Analiza Skupień

Szymon Talaga

14.12.2014

Raport przedstawia procedurę klasyfikacji respondentów na podstawie poprzednio skonsturowanych skal i zmiennych. Do klasyfikacji posłuży Analiza Skupień metodą k-średnich, gdzie optymalna liczba grup zostanie wybrana w oparciu o procedurę oceny różnic wewnątrz- i międzygrupowych względem wielu metryk (Charrad, Ghazzali, Boiteau, Niknafs, 2014). Metoda ta została zaimplementowana w pakiecie *NbClust* dostępnym w środowisku R.

Wczytanie niezbędnych pakietów i funkcji oraz wczytanie zbioru danych:

```
library(NbClust)
library(lattice)
library(psych)
library(mvnormtest)
library(MASS)
library(candisc)
library(biotools)
```

```
## ---
## biotools version 1.2
```

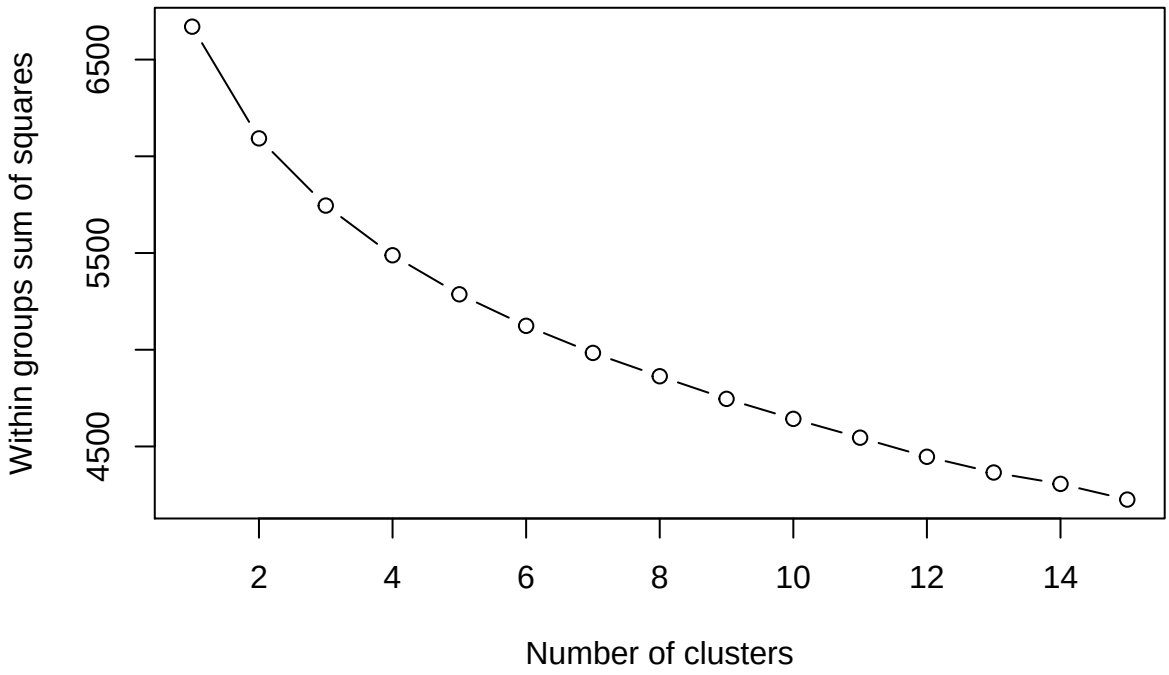
```
library(vcd)
library(knitr)
source("../DataReduction/KmeansHelper.R")
source("../HelperFunctionsMisc/ComputingMisc.R")

load("../MainData/MainData8.RData")
D.back = D
```

Przygotowanie danych do analizy skupień:

```
vars = names(D.back)[c(11:12, 29:30, 37, 44:dim(D)[2])]
D = D.back[, vars]
D.s = scale(D)
```

Wykres osypiska (na podstawie sum kwadratów wewnątrzgrupowych) w celu sprawdzenia, czy nie ma oczywistego najlepszego podziału:

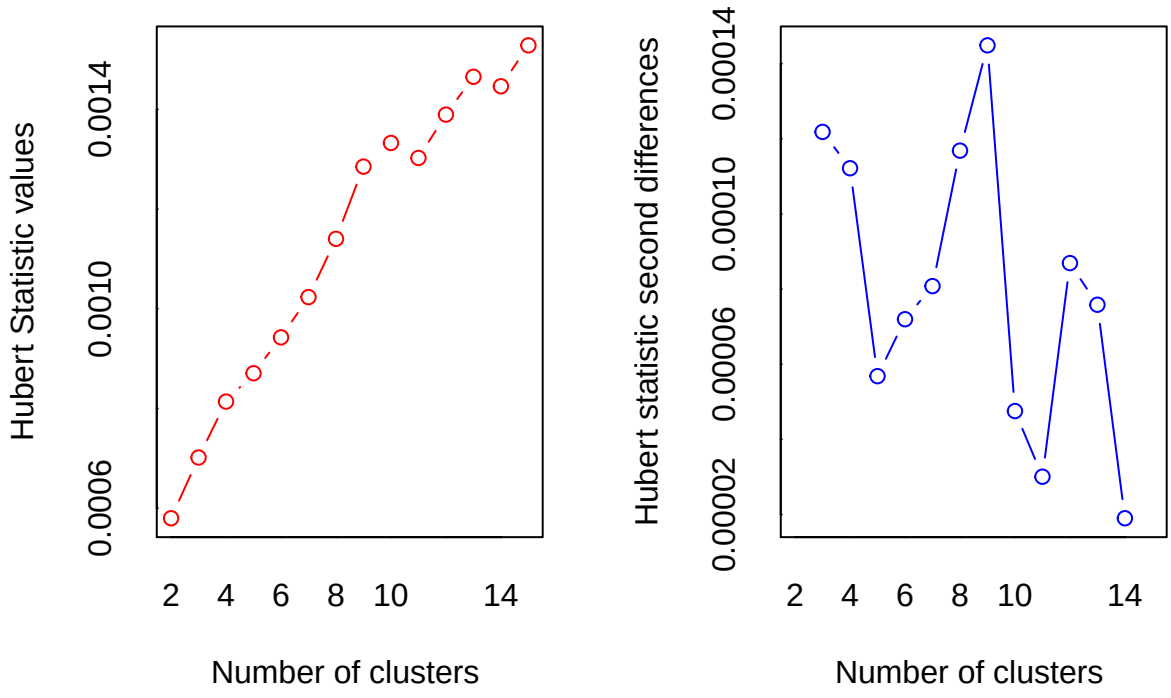


Jak widać wykres nigdzie nie wypłaszcza się gwałtownie. Oznacza to, że trzeba użyć bardziej zaawansowanej metody wyboru optymalnej liczby grup. W tym celu zostanie zastosowany algorytm NbClust, który wyznacza tę liczbę na podstawie ocen zebranych od 26 różnych metod oceny jakości podziału (opisy metod w: Charrad, Ghazzali, Boiteau, Niknafs, 2014). Dokładniej rzecz biorąc za optymalny podział zostaje uznany ten, który dostanie najwięcej głosów.

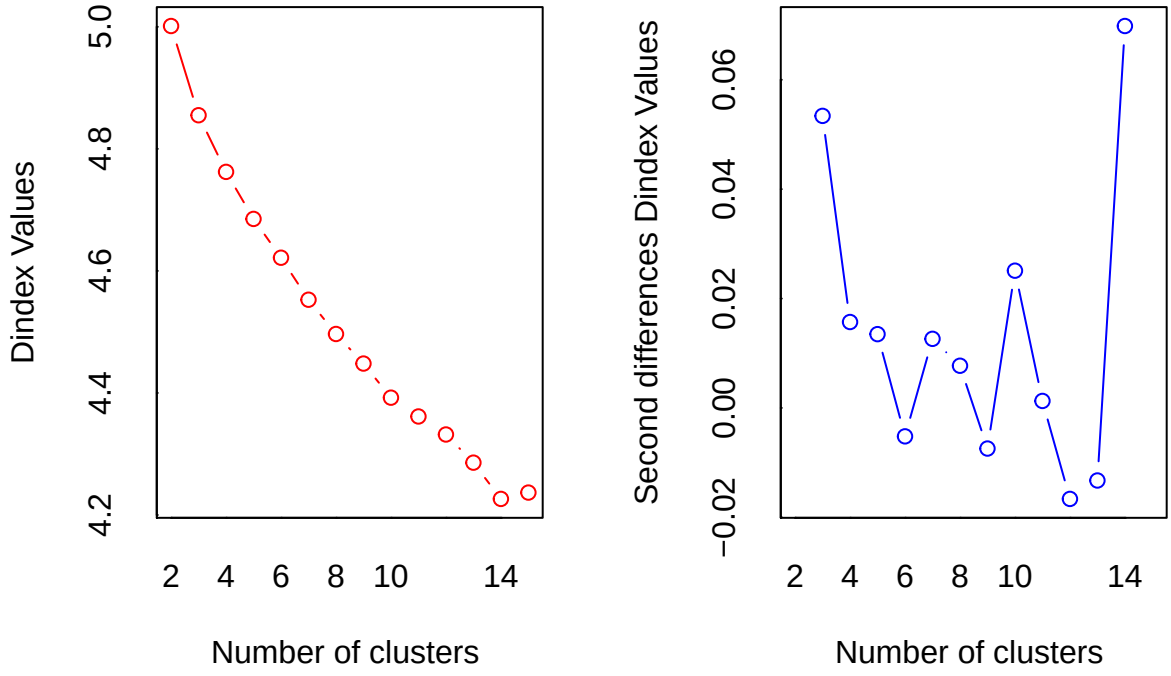
UWAGA: zaleca się, aby nie powtarzać tej części analizy, ponieważ wykonanie algorytmu NbClust zajmuje dosyć dużo czasu.

Wybór optymalnego podziału:

```
set.seed(5050) # ustawienie ziarna generatora liczb pseudolosowych
nc <- NbClust(D.s, min.nc=2, max.nc=15, method="kmeans", index="all")
```




```
## *** : The Hubert index is a graphical method of determining the number of clusters.
##             In the plot of Hubert index, we seek a significant knee that corresponds to a
##           significant increase of the value of the measure i.e the significant peak in Hubert
##                 index second differences plot.
##
```

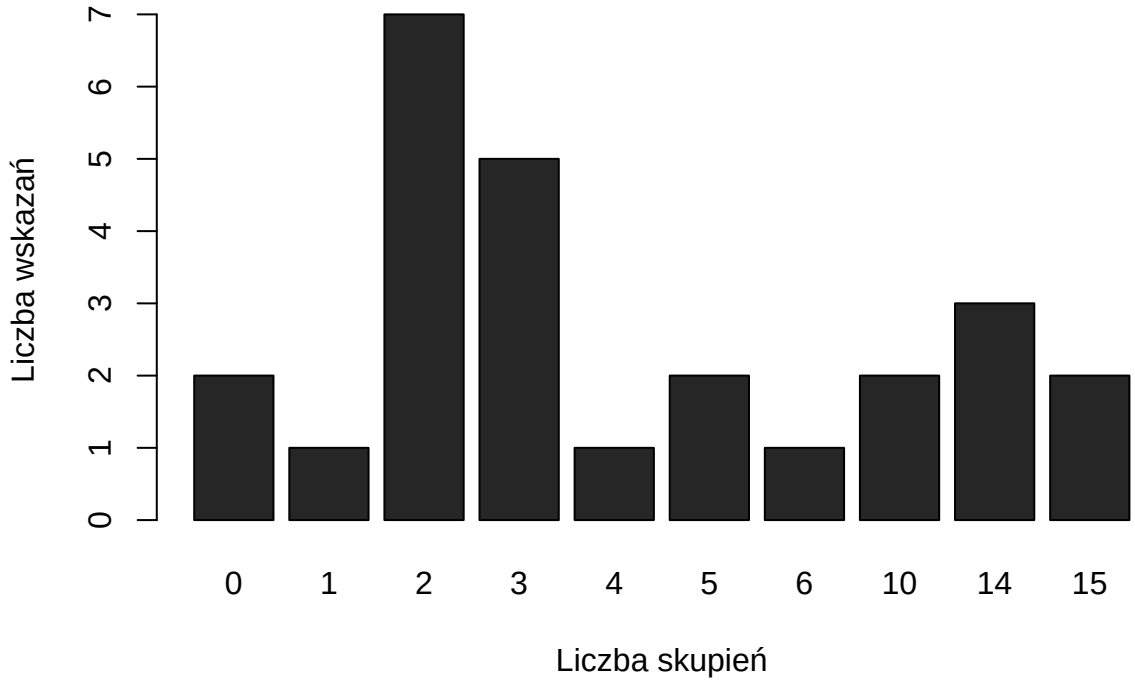



```
## *** : The D index is a graphical method of determining the number of clusters.
##           In the plot of D index, we seek a significant knee (the significant peak in Dindex
##           second differences plot) that corresponds to a significant increase of the value of
```

```
##                 the measure.
##
## All 231 observations were used.
##
## ******************************************************************
## * Among all indices:
## * 7 proposed 2 as the best number of clusters
## * 5 proposed 3 as the best number of clusters
## * 1 proposed 4 as the best number of clusters
## * 2 proposed 5 as the best number of clusters
## * 1 proposed 6 as the best number of clusters
## * 2 proposed 10 as the best number of clusters
## * 3 proposed 14 as the best number of clusters
## * 2 proposed 15 as the best number of clusters
##
##                     ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  2
##
##
## ******************************************************************
```

Jak widać procdura wskazała 2 skupienia jako optymalny punkt podzialu. Na drugim miejscu jest podział na 3 grupy. Decyduję się wybrać podział na 3 grupy, jako że do dalszych analiz bardziej przydatna będzie odrobinę bardziej zróżnicowana klasyfikacja, zaś podział na 3 skupienia wciąż można uznać za adekwatny.

Dokładne wyniki głosowania:



Dokonanie podziału:

```
set.seed(1234) # set the pseudorandom numbers generator
Kpart = kmeans(D.s, centers=3, iter.max=40, nstart=50)
```

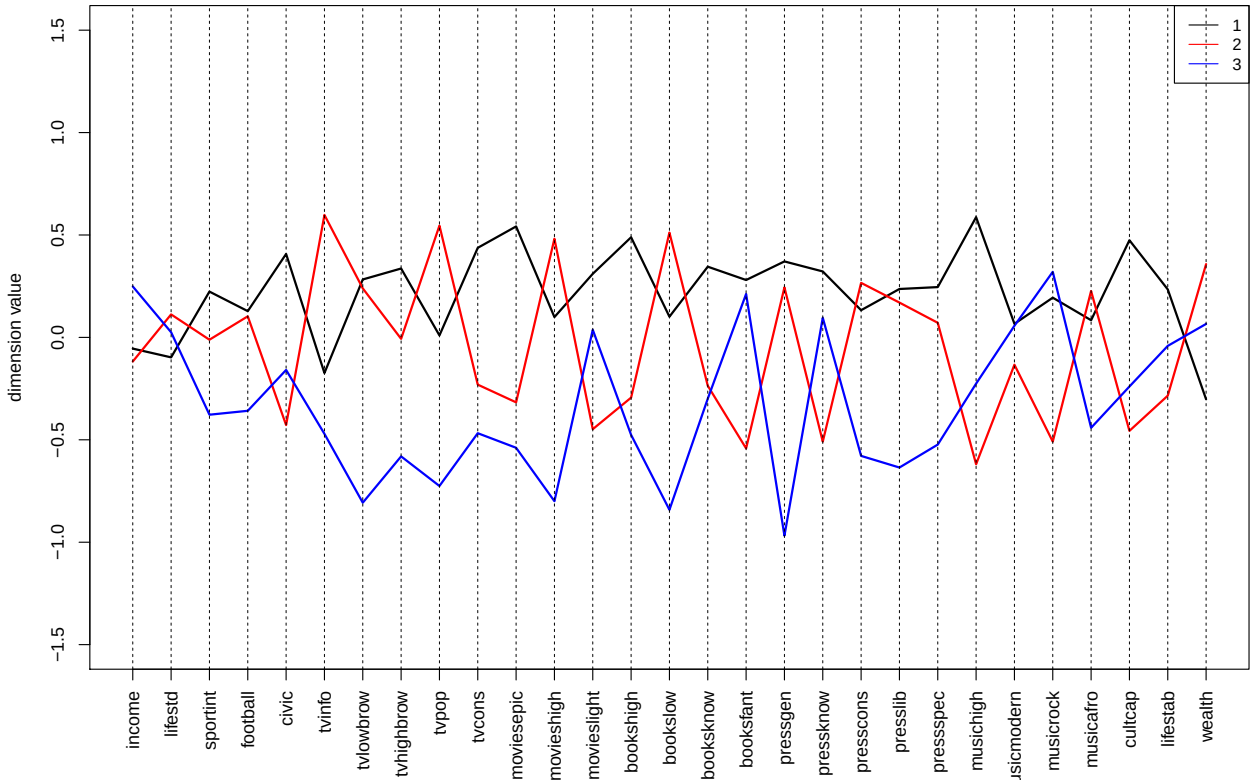
```
# Grupowanie k-średnich jest do pewnego stopnia zależne od warunków początkowych (które są losowe), dla
D.back$cluster = Kpart$cluster
```

Analiza i interpretacja skupień

Przygotowanie danych:

```
D.s = as.data.frame(D.s)
D.s$cluster = Kpart$cluster
centroids = Kpart$centers
```

Wykres profili centroidów:



Teraz zostanie przeprowadzony szereg jednoczynnikowych analiz wariancji w celu wybrania zbioru najbardziej różnicujących zmiennych. Ze względu na dużą ilość testów zostaną wybrane tylko te zmienne, które różnicują z $p \leq 0{,}01$.

```
Ftest.df = data.frame(Fval=vector(mode="numeric", length=29), pval=vector(mode="numeric", length=29))
rownames(Ftest.df) = colnames(centroids)
for(i in 1:29) {
      model = aov(D.s[,i] ~ cluster, data=D.s)
      Ftest.df[i, 1] = as.numeric(unlist(summary(model))[7])
      Ftest.df[i, 2] = as.numeric(unlist(summary(model))[9])
}
Ftest.df.sig = Ftest.df[Ftest.df$pval < 0.01, ]
kable(Ftest.df.sig)
```

	Fval	pval
sportint	13.631308	0.0002781
football	7.634218	0.0061926
civic	17.859179	0.0000344
tvlowbrow	43.956923	0.0000000
tvhighbrow	34.224097	0.0000000
tvpop	12.627018	0.0004617
tvcons	38.707161	0.0000000
moviesepic	60.990223	0.0000000
movieshigh	22.623485	0.0000035
bookshigh	46.428314	0.0000000
bookslow	25.008072	0.0000011
booksknow	19.280763	0.0000172
pressgen	75.682468	0.0000000
presscons	15.251620	0.0001238
presslib	26.728946	0.0000005
pressspec	21.634560	0.0000056
musichigh	40.242392	0.0000000
musicafro	7.772238	0.0057513
cultcap	28.005369	0.0000003
wealth	7.978536	0.0051510

Teraz zostaną sprawdzone różnice między parami średnich, żeby określić, które skupienia mają istotnie wyższe średnie w zakresie określonych zmiennych. Posłużą do tego testy t z poprawką Bonferroniego dla istotnośc.

Zainteresowanie sportem:

```
pairwise.t.test(D.s$sportint, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$sportint and D.s$cluster
##
##   1       2
## 2 0.35236 -
## 3 0.00077 0.10275
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$sportint, D.s$cluster, mean),
      odchylenie = tapply(D.s$sportint, D.s$cluster, sd)))
```

średnia	odchylenie
0.2236105	0.9988566
-0.0114077	0.9924289
-0.3774891	0.9082488

Widać, że skupienie 1 ma wyższą średnią od skupienia 3.

Zainteresowanie piłką nożną:

```
pairwise.t.test(D.s$football, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$football and D.s$cluster
##
##   1      2
## 2 1.0000 -
## 3 0.0095 0.0249
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$football, D.s$cluster, mean),
      odchylenie = tapply(D.s$football, D.s$cluster, sd)))
```

średnia	odchylenie
0.1282062	1.0187728
0.1028238	0.9633905
-0.3584137	0.9425169

Skupienia 1 i 2 mają wyższe średnie od skupienia 3.

Aktywność obywatelska:

```
pairwise.t.test(D.s$civic, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$civic and D.s$cluster
##
##   1       2
## 2 5.4e-08 -
## 3 0.00095 0.30940
##
## P value adjustment method: bonferroni
```

```r
kable(data.frame(
      średnia = tapply(D.s$civic, D.s$cluster, mean),
      odchylenie = tapply(D.s$civic, D.s$cluster, sd)))
```

średnia	odchylenie
0.4073117	1.0428914
-0.4279817	0.6248616
-0.1589565	1.0542212

Skupienie 1 wyżej od skupień 2 i 3.

Proste i niewymagające programy telewizyjne:

```r
pairwise.t.test(D.s$tvlowbrow, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$tvlowbrow and D.s$cluster
##
##   1       2
## 2 1       -
## 3 8.9e-12 5.7e-10
##
## P value adjustment method: bonferroni
```

```r
kable(data.frame(
      średnia = tapply(D.s$tvlowbrow, D.s$cluster, mean),
      odchylenie = tapply(D.s$tvlowbrow, D.s$cluster, sd)))
```

średnia	odchylenie
0.2828878	0.836629
0.2393272	0.759090
-0.8069999	1.112261

Skupienia 1 i 2 wyżej od 3.

Bardziej wymagające programy telewizyjne:

```r
pairwise.t.test(D.s$tvhighbrow, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$tvhighbrow and D.s$cluster
```

```
##
##   1       2
## 2 0.0524  -
## 3 3.7e-08 0.0018
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$tvhighbrow, D.s$cluster, mean),
      odchylenie = tapply(D.s$tvhighbrow, D.s$cluster, sd)))
```

średnia	odchylenie
0.3363518	0.7533082
-0.0070265	0.7491725
-0.5809688	1.3515361

Główne popularne stacje telewizyjne w Polsce:

```
pairwise.t.test(D.s$tvpop, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$tvpop and D.s$cluster
##
##   1       2
## 2 3e-04   -
## 3 3.2e-06 6.6e-14
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$tvpop, D.s$cluster, mean),
      odchylenie = tapply(D.s$tvpop, D.s$cluster, sd)))
```

średnia	odchylenie
0.0096866	0.8865802
0.5456890	0.6618836
-0.7254325	1.1032565

Skupienie 1 > Skupienie 2 > Skupienie 3

Telewizje konserwatywne:

```r
pairwise.t.test(D.s$tvcons, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$tvcons and D.s$cluster
##
##   1       2
## 2 1.3e-05 -
## 3 3.8e-08 0.44
##
## P value adjustment method: bonferroni
```

```r
kable(data.frame(
      średnia = tapply(D.s$tvcons, D.s$cluster, mean),
      odchylenie = tapply(D.s$tvcons, D.s$cluster, sd)))
```

średnia	odchylenie
0.4372773	0.8543152
-0.2306025	0.8922248
-0.4677745	1.0699051

Skupienie 1 wyżej niż skupienia 2 i 3.

Filmy fantasy, science-fiction etc.

```r
pairwise.t.test(D.s$moviesepic, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$moviesepic and D.s$cluster
##
##   1       2
## 2 3.2e-09 -
## 3 7.9e-12 0.46
##
## P value adjustment method: bonferroni
```

```r
kable(data.frame(
      średnia = tapply(D.s$moviesepic, D.s$cluster, mean),
      odchylenie = tapply(D.s$moviesepic, D.s$cluster, sd)))
```

średnia	odchylenie
0.5417827	0.672579
-0.3170292	0.919410
-0.5389142	1.119337

Skupienie 1 wyżej od skupień 2 i 3.

Ambitne kino:

```
pairwise.t.test(D.s$movieshigh, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$movieshigh and D.s$cluster
##
##   1       2
## 2 0.015   -
## 3 9.0e-09 2.9e-14
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$movieshigh, D.s$cluster, mean),
      odchylenie = tapply(D.s$movieshigh, D.s$cluster, sd)))
```

średnia	odchylenie
0.0992861	0.8031824
0.4817661	0.7332705
-0.7996369	1.1360639

Skupienie 2 > Skupienie 1 > Skupienie 3

Ambitna literatura:

```
pairwise.t.test(D.s$bookshigh, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$bookshigh and D.s$cluster
##
##   1       2
## 2 1.5e-07 -
## 3 2.3e-09 0.77
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$bookshigh, D.s$cluster, mean),
      odchylenie = tapply(D.s$bookshigh, D.s$cluster, sd)))
```

średnia	odchylenie
0.4888076	0.7127576
-0.2940171	0.8295525
-0.4758508	1.2412060

Skupienie 1 wyżej od skupień 2 i 3.

Prosta literatura:

```
pairwise.t.test(D.s$bookslow, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$bookslow and D.s$cluster
##
##   1       2
## 2 0.0064  -
## 3 8.8e-10 4.8e-16
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$bookslow, D.s$cluster, mean),
      odchylenie = tapply(D.s$bookslow, D.s$cluster, sd)))
```

średnia	odchylenie
0.1010902	0.8330907
0.5117213	0.7463504
-0.8416912	1.0337640

Skupienie 2 > Skupienie 1 > Skupienie 3

Literatura popularno-naukowa, faktu itp.

```
pairwise.t.test(D.s$booksknow, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$booksknow and D.s$cluster
##
##   1     2
## 2 3e-04 -
## 3 2e-04 1e+00
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$booksknow, D.s$cluster, mean),
      odchylenie = tapply(D.s$booksknow, D.s$cluster, sd)))
```

średnia	odchylenie
0.3452781	0.7399451
-0.2360608	1.0366123
-0.2992862	1.1665077

Skupienie 1 wyżej od skupień 2 i 3.

Ogólne czytanie prasy:

```
pairwise.t.test(D.s$pressgen, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$pressgen and D.s$cluster
##
##   1       2
## 2 0.97    -
## 3 < 2e-16 3.8e-14
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$pressgen, D.s$cluster, mean),
      odchylenie = tapply(D.s$pressgen, D.s$cluster, sd)))
```

średnia	odchylenie
0.3709190	0.7593871
0.2443052	0.7352502
-0.9679032	1.0511222

Skupienia 1 i 2 wyżej od skupienia 3.

Prasa konserwatywna:

```
pairwise.t.test(D.s$presscons, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$presscons and D.s$cluster
```

```
##
##   1       2
## 2 1       -
## 3 2.8e-05 2.5e-06
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$presscons, D.s$cluster, mean),
      odchylenie = tapply(D.s$presscons, D.s$cluster, sd)))
```

średnia	odchylenie
0.1328097	0.9498931
0.2661674	0.8735845
-0.5785502	1.0253678

Skupienia 1 i 2 wyżej od 3.

Standardowe popularne tytuły prasowe w Polsce:

```
pairwise.t.test(D.s$presslib, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$presslib and D.s$cluster
##
##   1       2
## 2 1       -
## 3 1.7e-07 5.8e-06
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$presslib, D.s$cluster, mean),
      odchylenie = tapply(D.s$presslib, D.s$cluster, sd)))
```

średnia	odchylenie
0.2363357	0.8940501
0.1700154	0.8410460
-0.6353458	1.1059603

Skupienia 1 i 2 wyżej od 3.

Praca specjalistyczna (hobbystyczna, branżowa etc.)

```
pairwise.t.test(D.s$pressspec, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$pressspec and D.s$cluster
##
##   1       2
## 2 0.7053  -
## 3 6.8e-06 0.0015
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$pressspec, D.s$cluster, mean),
      odchylenie = tapply(D.s$pressspec, D.s$cluster, sd)))
```

średnia	odchylenie
0.2455826	0.7906254
0.0712382	0.9863041
-0.5233314	1.1579277

Skupienia 1 i 2 wyżej od 3.

Ambitna muzyka:

```
pairwise.t.test(D.s$musichigh, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$musichigh and D.s$cluster
##
##   1       2
## 2 < 2e-16 -
## 3 7.7e-08 0.027
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$musichigh, D.s$cluster, mean),
      odchylenie = tapply(D.s$musichigh, D.s$cluster, sd)))
```

średnia	odchylenie
0.5869787	0.6821104
-0.6192893	0.8683364
-0.2257976	1.0619581

Skupienie 1 > Skupienie 3 > Skupienie 2.

Muzyka czarna / afroamerykańska itp.:

```
pairwise.t.test(D.s$musicafro, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$musicafro and D.s$cluster
##
##   1       2
## 2 1.00000 -
## 3 0.00381 0.00037
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$musicafro, D.s$cluster, mean),
      odchylenie = tapply(D.s$musicafro, D.s$cluster, sd)))
```

średnia	odchylenie
0.0838642	0.9218598
0.2265785	0.9420895
-0.4412847	1.0812968

Skupienia 1 i 2 wyżej od 3.

Kapitał kulturowy:

```
pairwise.t.test(D.s$cultcap, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$cultcap and D.s$cluster
##
##   1       2
## 2 5.8e-10 -
## 3 1.2e-05 0.54
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$cultcap, D.s$cluster, mean),
      odchylenie = tapply(D.s$cultcap, D.s$cluster, sd)))
```

średnia	odchylenie
0.4743648	0.7638881
-0.4562790	0.9880112
-0.2398567	1.0332275

Skupienie 1 wyżej od 2 i 3.

Posiadanie:

```
pairwise.t.test(D.s$wealth, D.s$cluster, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  D.s$wealth and D.s$cluster
##
##   1       2
## 2 3.7e-05 -
## 3 0.066   0.259
##
## P value adjustment method: bonferroni
```

```
kable(data.frame(
      średnia = tapply(D.s$wealth, D.s$cluster, mean),
      odchylenie = tapply(D.s$wealth, D.s$cluster, sd)))
```

średnia	odchylenie
-0.3022007	0.9541494
0.3577445	0.9054866
0.0657364	1.0451223

Skupienie 2 wyżej od skupienia 1.

Podsumowanie wyników powyższej analizy przedstawia poniższa tabela. Wartości 1 oznaczają przewagę danego skupienia w pewnym zakresie nad tymi skupieniami, które mają wartość -1. Wartość 0 odpowiada pozycji pośredniej pod względem danej zmiennej.

```
c1 = c(1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,1,1,1,-1)
c2 = c(0,1,-1,1,1,0,1,-1,1,-1,1,-1,1,1,1,1,-1,1,-1,1)
c3 = c(-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,0,-1,-1,0)
Ftest.df.sig = cbind(Ftest.df.sig, c1,c2,c3)
kable(Ftest.df.sig[,3:5])
```

	c1	c2	c3
sportint	1	0	-1

	c1	c2	c3
football	1	1	-1
civic	1	-1	-1
tvlowbrow	1	1	-1
tvhighbrow	1	1	-1
tvpop	1	0	-1
tvcons	1	1	-1
moviesepic	1	-1	-1
movieshigh	0	1	-1
bookshigh	1	-1	-1
bookslow	0	1	-1
booksknow	1	-1	-1
pressgen	1	1	-1
presscons	1	1	-1
presslib	1	1	-1
pressspec	1	1	-1
musichigh	1	-1	0
musicafro	1	1	-1
cultcap	1	-1	-1
wealth	-1	1	0

Teraz można spróbować przedstawić odpowiednią interpretację otrzymanych skupień.

Interpretacja skupień

Na początek jednak zostanie przedstawionch kilka dodatkowych analiz.

Skupienie a prestiż zawodu (uproszczona klasyfikacja ISCO):

```
ISCOtab = table(D.back$cluster, D.back$ISCObroad)
kable(ISCOtab)
```

no_job	low:phys/sale/serv/assoc_pro	high:mng/pro/self
27	28	45
44	13	17
27	7	23

```
assocstats(ISCOtab)
```

```
##                     X^2 df   P(> X^2)
```

```
## Likelihood Ratio 22.569  4 0.00015434
## Pearson          21.723  4 0.00022749
##
## Phi-Coefficient   : 0.307
## Contingency Coeff.: 0.293
## Cramer's V        : 0.217
```

Ewidentnie w skupieniu pierwszym przeważają osoby posiadające pracę i to raczej bardziej prestiżową. W skupieniu drugim duży udział mają osoby niepracujące, a w trzecim niepracujące, lub posiadające raczje dobrą pracę. Jak pokazują wyniki testów zależnośc między zmiennymi jest umiarkowanie silna (wartość V-Crammera) i istotna.

Skupienie a rodzaj wykształcenia:

```
eduprogtab = table(D.back$cluster, D.back$eduprog)
kable(eduprogtab)
```

soc/beh	no_uni_edu	law/biz/menag	human/lang	STEM/med
27	2	18	39	14
20	4	23	14	13
14	5	6	13	19

```
assocstats(eduprogtab)
```

```
##                     X^2 df  P(> X^2)
## Likelihood Ratio 24.244  8 0.0020859
## Pearson          24.850  8 0.0016482
##
## Phi-Coefficient   : 0.328
## Contingency Coeff.: 0.312
## Cramer's V        : 0.232
```

Jak widać w skupieniu pierwzym przeważają osoby o wykształceniu humanistycznym/społecznym. Skupienie drugie charakteryzuje się przewagą osób z wykształceniem w zakresie nauk społecznych lub ekonomiczno-menadżersko-biznesowych. W grupie trzeciej przeważają (zarówno względnie jak i bezwzględnie) osoby o wykształceniu ścisłym.

Skupienie a wymiar pracy:

```
worktimetab = table(D.back$cluster, D.back$worktime)
kable(worktimetab)
```

no_job	job	flexible
26	31	43
44	21	9
26	16	15

```
assocstats(worktimetab)
```

```
##                     X^2 df   P(> X^2)
## Likelihood Ratio 27.422  4 1.6330e-05
## Pearson          26.036  4 3.1118e-05
##
## Phi-Coefficient   : 0.336
## Contingency Coeff.: 0.318
## Cramer's V        : 0.237
```

Widać, że w skupieniu pierwszym przeważają osoby o nieregulowanym czasie pracy, a w drugim osoby niepracujące (co już wiadomo po poprzedniej analizie). W skupieniu drugi oba rodzaje pracy rozkładają się po równo i są porównywalnie często co brak pracy.

Skupienie I (n = 100)

Wolne Zawody : do tego skupienia należą przede wszystkim osoby o wykształceniu humanistycznym/społecznym, które posiadają pracę i jest to często praca o nieregulowanym czasie. Charakteryzują się podwyższonym profilu konsumpcji i udziału w kulturze. Jednocześnie występuje u nich trend do odrobinę rzadszego posiadania dóbr takich jak własne mieszkanie, samochód itp.

Skupienie II (n = 74)

Studenci : nazwa nie jest zapewne idealna, ale dosyć dobrze oddaje charakter tej grupy. Skupia ona przede wszystkim osoby bez pracy, charakteryzuje się delikatną przewagą studentów kierunków ekonomicznych, biznesowych i menadżerskich. Ma mieszany profil udziału i konsumpcji kultury - niektóre rzeczy są w tej grupiep popularne, inne nie i czasem są to dzieła ambitne, a czasem nie. Grupa charakteryzuje się również lekko podwyższonym wskaźnikiem posiadania, co oznacza, że jej członkowie odrobinę częściej posiadają własne mieszkania, samochody, tablety itp. Sugeruje to, że dosyć typowym reprezentatem tej grupy mogą być studenci kierunków biznesowo-ekonomicznym, którzy pochodzą z rodzin z zamożniejszej części klasy średniej, przez co mają ułatwiony start - np. dostają od rodziny samochód bądź mieszkanie.

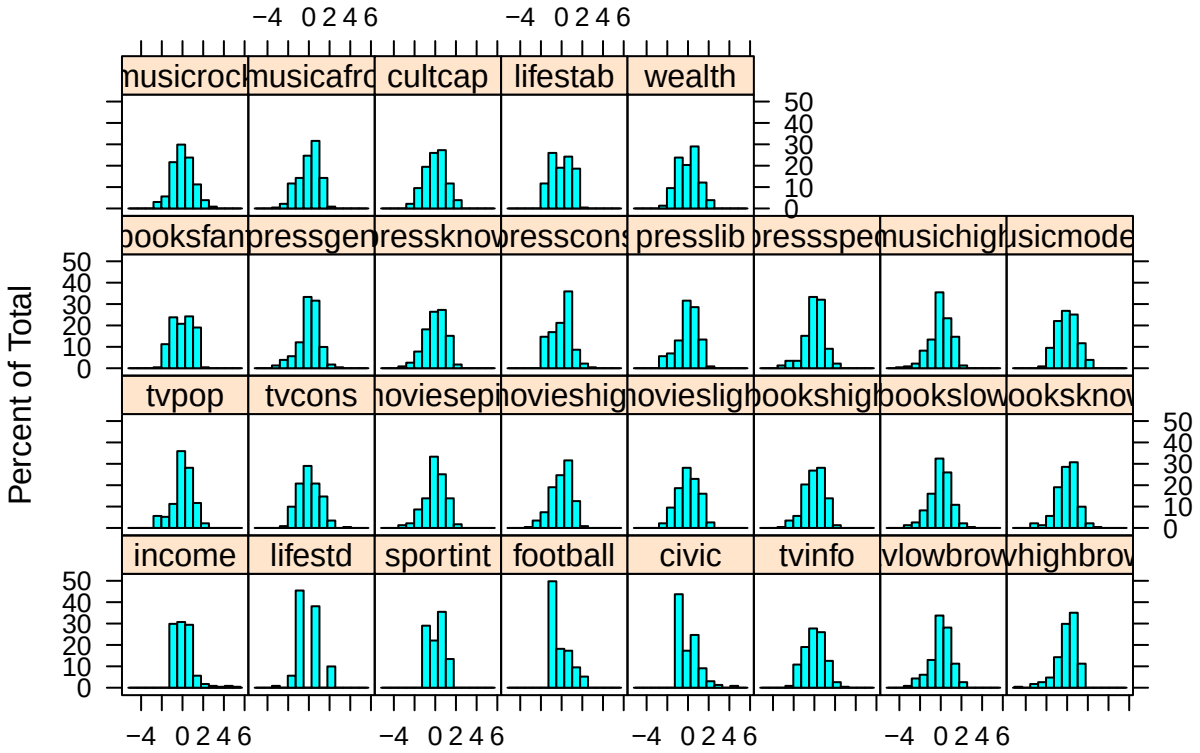
Skupienie III (n = 57)

Kulturalnie Wycofani : grupa ta skupia osoby o obniżonym profilu uczestnictwa i konsumpcji kultury. Charakteryzuje się ona również przewagą osób o wykształceniu ścisłym. Należą do niej zarówno osoby pracujące i niepracujące.

Na koniec zostanie przeprowadzona Liniowa Analiza Dyskryminacji, co da jeszcze bardziej pogłębiony wgląd w różnice między skupieniami.

LDA

Rozkłady wszystkich zmiennych użytych w analizie skupień:



Jak widać większość zmiennych jest względnie symetryczna i ma rozkłady zbliżone do normalnego.

Dodanie skupień jako zmiennej do zbiorów danych:

```
cluster = D.s$cluster
cluster[cluster==1] = "Wolne_Zawody"
cluster[cluster==2] = "Studenci"
cluster[cluster==3] = "Kulturalnie_wycofani"
cluster = as.factor(cluster)
cluster = factor(cluster, levels(cluster)[c(3,2,1)])
D.s$cluster = cluster
D.back$cluster = cluster
```

Test wielowymiarowej normalności rozkładu zmiennych:

```
mshapiro.test(as.matrix(t(D.s[,1:29])))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.7735, p-value < 2.2e-16
```

Zmienne nie mają wielowymarowego rozkładu normalnego. Nie jest to zaskoczenie bo zmiennych jest dosyć dużo - aż 29.

Test M Boxa równości macierzy wariancji-kowariancji:

```r
boxM(D.s[,1:29], D.s$cluster)
```

```
##
##  Box's M-test for Homogeneity of Covariance Matrices
##
## data:  D.s[, 1:29]
## Chi-Sq (approx.) = 1096.376, df = 870, p-value = 2.42e-07
```

Macierze nie są homogeniczne. To też nie powinno dziwić, ponieważ: 1) jest duża zmiennych; 2) nie ma wielowymiarowego rozkładu normalnego, a tym przypadku test Boxa prawie zawsze daje istotne wyniki.

Pomimo braku pewności co do spełnienia założeń analiza dyskryminacyjna zostanie przeprowadzona. Jej cel jest głównie eksploracyjny, więc nawet jeżeli wyniki będą w jakiś sposób lekko zniekształcone, to nie będzie to miało wpływu na dalsze analizy.

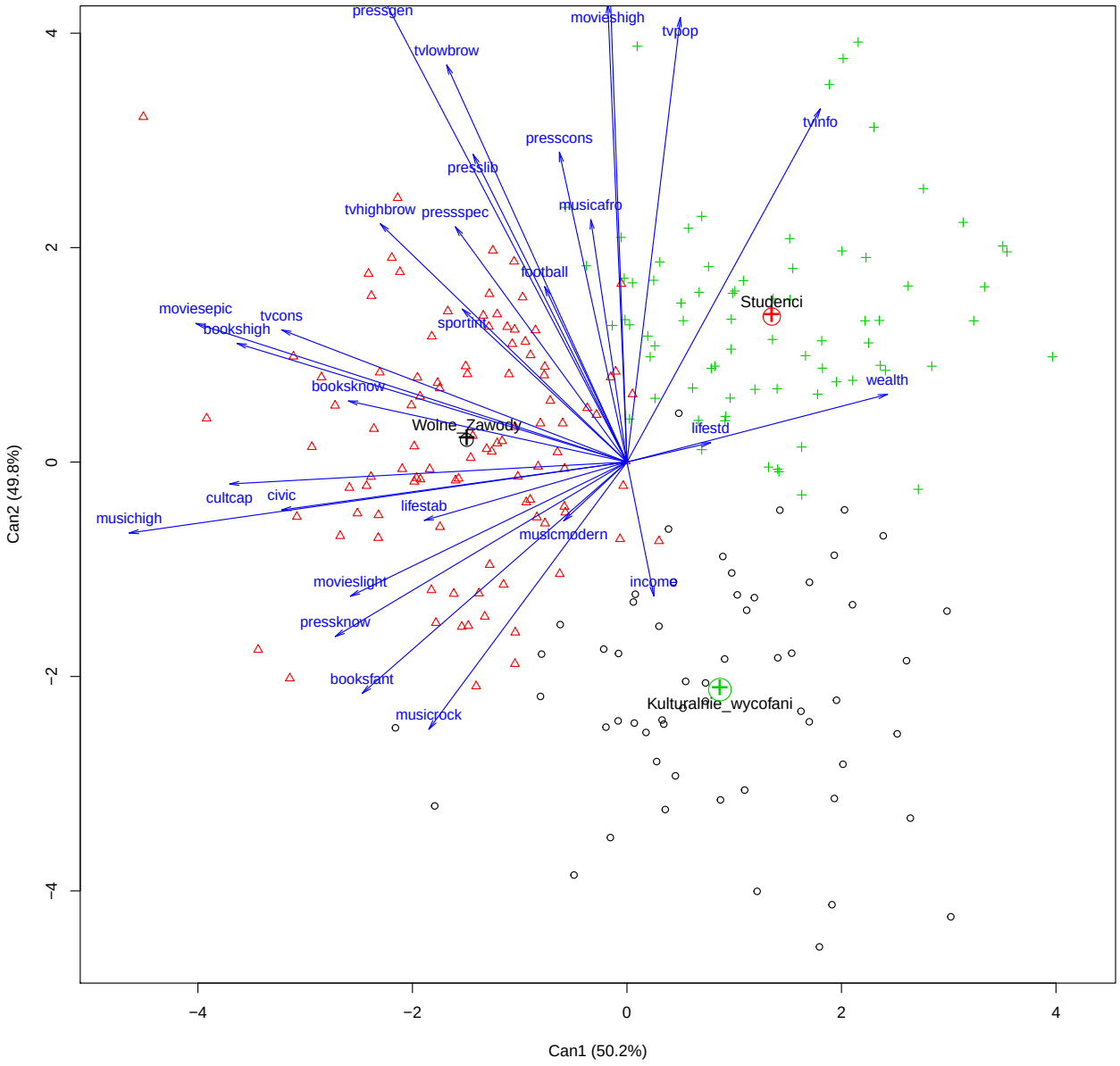
```r
manova = manova(as.matrix(D.s[,1:29]) ~ D.s$cluster)
lda1 = candisc(manova)
summary(lda1)
```

```
##
## Canonical Discriminant Analysis for D.s$cluster:
##
##    CanRsq Eigenvalue Difference Percent Cumulative
## 1 0.63730     1.7571   0.014346  50.205     50.205
## 2 0.63541     1.7428   0.014346  49.795    100.000
##
## Class means:
##
##                          Can1     Can2
## Wolne_Zawody         -1.49307  0.20572
## Studenci              1.35008  1.35708
## Kulturalnie_wycofani  0.86669 -2.12272
##
##  std coefficients:
##                    Can1       Can2
## income        0.1910408 -0.1275841
## lifestd       0.2557695  0.0616579
## sportint     -0.0713600 -0.1033593
## football      0.0089165  0.1508071
## civic        -0.2583898 -0.0059802
## tvinfo        0.1726008  0.0706178
## tvlowbrow    -0.2112494  0.3426051
## tvhighbrow   -0.0676308  0.1431254
## tvpop         0.1537042  0.0404011
## tvcons       -0.3383723  0.0564613
## moviesepic   -0.3536090  0.0995497
## movieshigh   -0.0588803  0.3257187
## movieslight -0.0800754 -0.1306317
## bookshigh     0.0277903  0.2080644
## bookslow      0.1369966  0.4374805
## booksknow    -0.0064131 -0.0020443
## booksfant    -0.3425577 -0.1500579
## pressgen     -0.0792624  0.2561039
```

```
## pressknow   -0.0509866 -0.1114493
## presscons    0.0796528  0.0977695
## presslib    -0.1979849  0.0666020
## pressspec   -0.1412031  0.2256216
## musichigh   -0.3505108 -0.2578368
## musicmodern -0.0047876 -0.1557049
## musicrock    0.0117292 -0.3291947
## musicafro   -0.0940089  0.1864152
## cultcap     -0.3409368 -0.0941472
## lifestab    -0.4044113  0.0456437
## wealth       0.2259147  0.0685042
```

Obie kanoniczne funckje dyskryminacyjne są istotne i tłumaczą zbliżone odsetki wariancji (co oznacza, że obie są tak samo istotne dla rozróżniania pomiędzy skupieniami). Jako, że czasem jeden obraz może być wart więcej niż tysiąc słów (czy liczb), inspekcja korelacji między zmiennymi a funkcjami dyskryminacyjnymi zostanie dokonana w oparciu o graficzną reprezentację modelu:

```
plot(lda1, cex=.9)
```




```
## Vector scale factor set to  7
```

Warto również spojrzeć na to, jak dużą część zmienności wyników dla funkcji kanonicznych tłumaczy przynależność grupowa. W tym celu wystarczy sprawdzić stosunek korelacyjny η^2 dla modelu manova, na którym oparta była analiza dyskryminacyjna.

```
etasq(manova, partial=FALSE, method="pillai")
```

```
##                  eta^2
## D.s$cluster 0.6363549
```

Jak widać przynależność do skupieni tłumaczy prawie 64% zmienności w 29-wymiarowej przestrzeni zmiennych, co należy uznać za bardzo dobry wynik. Świadczy to pozytywnie o dokonanej klasyfikacji respondentów i jest dowodem na jej adekwatność wobec rzeczywiście występujących różnic.

Na zakończenie warto przyjrzeć się jeszcze raz współczynnikom strukturalnym zmiennych, czyli ich prostym korelacjom z funkcjami dyskryminacyjnymi. Pozwoli to przedstawić ostateczną interpretację funkcji. Dla ułatwienia podane zostaną tylko te zmienne, które korelują z przynajmniej jedną z funkcji na poziomie 0,40 lub większym.

```
structure = as.data.frame(lda1$structure)
structure = with(structure, structure[abs(Can1) >= 0.4 | abs(Can2) >= 0.4, ])
kable(round(structure, 2))
```

	Can1	Can2
civic	-0.46	-0.06
tvinfo	0.26	0.47
tvlowbrow	-0.24	0.53
tvpop	0.07	0.59
tvcons	-0.46	0.18
moviesepic	-0.57	0.18
movieshigh	-0.03	0.61
bookshigh	-0.52	0.16
bookslow	-0.02	0.64
pressgen	-0.33	0.62
presscons	-0.09	0.41
presslib	-0.21	0.41
musichigh	-0.66	-0.09
cultcap	-0.53	-0.03

Widać, że najsilniej korelują (ujemnie) z pierwszą z funkcji zmienne takie jak: tvcons, moviesepic, bookshigh, musichigh i cultcap. Oznacza to, że funkcja ta w najogólniejszym sensie odpowiada silnemu klasycznemu kapitałowi kulturowemu, być może z lekka tendencją do preferowania bardziej tradycyjnego stylu życia. Należy przy tym pamiętać, że skala ta jest odwrócona, więc ujemne wyniki odpowiadają wysokiemu kapitałowi kulturowemu.

Druga kanoniczna funckja dyskryminacyjna koreluje w większym stopniu z tym, co można by nazwać bardzie standardowym i popularnym gustem. Jedynie silna wkład zmiennej movieshigh wydaje się z tego wyłamywać. Jednakże już wcześniej dało się zauważyć, że zmienna ta ma bardzo często przeciwne kierunki korelacji w stosunku do bookshigh i movieshigh, co sugeruje, że kino, uchodzące za ambitne, pełni odrobinę inną funkcję od literatury czy muzyki.

Na koniec warto sprawdzić, jakie są średnie centroidów w zakresie wyników dla funckji dyskryminacyjnych:

```
kable(lda1$means)
```

	Can1	Can2
Wolne_Zawody	-1.4930750	0.2057164
Studenci	1.3500838	1.3570757

	Can1	Can2
Kulturalnie_wycofani	0.8666894	-2.1227235

Wyniki można właściwie uznać za doskonale zgodne z przedstawionymi wcześniej interpretacjom. Skupienie pierwsze charakteryzuje się wysokim klasycznym kapitałem kulturowym (ujemne Can1) i niskim natężeniem gustu popularnego (niskie Can2).

Grupa Studenci ma niski kapitał kulturowy i silniejszą preferencje do gustu popularnego. Pozawala to wprowadzić pewną poprawkę do interpretacji tego skupienia, którą dało się zresztą przewidzieć już wcześniej: tak naprawdę do skupienie zdaje się w większym stopniu odpowiadać grupię, którą można by nazwać młodą klasą średnią (przed wejściem na rynek pracy).

Skupienie Kulturalnie_wycofani ma nienajwyższy kapitał kulturowy (ale wyższy od młodej klasy średniej) i zdecydowanie bardzo małe zainteresowanie kulturą popularną, co jest w zgodzie z wcześniejszą interpretacją.