prvi pogled razaznaju uzorke u grafičkim prikazima jer bi im inače obrasci mogli izmaći ako su predstavljeni numerički ili tekstualno. Međutim, čak i najnaprednije tehnike vizualizacije podataka ne prelaze mnogo više od pet dimenzija. Upravo iz tog razloga, bitno je koristiti metode redukcije dimenzija, kako bi se mogli prikazati odnosi među stotinama, ili čak tisućama dimenzija (varijabli).

Znanstvenicima na raspolaganju stoji mnogo poznatih tehnika za redukciju dimenzija skupa podataka, međutim poteškoća u njihovoj primjeni je u tome što je svaka tehnika redukcije dimenzija dizajnirana tako da održava samo određene aspekte izvornih podataka i stoga može biti prikladna za jedan zadatak, a neprikladna za neki drugi. Za mnoge zadatke dubinske analize podataka može biti teško objektivno pronaći pravu metodu ili kombinaciju parametara za redukciju dimenzija. Da bi se pronašla prava tehnika za redukciju dimenzija, u obzir treba uzeti prirodu i razlučivost podataka. Treba shvatiti traže li se samo mali odnosi ili je fokus na dugoročnim odnosima. U prvom slučaju treba razmotriti samo tehnike redukcije dimenzija koje čuvaju lokalnu strukturu, a u drugom se treba se usredotočiti na tehnike redukcije dimenzija koje čuvaju globalnu strukturu.

Prije korištenja bilo kakve automatizirane metode za redukciju dimenzija, potrebno je skup podataka praktički razmotriti jer je vrlo lako moguće da će se uočiti određene varijable

koje su kandidati za eliminaciju ili ekstrakciju. Nakon toga potrebno je upoznati podatke i njihove karakteristike kroz razne sažetke i grafikone. U ovom radu detaljnije su opisane metode prikaza sažetka cijelog skupa podataka te agregacije i stvaranja zaokretnih tablica. Nakon toga objašnjena je analiza korelacije, koja je vrlo koristan pristup za pronalaženje odnosa među varijablama. Tako su na korištenom skupu podataka analizom matrice i grafa korelacije uočene varijable koje se potencijalno mogu ukloniti iz skupa podataka bez gubitka mnogo informacija. Opisan je i postupak smanjenja broja kategorija u kategoričkim varijablama jer mnoge metode dubinske analize podataka zahtijevaju njihovo razdvajanje na više izvedenih varijabli. Kako u skupu podataka „edukacija88“ nije bilo kategoričkih varijabli, jedna numerička varijabla je pretvorena u numeričku te je pomoću pivot tablica prikazano koje bi se kategorije (varijable) mogle kombinirati u jednu.

Kao metoda koja, za razliku od prethodno predstavljenih, zanemaruje odnos s izlaznom varijablom i stvara nove izvedene varijable, predstavljena je analiza glavnih komponenti (PCA). Koraci algoritma su detaljno opisani, nakon čega je napravljen praktični primjer i zaključeno je da ovaj algoritam nije odgovarajući za redukciju dimenzija korištenog skupa podataka jer bi trebalo previše glavnih komponenti kako bi se opisala većina varijance koju skup posjeduje. Dodatno su, samo teoretskom osnovom, predstavljeni i ostali poznatiji algoritmi redukcije dimenzija koje nastoje na različite načine projicirati podatke u dimenzijama nižim od početne. U posljednjem poglavlju ukratko je spomenut drugačiji pristup redukciji dimenzija, koji pretpostavlja korištenje regresijskog modela.

Na posljetku, potrebno je spomenuti da je područje dubinske analize podataka ušlo u svakodnevnu primjenu brzinom kojom se nitko nije nadao i definitivno je potrebno konstantno produbljivati znanja u ovom području kako bi se iskoristile sve njegove prednosti.

11. Popis literature

[1] Čanić, J. „Primjena logističke regresije u znanosti o podacima.“ Diplomski rad, Odjel za informatiku Sveučilišta u Rijeci, 2017.

[2] Jansen, S. *Hands-On Machine Learning for Algorithmic Trading*. Birmingham: Packt, 2018.

[3] Larose, D.T., i Larose, C.D. *Data Mining and Predictive Analytics*. New York: Wiley, 2015.

[4] Shmueli, G., Bruce, P.C., Yahav, I., Patel, N.R., i Lichtendahl, K.C. *Data Mining for Business Analytics*. New York: Wiley, 2018.

[5] Andale, S. „Multidimensional Scaling: Definition, Overview, Examples“. *Statistics How To*. Pristupano 07. travnja 2020. https://www.statisticshowto.com/multidimensional-scaling/

[6] „Data Mining – What it is and why it matters“. *SAS*. Pristupano 25. ožujka 2020. https://www.sas.com/en_us/insights/analytics/data-mining.html

[7] Freytag, S. „Workshop: Dimension reduction with R“. *RPubs*. Pristupano 02. travnja 2020. https://rpubs.com/Saskia/520216

[8] Jaadi, Z. „A step by step explanation of Principal Component Analysis“. *BuiltIn*. Pristupano 04. travnja 2020. https://builtin.com/data-science/step-step-explanation-principal-component-analysis

[9] Ngo, L. „How to read PCA biplots and scree plots“. *BioTuring*. Pristupano 07. travnja 2020. https://blog.bioturing.com/2018/06/18/how-to-read-pca-biplots-and-scree-plots/

12. Popis slika

13. Popis priloga

1. *R* skripta „Gasparovic_diplomski.R“ koja sadrži kod korišten za izradu praktičnih primjera

2. Skup podataka „edukacija88.csv“ korišten za izradu praktičnih primjera