3. laboratorijska vježba

Multivarijatna analiza podataka

ak. god. 2021/2022

1. Uvod i upute za predaju

Cilj ove laboratorijske vježbe je primijeniti osnovne koncepte multivarijatne analize podataka, istražiti podatke te ispitati hipoteze. Preduvjet za rješavanje vježbe je osnovno znanje programskog jezika *R* i rad s *R Markdown* dokumentima. Sama vježba je koncipirana kao projekt u kojem istražujete i eksperimentirate koristeći dane podatke - ne postoji nužno samo jedan točan način rješavanja svakog podzadatka.

Rješavanje vježbe svodi se na čitanje uputa u tekstu ovog dokumenta, nadopunjavanje blokova kôda (možete dodavati i dodatne blokove kôda ukoliko je potrebno) i ispisivanje rezultata (u vidu ispisa iz funkcija, tablica i grafova). Vježbu radite samostalno, a svoje rješenje branite na terminima koji su vam dodijeljeni u kalendaru. Pritom morate razumjeti teorijske osnove u okviru onoga što je obrađeno na predavanjima i morate pokazati da razumijete sav kôd koji ste napisali.

Vaše rješenje potrebno je predati u sustav *Moodle* u obliku dvije datoteke:

1. Ovaj .Rmd dokument s Vašim rješenjem (naziva IME_PREZIME_JMBAG.rmd),

2. PDF ili HTML dokument kao izvještaj generiran iz vašeg .Rmd rješenja (također naziva IME_PREZIME_JMBAG).

Rok za predaju je **7. lipnja 2022. u 23:59h**. **Jedan od uvjeta za prolaz predmeta je minimalno ostvarenih 50% bodova na svim laboratorijskim vježbama. Nadoknade laboratorijskih vježbi neće biti organizirane.** Za sva dodatna pitanja svakako se javite na email adresu predmeta: *map@fer.hr*.

2. Podatkovni skup

U ovoj laboratorijskoj vježbi analizirat ćemo skup podataka s najpoznatijim klasičnim skladbama prema glasovima korisnika stranice classicalmusiconly.com. Varijable su redom:

- `title` — naziv skladbe
- `composer` — ime skladatelja
- `category` — kategorija skladbe
- `year` — godina kada je skladba napisana
- `stars` — broj glasova (koliko korisnika je označilo skladbu kao favorit)

Varijable `composer` i `category` treba tretirati kao da se radi o kategorijskim varijablama.

Napomena: ako koristite funkciju `factor` za modeliranje kategorijskih varijabli, tada je potrebno nakon filtriranja skupa podataka ponovno pozvati `factor` nad preostalim vrijednostima kategorijske varijabli — u suprotnom će izbačene vrijednosti ostati zapamćene, što će davati nepregledne rezultate.

2.1. Predobrada i analiza podataka

Učitajte datoteku `classical.tsv` i proučite podatke.

```
# Vaš kod ovdje
data <- read.csv("classical.tsv", sep = "\t")
```

```
summary(data)
```

```
##       rank           title              composer           category
##  Min.   :   1.0   Length:2415        Length:2415        Length:2415
##  1st Qu.: 604.5   Class :character   Class :character   Class :character
##  Median :1208.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   :1208.0
##  3rd Qu.:1811.5
##  Max.   :2415.0
##
##       year          stars
##  Min.   :1487   Min.   :   0.00
##  1st Qu.:1835   1st Qu.:   1.00
##  Median :1895   Median :   2.00
##  Mean   :1880   Mean   :  38.17
##  3rd Qu.:1933   3rd Qu.:   5.00
##  Max.   :2015   Max.   :2515.00
##  NA's   :242
```

```
# Vaš kod ovdje

data$composer <- factor(data$composer)
data$category <- factor(data$category)
```

```
# Vaš kod ovdje

summary(data)
```

```
##       rank           title                              composer
##  Min.   :   1.0   Length:2415        Wolfgang Amadeus Mozart:  85
##  1st Qu.: 604.5   Class :character   Ludwig van Beethoven   :  81
##  Median :1208.0   Mode  :character   Johann Sebastian Bach  :  73
##  Mean   :1208.0                      Johannes Brahms        :  64
##  3rd Qu.:1811.5                      Franz Schubert         :  62
##  Max.   :2415.0                      Joseph Haydn           :  57
##                                      (Other)                :1993
##                 category         year          stars
##  Unsorted Orchestral: 321   Min.   :1487   Min.   :   0.00
##  Piano Sonata       : 290   1st Qu.:1835   1st Qu.:   1.00
##  Symphony           : 266   Median :1895   Median :   2.00
##  Opera              : 201   Mean   :1880   Mean   :  38.17
##  Choral orchestral  : 129   3rd Qu.:1933   3rd Qu.:   5.00
##  String Quartet     : 129   Max.   :2015   Max.   :2515.00
##  (Other)            :1079   NA's   :242
```

Proučite koliki je **ukupan broj skladbi po pojedinom skladatelju**, te koliki je **ukupan broj skladbi po pojedinoj kategoriji**. **Ispišite** ih poredano silazno po broju skladbi.

```
# Vaš kod ovdje


# ukupan broj skladbi po pojedinom skladatelju

data %>%
  group_by(composer) %>%
```

```
  summarize(ncompositions_composer = n())%>%
  arrange(desc(ncompositions_composer))
```

```
## # A tibble: 292 x 2
##    composer                ncompositions_composer
##    <fct>                                    <int>
##  1 Wolfgang Amadeus Mozart                     85
##  2 Ludwig van Beethoven                        81
##  3 Johann Sebastian Bach                       73
##  4 Johannes Brahms                             64
##  5 Franz Schubert                              62
##  6 Joseph Haydn                                57
##  7 Robert Schumann                             48
##  8 Claude Debussy                              38
##  9 Dmitri Shostakovich                         38
## 10 Béla Bartók                                 35
## # ... with 282 more rows
```

```
# ukupan broj skladbi po pojedinoj kategoriji


data %>%
  group_by(category) %>%
  summarize(ncompositions_category = n())%>%
  arrange(desc(ncompositions_category))
```

```
## # A tibble: 39 x 2
##    category            ncompositions_category
##    <fct>                                <int>
##  1 Unsorted Orchestral                    321
##  2 Piano Sonata                           290
##  3 Symphony                               266
##  4 Opera                                  201
##  5 Choral orchestral                      129
##  6 String Quartet                         129
##  7 Lieder / Song                          123
##  8 Piano Concerto                         106
##  9 Chant                                   82
## 10 Violin Concerto                         82
## # ... with 29 more rows
```

Ispišite imena prvih deset skladatelja čije skladbe imaju **najveći ukupan broj glasova**, te prvih deset skladatelja čije skladbe imaju **najveći prosječan broj glasova**.

```
# Vaš kod ovdje

limit <- 10


# najveći ukupan broj glasova

data %>%
  group_by(composer) %>%
  summarise(nvotes = sum(stars)) %>%
  arrange(desc(nvotes)) %>%
```

```
  top_n(limit)
```

```
## Selecting by nvotes
```

```
## # A tibble: 10 x 2
##    composer                nvotes
##    <fct>                    <int>
##  1 Ludwig van Beethoven     11797
##  2 Wolfgang Amadeus Mozart   9879
##  3 Johann Sebastian Bach     9264
##  4 Pyotr Ilyich Tchaikovsky  5790
##  5 Johannes Brahms           4274
##  6 Frédéric Chopin           4176
##  7 Antonio Vivaldi           2677
##  8 Antonín Dvořák            2617
##  9 Maurice Ravel             2122
## 10 Claude Debussy            2086
```

```
# najveći prosječan broj glasova

data %>%
  group_by(composer) %>%
  summarise(nvotes = sum(stars)) %>%
  mutate(avgvotes = round((nvotes/sum(nvotes)*100))) %>%
  arrange(desc(avgvotes)) %>%
  top_n(limit)
```

```
## Selecting by avgvotes
```

```
## # A tibble: 16 x 3
##    composer                nvotes avgvotes
##    <fct>                    <int>    <dbl>
##  1 Ludwig van Beethoven     11797       13
##  2 Wolfgang Amadeus Mozart   9879       11
##  3 Johann Sebastian Bach     9264       10
##  4 Pyotr Ilyich Tchaikovsky  5790        6
##  5 Frédéric Chopin           4176        5
##  6 Johannes Brahms           4274        5
##  7 Antonín Dvořák            2617        3
##  8 Antonio Vivaldi           2677        3
##  9 Claude Debussy            2086        2
## 10 Dmitri Shostakovich       1616        2
## 11 Franz Schubert            1998        2
## 12 Gustav Mahler             2058        2
## 13 Igor Stravinsky           1524        2
## 14 Maurice Ravel             2122        2
## 15 Richard Wagner            1816        2
## 16 Sergei Rachmaninoff       1802        2
```

Iz podataka **uklonite** sve skladbe čiji skladatelji se pojavljuju vrlo rijetko (npr. manje od 5–10 puta). Zatim, uklonite sve skladbe čija kategorija se pojavljuje vrlo rijetko (npr. manje od 5–10 puta).

```
# Vaš kod ovdje

limit = 7
```

```r
# uklonite sve skladbe čiji skladatelji se pojavljuju vrlo rijetko

# extract
composers <-

data %>%
  group_by(composer) %>%
  summarise(ntotal = n()) %>%
  filter(ntotal <= limit)



# uklonite sve skladbe čija kategorija se pojavljuje vrlo rijetko

# extract
categories <-

data %>%
  group_by(category) %>%
  summarise(ntotal = n()) %>%
  filter(ntotal <= limit)




# ukloni sve sto nije u extract
clean_data <-

data %>%
  filter(!(composer %in% composers$composer)) %>%
  filter(!(category %in% categories$category))


# ukloni prazne kategorije
clean_data$composer <- droplevels(clean_data$composer)
clean_data$category <- droplevels(clean_data$category)


# rezultati

View(clean_data)
```

U ostatku vježbe koristite ovaj filtrirani podskup podataka. Obratite pozornost i na nedostajuće vrijednosti (ako ih ima). Razmislite na koji način ćete ih tretirati u ostatku vježbe.

3. Višedimenzionalno skaliranje

3.1. Metričko skaliranje

Izračunajte i prikažite kontingencijsku tablicu za varijable `composer` i `category`. Budući da će kontingencijska tablica biti jako velika, prikažite samo njezin dio koji sadrži nekolicinu najčešćih skladatelja i kategorija.

```
# Vaš kod ovdje


kont_tablica = table(clean_data$composer, clean_data$category)

kont_tablica[0:10, 0:10]
```

```
##
##                      Ballet Cello Concerto Cello Sonata Chant Choral orchestral
##   Aaron Copland           2              0            0     0                  0
##   Alban Berg              0              0            0     0                  0
##   Alexander Glazunov      2              0            0     0                  0
##   Alexander Scriabin      0              0            0     0                  0
##   Alfred Schnittke        1              0            1     1                  0
##   Anton Bruckner          0              0            0     1                  1
##   Anton Webern            0              0            0     0                  0
##   Antonín Dvořák          0              1            0     0                  1
##   Antonio Vivaldi         0              0            0     0                  5
##   Arnold Schoenberg       0              0            0     0                  0
##
##                      Clarinet Sonata Double Concerto Harpsichord conceto
##   Aaron Copland                    0               0                   0
##   Alban Berg                       1               0                   0
##   Alexander Glazunov               0               0                   0
##   Alexander Scriabin               0               0                   0
##   Alfred Schnittke                 0               0                   0
##   Anton Bruckner                   0               0                   0
##   Anton Webern                     0               0                   0
##   Antonín Dvořák                   0               0                   0
##   Antonio Vivaldi                  0               0                   0
##   Arnold Schoenberg                0               0                   0
##
##                      Lieder / Song Mass / Requiem
##   Aaron Copland                  0              0
##   Alban Berg                     2              0
##   Alexander Glazunov             0              0
##   Alexander Scriabin             0              0
##   Alfred Schnittke               0              1
##   Anton Bruckner                 0              3
##   Anton Webern                   0              0
##   Antonín Dvořák                 0              1
##   Antonio Vivaldi                0              0
##   Arnold Schoenberg              3              0
```

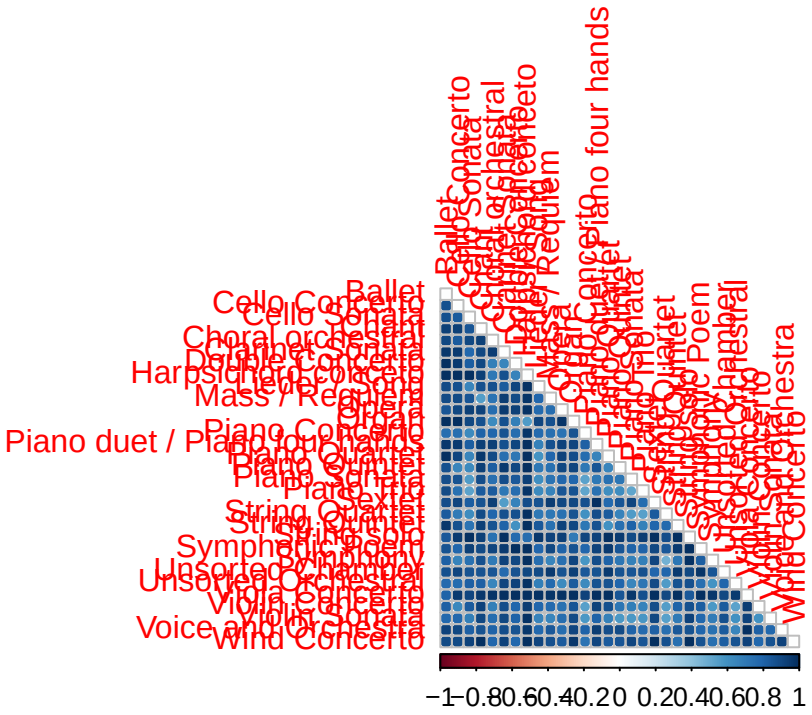
Izračunajte udaljenosti među kategorijama koristeći *totalnu varijacijsku udaljenost*. Neka je C kontingencijska matrica dimenzija $N \times M$, te neka su skladatelji raspoređeni po retcima, a kategorije po stupcima.

Udaljenost između dvije kategorije i i j računa se kao

$$\delta_{i,j} = \frac{1}{2} \sum_{k=1}^{N} \left| \frac{C_{k,i}}{S_i} - \frac{C_{k,j}}{S_j} \right|,$$

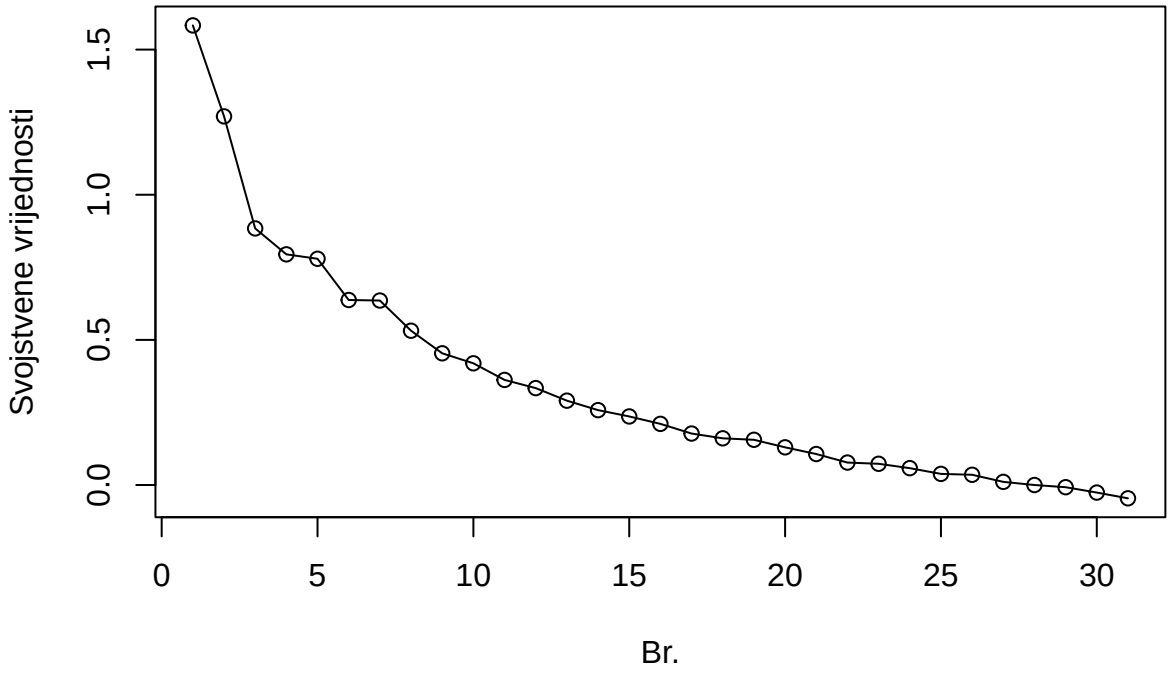
gdje je $S_i = \sum_{k=1}^{N} C_{k,i}$. **Izračunajte i vizualizirajte matricu udaljenosti kategorija.** Koji parovi kategorija su najsličniji, a koji najrazličitiji?

```
# Vaš kod ovdje


mat <- as.matrix(kont_tablica)

n <- ncol (mat)
dist_mat <- matrix(NA, n, n)
diag(dist_mat) <- 0


for (i in 1:(n-1))
{
  for (j in (i+1):n)
  {
    d2 <- (1/2)*sum(abs((mat[, i] / sum(mat[, i])) - (mat[,j] / sum(mat[,j]))))
    dist_mat[i, j] <- dist_mat[j, i] <- d2
  }
}


#
colnames(dist_mat) <- colnames(kont_tablica)
rownames(dist_mat) <- colnames(kont_tablica)


# vizualizirajte matricu udaljenosti kategorija
corrplot(dist_mat, type="lower")
```



Provedite postupak klasičnog višedimenzionalnog skaliranja. **Skicirajte** *scree plot* svojstvenih vrijednosti. Koliki broj dimenzija bi objasnio većinu varijance?

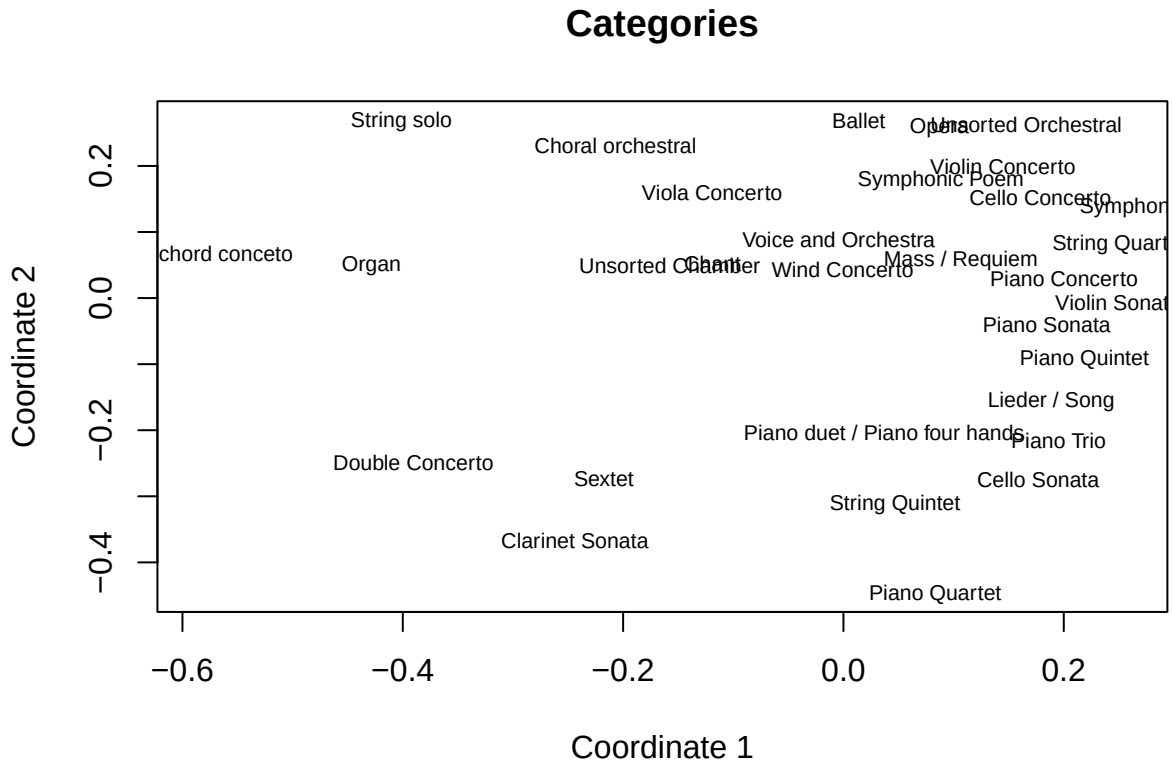
```
# Vaš kod ovdje

scaled = cmdscale(dist_mat, k=2, eig=T)


plot(x = seq(1:length(scaled$eig)), y = scaled$eig, type = "o", xlab = "Br.", ylab = "Svojstvene vrijedn
```



Prikažite kategorije na grafu raspršenja s **dvije dimenzije** i **označite** koju kategoriju pojedina točka

predstavlja. Uočavate li kakvo prirodno grupiranje kategorija?

```
# Vaš kod ovdje

mds <- cmdscale(dist_mat, k=2, eig=TRUE)
x <- mds$points[, 1]
y <- mds$points[, 2]

y <- -y

plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2", main="Categories", type="n")
text(x, y, labels = row.names(dist_mat), cex=0.7)
```



Prikažite graf raspršenja za skaliranje dobiveno *Sammon* metodom i **označite** koju kategoriju pojedina točka predstavlja. Usporedite ga s gore dobivenim grafom. Kakve razlike uočavate?

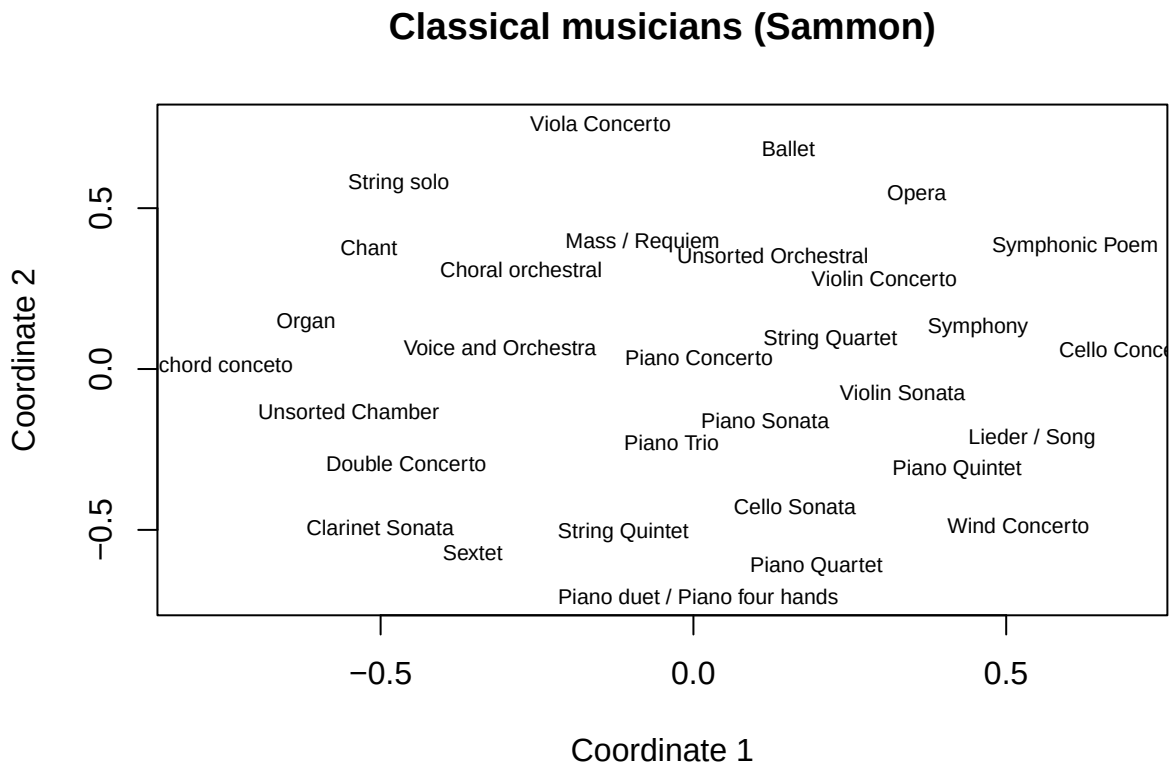
```
# Vaš kod ovdje


mds_sammon=sammon(dist_mat, y = cmdscale(dist_mat, 2), k = 2, niter = 100, trace = TRUE, magic = 0.2, t
```

```
## Initial stress        : 0.32628
## stress after  10 iters: 0.12318, magic = 0.461
## stress after  20 iters: 0.11354, magic = 0.228
## stress after  30 iters: 0.11076, magic = 0.500
## stress after  40 iters: 0.10992, magic = 0.500
## stress after  50 iters: 0.10935, magic = 0.500
## stress after  60 iters: 0.10924, magic = 0.500
## stress after  70 iters: 0.10921, magic = 0.500
```

```
## 'magic' je parametar koji kontrolira korak Newtonove metode
names(mds_sammon)
```

```
## [1] "points" "stress" "call"
```

```
x1 <- mds_sammon$points[, 1]
y1 <- mds_sammon$points[, 2]

y1 <- -y1

plot(x1, y1, xlab="Coordinate 1", ylab="Coordinate 2", main="Classical musicians (Sammon)", type="n")

text(x1, y1, labels = row.names(dist_mat), cex=0.7)
```



3.2. Nemetričko skaliranje

Odaberite proizvoljan broj skladatelja (npr. 10–30) s **najvećim prosječnim brojem** glasova po skladbama, te **kreirajte** novi podskup podataka tako da sadrži samo skladbe tih autora.

```
# Vaš kod ovdje

limit <- 10



# najvećim prosječnim brojem glasova - 10

data_filtered <-

data %>%
```

```
  group_by(composer) %>%
  summarise(nvotes = sum(stars)) %>%
  mutate(avg = round((nvotes/sum(nvotes)*100))) %>%
  arrange(desc(avg)) %>%
  top_n(limit)
```

```
## Selecting by avg
```

```
# novi podskup podataka tako da sadrži samo skladbe tih autora
data_clean3 <-

  data %>%
    filter(composer %in% data_filtered$composer) %>%
    select(composer, category)


data_clean3$composer <- droplevels(data_clean3$composer)
data_clean3$category <- droplevels(data_clean3$category)
```

U nastavku vježbe koristite ovako generirani podskup podataka.

Definirajte jednu proizvoljnu mjeru različitosti između dva skladatelja. Vaša mjera različitosti može uključivati npr.

- euklidsku udaljenost između broja skladbi po kategorijama,
- korelacijsku udaljenost između broja skladbi po kategorijama,
- totalnu varijacijsku udaljenost između broja skladbi po kategorijama,
- ukupan broj skladbi po pojedinoj kategoriji,
- prosječnu godinu izdanja svih skladbi,
- ukupan broj glasova po svim skladbama,
- prosječan broj glasova po svim skladbama,
- ...

Pokušajte konstruirati mjeru različitosti koju ćete moći intuitivno interpretirati.

Izračunajte matricu različitosti za skladatelje koristeći Vašu mjeru različitosti. **Izračunajte** izometrično skaliranje i **prikažite** rezultat grafom raspršenja s **dvije dimenzije**, te **označite** koju kategoriju pojedina točka predstavlja. Možete li interpretirati dobiveni graf u skladu s korištenom mjerom različitosti?

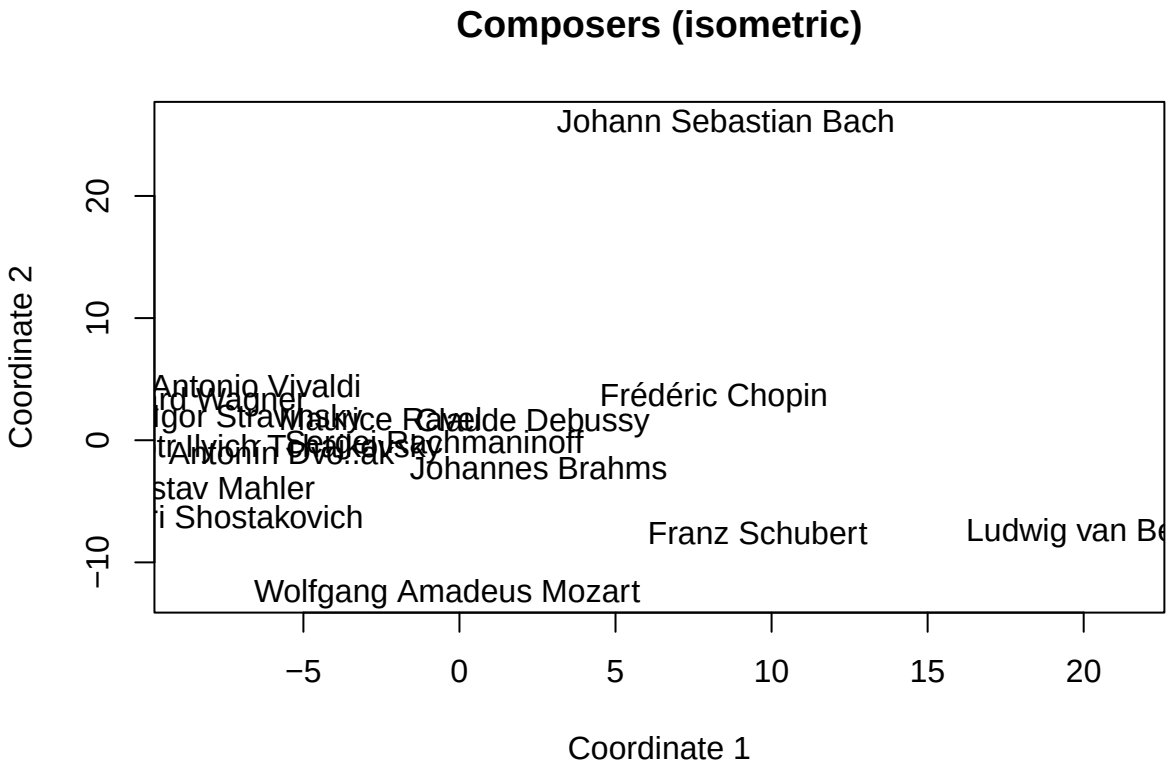
```
# Vaš kod ovdje

kont_tablica_3 = table(data_clean3$composer,data_clean3$category)
distance_difference = dist(kont_tablica_3, method = "euclidean")
```

```
# Vaš kod ovdje

mds_iso=isoMDS(distance_difference, y = cmdscale(distance_difference, 2), k=2)
```

```
## initial  value 16.719962
## iter   5 value 11.292060
## iter  10 value 10.035570
## final  value 9.947763
## converged
```

```
plot(mds_iso$points, xlab="Coordinate 1", ylab="Coordinate 2",
     main="Composers (isometric)",type = "n")
text(mds_iso$points, labels = row.names(kont_tablica_3))
```

```
## Warning in text.default(mds_iso$points, labels = row.names(kont_tablica_3)):
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in text.default(mds_iso$points, labels = row.names(kont_tablica_3)):
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in text.default(mds_iso$points, labels = row.names(kont_tablica_3)):
## font metrics unknown for Unicode character U+0159
```

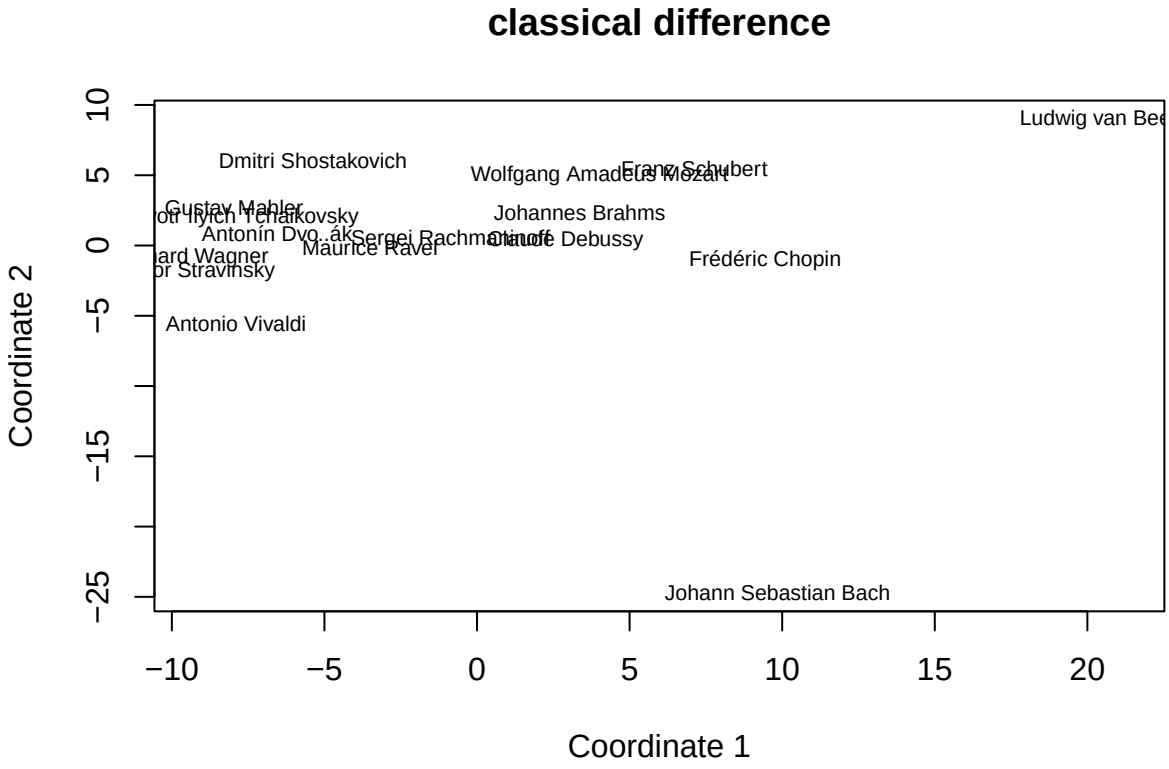


Ponovite jednu metodu **metričkog skaliranja** po izboru (klasično ili *Sammon*), ovaj puta korištenjem proizvoljne **mjere udaljenosti** nad skladateljima (npr. euklidska/korelacijska/*totalna varijacijska* udaljenost između broja skladbi po pojedinoj kategoriji). Pokušajte odabrati mjeru udaljenosti koju ćete lakše moći interpretirati. Možete li objasniti razlike u odnosu na prethodno dobiveni graf?

```
# Vaš kod ovdje

mds <- cmdscale(distance_difference, k=2, eig=TRUE)
x <- mds$points[, 1]
y <- mds$points[, 2]

y <- -y

plot(x, y, xlab="Coordinate 1", ylab="Coordinate 2",
     main="classical difference", type="n")
text(x, y, labels = row.names(kont_tablica_3), cex=0.7)
```

```
## Warning in text.default(x, y, labels = row.names(kont_tablica_3), cex = 0.7):
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in text.default(x, y, labels = row.names(kont_tablica_3), cex = 0.7):
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in text.default(x, y, labels = row.names(kont_tablica_3), cex = 0.7):
```

```
## font metrics unknown for Unicode character U+0159
```



4. Analiza korespondencije

Kreirajte novi podskup podataka tako da sadrži samo skladbe **prvih 5–10 skladatelja** po nekom kriteriju — proizvoljno odaberite kriterij po kojemu ćete ih poredati (npr. ukupan broj skladbi, ukupan broj glasova po svim skladbama, prosječan broj glasova po svim skladbama...). Zatim **odredite** koje kategorije skladbi se najčešće pojavljuju u dobivenom podskupu, te unutar njega zadržite samo skladbe koje pripadaju **5–10 najčešćih kategorija**.

```
# Vaš kod ovdje

limit <- 10


# Kreirajte novi podskup podataka - broj skladbi


# extract
data_kriterij <-

  data %>%
  group_by(composer) %>%
  summarize(ntitles = n()) %>%
  arrange(desc(ntitles)) %>%
  top_n(limit)
```

```
## Selecting by ntitles
```

```
data_clean4 <-
  clean_data %>%
    filter(composer %in% data_kriterij$composer) %>%
    select(composer, category)




# odredite koje kategorije skladbi se najčešće pojavljuju u dobivenom podskupu


# extract
data_category <-

  data_clean4 %>%
  group_by(category) %>%
  summarize(ntitles = n()) %>%
  arrange(desc(ntitles)) %>%
  top_n(limit)
```

```
## Selecting by ntitles
```

```
data_clean4 <-
  data_clean4 %>%
    filter(category %in% data_category$category)



# clean empty category variables

data_clean4$composer <- droplevels(data_clean4$composer)
data_clean4$category <- droplevels(data_clean4$category)
```

U nastavku vježbe koristite ovako generirani podskup podataka.

Izračunajte i prikažite kontingencijsku tablicu između skladatelja i kategorije skladbi koje su napisali.

```
# Vaš kod ovdje

kont_tablica = table(data_clean4$composer, data_clean4$category)
kont_tablica
```

```
##
##                          Choral orchestral Lieder / Song Opera Piano Concerto
##   Béla Bartók                             0             1     1              3
##   Claude Debussy                          0             5     1              1
##   Dmitri Shostakovich                     0             1     2              2
##   Franz Schubert                          1            19     1              0
##   Johann Sebastian Bach                  23             0     0              0
##   Johannes Brahms                         1             7     0              2
##   Joseph Haydn                            2             0     0              1
##   Ludwig van Beethoven                    0             1     1              5
##   Robert Schumann                         1             6     0              3
##   Sergei Prokofiev                        0             0     3              5
##   Wolfgang Amadeus Mozart                 3             0     7             13
##
```

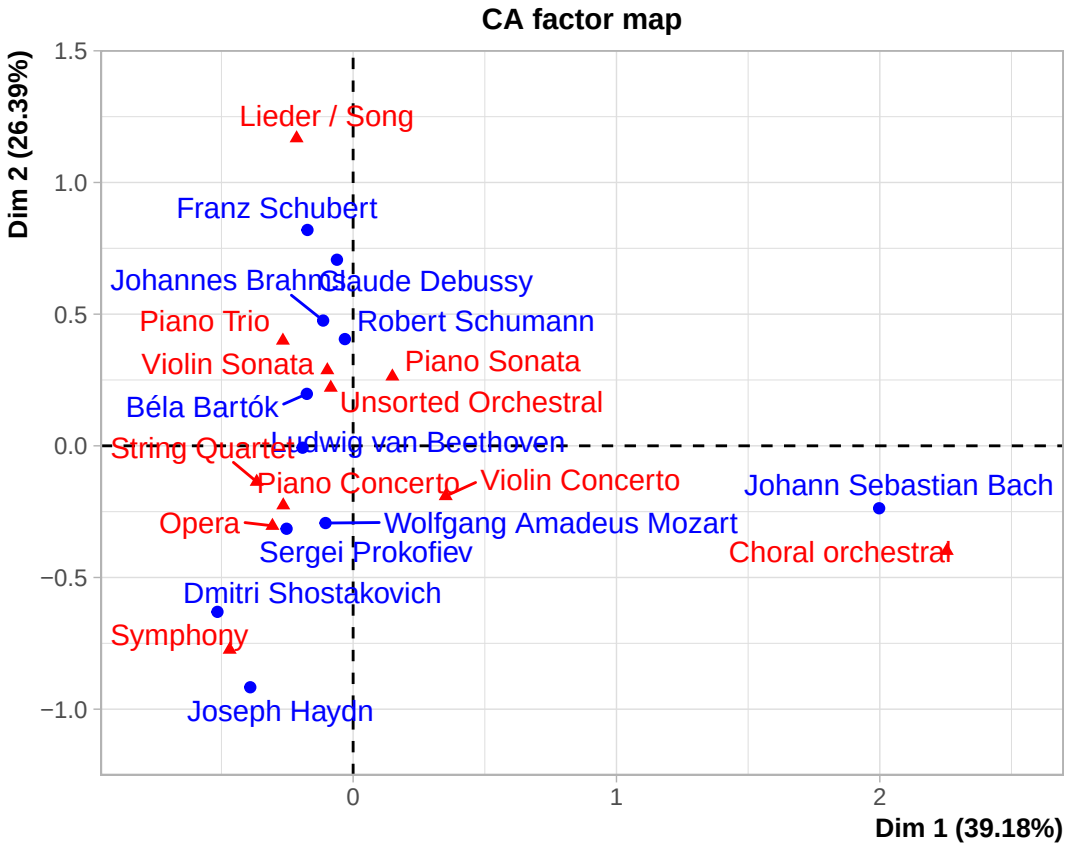
```
##                          Piano Sonata Piano Trio String Quartet Symphony
##   Béla Bartók                       5          1              6        0
##   Claude Debussy                   12          1              1        0
##   Dmitri Shostakovich               1          1              7       14
##   Franz Schubert                   13          3              4        6
##   Johann Sebastian Bach            14          0              0        0
##   Johannes Brahms                  10          5              3        4
##   Joseph Haydn                      7          2              6       32
##   Ludwig van Beethoven             26          3             17        9
##   Robert Schumann                  18          3              3        4
##   Sergei Prokofiev                  6          0              1        7
##   Wolfgang Amadeus Mozart          10          1              4       11
##
##                          Unsorted Orchestral Violin Concerto Violin Sonata
##   Béla Bartók                               6               2             3
##   Claude Debussy                            6               0             1
##   Dmitri Shostakovich                       3               1             0
##   Franz Schubert                            1               0             4
##   Johann Sebastian Bach                     2               3             1
##   Johannes Brahms                           6               1             3
##   Joseph Haydn                              0               0             0
##   Ludwig van Beethoven                      3               3             4
##   Robert Schumann                           2               1             1
##   Sergei Prokofiev                          4               2             2
##   Wolfgang Amadeus Mozart                   6               3             6
```

```
tablica <- as.data.frame.matrix(table(as.factor(data_clean4$composer),as.factor(data_clean4$category)))
res.ca <- CA(tablica, graph = FALSE)
```

Prikažite graf analize korespondencije između varijabli `composer` i `category`. Obratite pozornost na ukupnu objašnjenu varijancu. Na temelju dobivenog grafa i kontingencijske tablice iz prethodnog zadatka pokušajte odgovoriti na sljedeća pitanja:

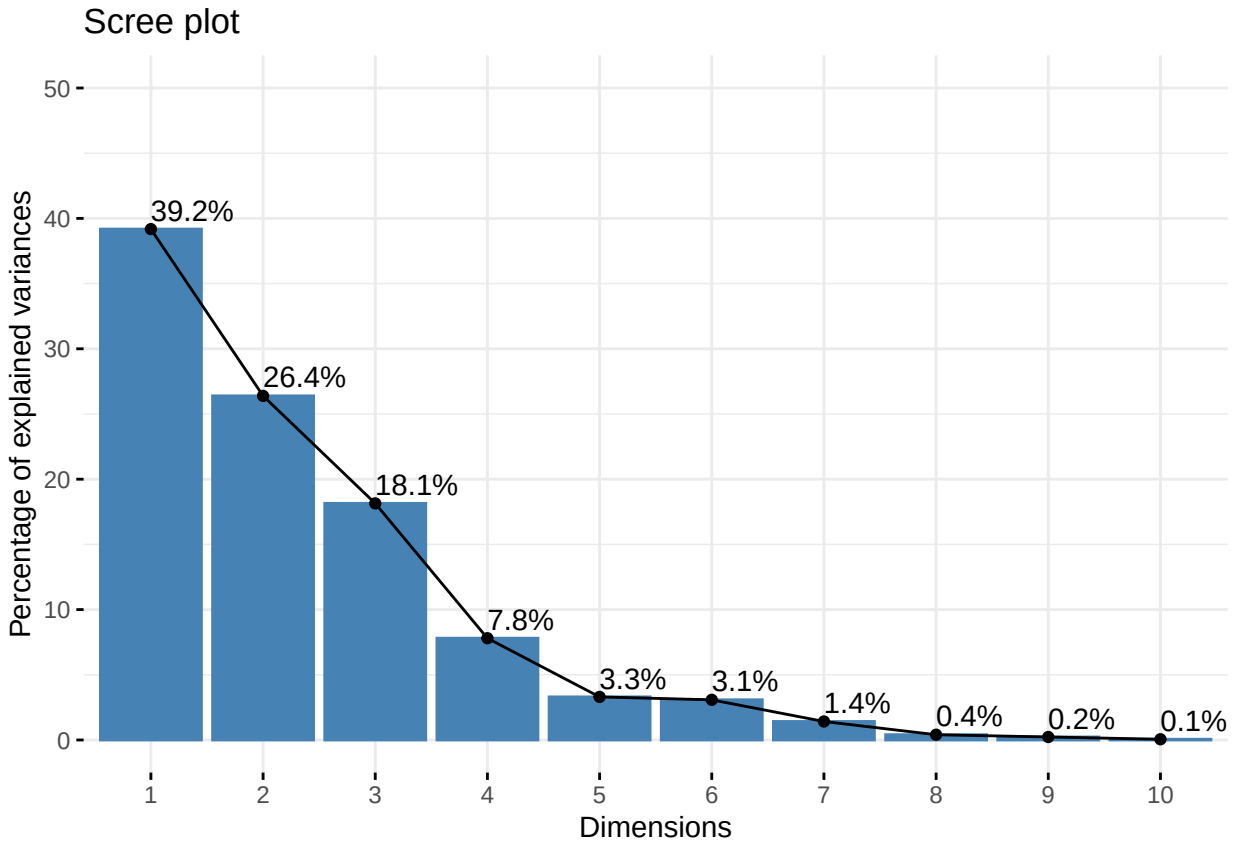
- Koji skladatelji se najviše izdvajaju po pojedinoj kategoriji?
- Koji skladatelji su određenu kategoriju skladbi pisali znatno više ili manje u usporedbi s drugim skladateljima?
- Koji skladatelji su određenu kategoriju skladbi pisali znatno više ili manje od drugih kategorija?
- Koji skladatelji su pisali prosječan broj skladbi u svim kategorijama?

```
# Vaš kod ovdje
CA(tablica, graph = TRUE)
```




```
## **Results of the Correspondence Analysis (CA)**
## The row variable has  11  categories; the column variable has 11 categories
## The chi square of independence between the two variables is equal to 500.4806 (p-value =  1.417544e-5
## *The results are available in the following objects:
##
##    name              description
## 1  "$eig"            "eigenvalues"
## 2  "$col"            "results for the columns"
## 3  "$col$coord"      "coord. for the columns"
## 4  "$col$cos2"       "cos2 for the columns"
## 5  "$col$contrib"    "contributions of the columns"
## 6  "$row"            "results for the rows"
## 7  "$row$coord"      "coord. for the rows"
## 8  "$row$cos2"       "cos2 for the rows"
## 9  "$row$contrib"    "contributions of the rows"
## 10 "$call"           "summary called parameters"
## 11 "$call$marge.col" "weights of the columns"
## 12 "$call$marge.row" "weights of the rows"
```

```
fviz_screeplot(res.ca, addlabels = TRUE, ylim = c(0, 50))
```



Kreirajte novi podskup podataka tako da sadrži **5–10 skladatelja po Vašem izboru**. Zatim ponovno **odredite** koje kategorije skladbi se najčešće pojavljuju u dobivenom podskupu, te zadržite samo skladbe koje pripadaju **prvih 5–10 kategorija**. **Izračunajte i prikažite** kontingencijsku matricu. **Prikažite** graf analize korespondencije. Pokušajte ponovo odgovoriti na gornja pitanja. Kakve nove zaključke možete izvesti?

```
# Vaš kod ovdje


limit <- 10


# Kreirajte novi podskup podataka - broj skladbi


# extract
data_kriterij <-

  data %>%
  group_by(composer) %>%
  summarize(nstars = sum(stars)) %>%
  arrange(desc(nstars)) %>%
  top_n(limit)
```

```
## Selecting by nstars
```

```
data_clean4 <-
  clean_data %>%
    filter(composer %in% data_kriterij$composer) %>%
```

```
    select(composer, category)




# odredite koje kategorije skladbi se najčešće pojavljuju u dobivenom podskupu


# extract
data_category <-

  data_clean4 %>%
  group_by(category) %>%
  summarize(ntitles = n()) %>%
  arrange(desc(ntitles)) %>%
  top_n(limit)
```

```
## Selecting by ntitles
```

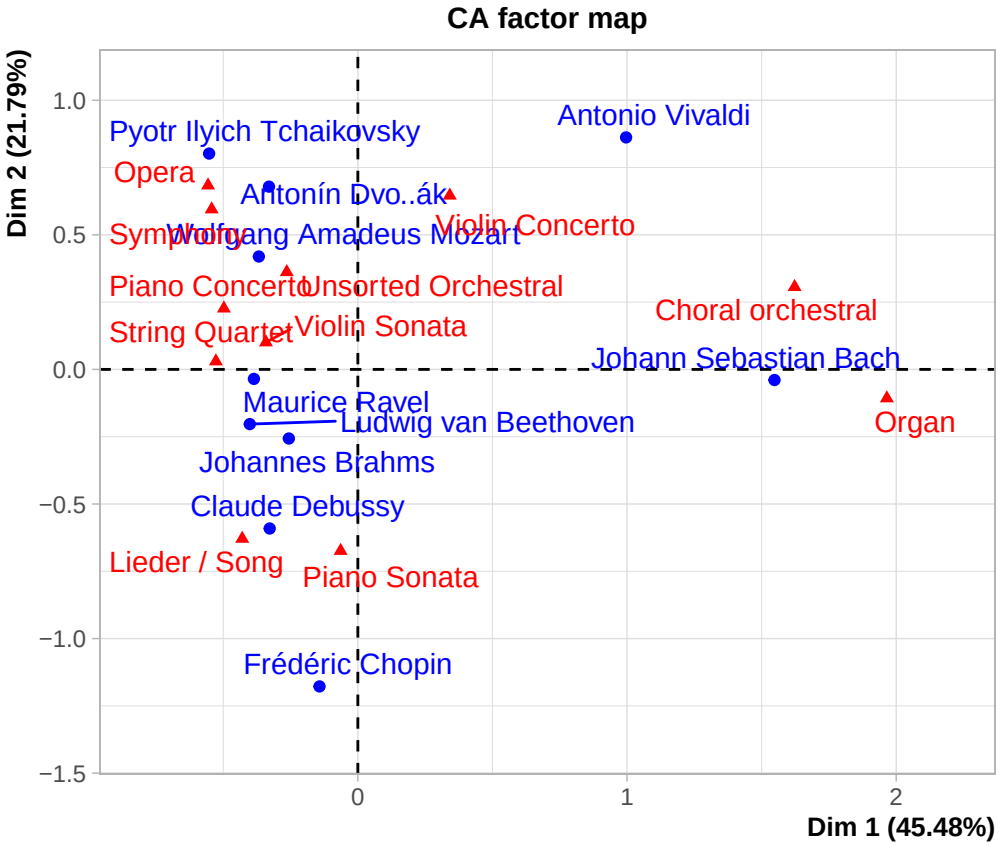
```
data_clean4 <-
  data_clean4 %>%
    filter(category %in% data_category$category)



# clean empty category variables

data_clean4$composer <- droplevels(data_clean4$composer)
data_clean4$category <- droplevels(data_clean4$category)
```

```
tablica <- as.data.frame.matrix(table(as.factor(data_clean4$composer),as.factor(data_clean4$category)))
res.ca <- CA(tablica, graph = TRUE)
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <c5>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Antonín Dvořák' in 'mbcsToSbcs': dot substituted for <99>
```




```
print(res.ca)
```

```
## **Results of the Correspondence Analysis (CA)**
## The row variable has  10  categories; the column variable has 11 categories
## The chi square of independence between the two variables is equal to 409.5948 (p-value =  2.726568e-
## *The results are available in the following objects:
##
##    name              description
## 1  "$eig"            "eigenvalues"
## 2  "$col"            "results for the columns"
## 3  "$col$coord"      "coord. for the columns"
## 4  "$col$cos2"       "cos2 for the columns"
## 5  "$col$contrib"    "contributions of the columns"
## 6  "$row"            "results for the rows"
## 7  "$row$coord"      "coord. for the rows"
## 8  "$row$cos2"       "cos2 for the rows"
## 9  "$row$contrib"    "contributions of the rows"
```

```
## 10 "$call"           "summary called parameters"
## 11 "$call$marge.col" "weights of the columns"
## 12 "$call$marge.row" "weights of the rows"
```

```
fviz_screeplot(res.ca, addlabels = TRUE, ylim = c(0, 50))
```