

*Multivariate Analysis for the Behavioral Sciences,*  
Second Edition (Chapman and Hall/CRC, 2019)

**Examples of Chapter 14:  
Multidimensional Scaling and  
Correspondence Analysis**

*Kimmo Vehkalahti and Brian S. Everitt*

*10 December 2018*

**Contents**

<b>Examples</b>	<b>2</b>
<b>Table 14.1: Birth and Death Rates for Seven Countries</b>	<b>2</b>
Table 14.2 . . . . .	3
Table 14.4 . . . . .	4
Figure 14.1 . . . . .	5
<b>Table 14.3: Dissimilarity Data for All Pairs of 10 Colas for 2 Subjects</b>	<b>6</b>
<b>Table 14.5: Road Distances of 15 Selected Places in Finland</b>	<b>7</b>
Table 14.6 . . . . .	9
Figure 14.2 . . . . .	10
Figure 14.3 . . . . .	13
<b>Table 14.7: Proximity Matrix of Ten Remarkable Classical Music Composers Selected and Compared by Olli Mustonen</b>	<b>14</b>
Figure 14.4 . . . . .	15
Figure 14.5 . . . . .	17
Figure 14.6 . . . . .	19
Figure 14.7 . . . . .	20
<b>Table 14.8: The Influence of Age of Relationship Status with Boyfriends</b>	<b>22</b>
Figure 14.8 . . . . .	24
<b>Table 14.9: Types of Work Activities and Main Advantages of Job from 6933 Survey Respondents</b>	<b>27</b>
Figure 14.9 . . . . .	29

## Examples

**Table 14.1: Birth and Death Rates for Seven Countries**

**\*\*Source of data:\*\*** <https://population.un.org/wpp/Download/Standard/Population/>

Countries selected manually to be the same ones as in **Chapter 17**.

```
country7 <- read.table("data/country7.txt", sep = '\t', header = TRUE)
country7
```

##	BirthR	DeathR
## Japan	8.4	9.9
## Italy	8.6	10.2
## Spain	9.4	8.6
## United Kingdom	12.4	9.0
## Finland	10.8	9.5
## Cuba	11.2	7.6
## United States	12.5	8.2

Table 14.2

```
options(digits = 2)
dist7 <- dist(country7)
dist7
```

```
##           Japan           Italy           Spain
## Italy           0.36
## Spain           1.64           1.79
## United Kingdom  4.10           3.98           3.03
## Finland         2.43           2.31           1.66
## Cuba            3.62           3.68           2.06
## United States   4.44           4.38           3.13
##           United Kingdom  Finland           Cuba
## Italy
## Spain
## United Kingdom
## Finland           1.68
## Cuba              1.84           1.94
## United States     0.81           2.14           1.43
```

Table 14.4

```
cscal7 <- cmdscale(dist7, k = 2)
cscal7
```

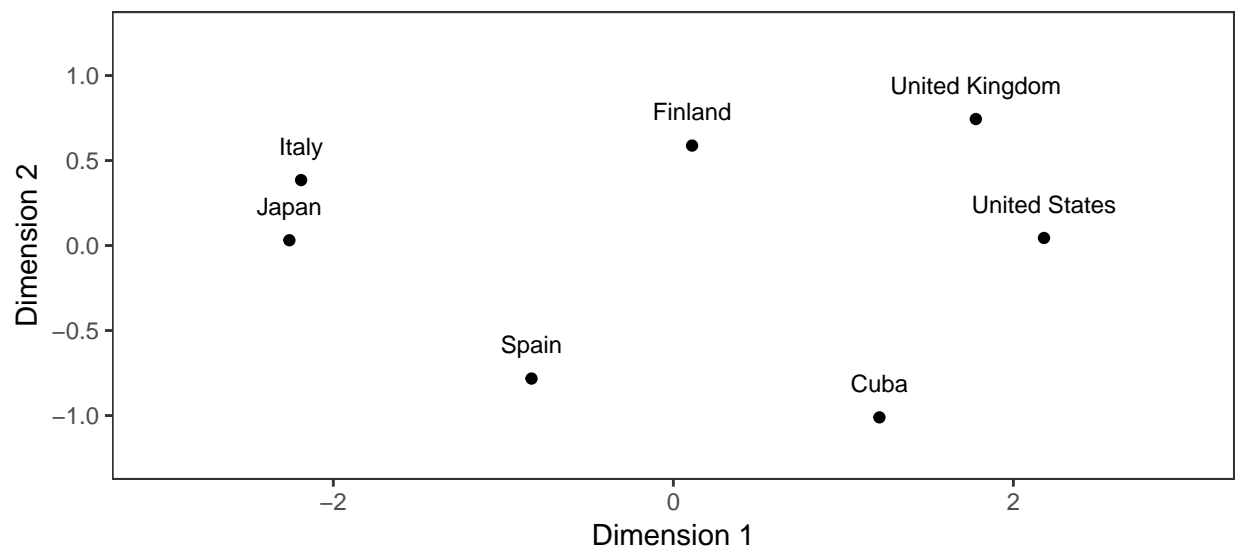
##	[,1]	[,2]
## Japan	-2.26	0.031
## Italy	-2.19	0.385
## Spain	-0.83	-0.782
## United Kingdom	1.78	0.744
## Finland	0.11	0.588
## Cuba	1.21	-1.011
## United States	2.18	0.045

Figure 14.1

```
country7 <- cbind(country7, cscal7)
names(country7) <- c("BirthR", "DeathR", "Dim1", "Dim2")
countries <- trimws(rownames(country7))

library(ggplot2)

p1 <- ggplot(country7, aes(x = Dim1, y = Dim2))
p2 <- p1 + geom_point() + geom_text(aes(label = countries),
                                   position = position_nudge(y = +0.20), size=3)
p3 <- p2 + scale_x_continuous(name = "Dimension 1", limits = c(-3, 3))
p4 <- p3 + scale_y_continuous(name = "Dimension 2", limits = c(-1.25, 1.25))
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p7 <- p6 + coord_fixed(ratio = 1)
p7
```



**Table 14.3: Dissimilarity Data for All Pairs of 10 Colas for 2 Subjects**

These data give an example of (symmetric) dissimilarity (proximity) matrices. There are no analyses involved here, but see some of the further examples (like the classical music composers) for possible ways to visualize and analyze these types of data sets.

Subject 1:

	Cola Number									
	1	2	3	4	5	6	7	8	9	10
1	0									
2	16	0								
3	81	47	0							
4	56	32	71	0						
5	87	68	44	71	0					
6	60	35	21	98	34	0				
7	84	94	98	57	99	99	0			
8	50	87	79	73	19	92	45	0		
9	99	25	53	98	52	17	99	84	0	
10	16	92	90	83	79	44	24	18	98	0

Subject 2:

	Cola Number									
	1	2	3	4	5	6	7	8	9	10
1	0									
2	20	0								
3	75	35	0							
4	60	31	80	0						
5	80	70	37	70	0					
6	55	40	20	89	30	0				
7	80	90	90	55	87	88	0			
8	45	80	77	75	25	86	40	0		
9	87	35	50	88	60	10	98	83	0	
10	12	90	96	89	75	40	27	14	90	0

Table 14.5: Road Distances of 15 Selected Places in Finland

```
fitowns <- c("Helsinki", "Joensuu", "Jyväskylä", "Kilpisjärvi", "Kokkola", "Kotka",
            "Kuopio", "Kuusamo", "Lappeenranta", "Nuorgam", "Oulu", "Rovaniemi",
            "Tampere", "Turku", "Vaasa")
ftwns <- abbreviate(fitowns, 3, strict = TRUE)
```

```
## Warning in abbreviate(fitowns, 3, strict = TRUE): abbreviate used with non-
## ASCII chars
```

```
fidist <- matrix(
  c( 0, 438, 272, 1202, 491, 134, 383, 802, 223, 1328, 612, 832, 174, 166, 419,
    438, 0, 245, 979, 429, 343, 136, 459, 236, 1045, 393, 550, 394, 549, 494,
    272, 245, 0, 931, 241, 244, 144, 551, 219, 1056, 339, 561, 151, 304, 282,
    1202, 979, 931, 0, 789, 1177, 878, 620, 1144, 649, 592, 428, 1079, 1226, 910,
    491, 429, 241, 789, 0, 488, 315, 411, 460, 916, 196, 419, 324, 436, 121,
    134, 343, 244, 1177, 488, 0, 316, 735, 108, 1303, 585, 806, 243, 295, 484,
    383, 136, 144, 878, 315, 316, 0, 417, 264, 1006, 286, 509, 293, 448, 379,
    802, 459, 551, 620, 411, 735, 417, 0, 682, 617, 212, 191, 702, 848, 533,
    223, 236, 219, 1144, 460, 108, 264, 682, 0, 1271, 551, 774, 275, 364, 501,
    1328, 1045, 1056, 649, 916, 1303, 1006, 617, 1271, 0, 719, 495, 1207, 1353, 1035,
    612, 393, 339, 592, 196, 585, 286, 212, 551, 719, 0, 222, 491, 633, 318,
    832, 550, 561, 428, 419, 806, 509, 191, 774, 495, 222, 0, 712, 856, 541,
    174, 394, 151, 1079, 324, 243, 293, 702, 275, 1207, 491, 712, 0, 153, 244,
    166, 549, 304, 1226, 436, 295, 448, 848, 364, 1353, 633, 856, 153, 0, 331,
    419, 494, 282, 910, 121, 484, 379, 533, 501, 1035, 318, 541, 244, 331, 0
  ), nrow = 15, ncol = 15, byrow = TRUE,
  dimnames = list(fitowns, ftwns))
```

```
fidist
```

```
##           Hls  Jns  Jyv  Klp  Kkk  Ktk  Kup  Ksm  Lpp  Nrg  Oul  Rvn  Tmp
## Helsinki      0  438  272 1202  491  134  383  802  223 1328  612  832  174
## Joensuu      438    0  245  979  429  343  136  459  236 1045  393  550  394
## Jyväskylä     272  245    0  931  241  244  144  551  219 1056  339  561  151
## Kilpisjärvi 1202  979  931    0  789 1177  878  620 1144  649  592  428 1079
## Kokkola      491  429  241  789    0  488  315  411  460  916  196  419  324
## Kotka        134  343  244 1177  488    0  316  735  108 1303  585  806  243
## Kuopio        383  136  144  878  315  316    0  417  264 1006  286  509  293
## Kuusamo       802  459  551  620  411  735  417    0  682  617  212  191  702
## Lappeenranta 223  236  219 1144  460  108  264  682    0 1271  551  774  275
## Nuorgam      1328 1045 1056  649  916 1303 1006  617 1271    0  719  495 1207
## Oulu          612  393  339  592  196  585  286  212  551  719    0  222  491
## Rovaniemi     832  550  561  428  419  806  509  191  774  495  222    0  712
## Tampere       174  394  151 1079  324  243  293  702  275 1207  491  712    0
## Turku         166  549  304 1226  436  295  448  848  364 1353  633  856  153
## Vaasa         419  494  282  910  121  484  379  533  501 1035  318  541  244
##
##           Trk  Vas
## Helsinki     166  419
## Joensuu      549  494
## Jyväskylä     304  282
## Kilpisjärvi 1226  910
## Kokkola       436  121
## Kotka         295  484
```

## Kuopio	448	379
## Kuusamo	848	533
## Lappeenranta	364	501
## Nuorgam	1353	1035
## Oulu	633	318
## Rovaniemi	856	541
## Tampere	153	244
## Turku	0	331
## Vaasa	331	0



Table 14.6

```
n <- dim(fidist)[1]

fiscal <- cmdscale(d = fidist, k = n - 1, eig = TRUE, list. = TRUE)

## Warning in cmdscale(d = fidist, k = n - 1, eig = TRUE, list. = TRUE): only
## 8 of the first 14 eigenvalues are > 0

fiscal$eig

## [1] 2.4e+06 2.9e+05 2.0e+05 5.7e+04 1.6e+04 1.5e+04 7.8e+03
## [8] 7.2e+02 -1.1e-10 -9.9e+02 -5.9e+03 -1.3e+04 -2.1e+04 -3.7e+04
## [15] -5.6e+04

as.matrix(format(fiscal$eig, scientific = FALSE, justify = "right", nsmall = 0L, digits = 0))

##      [,1]
## [1,] "2395788"
## [2,] " 285542"
## [3,] " 198140"
## [4,] "  56660"
## [5,] "  16092"
## [6,] "  15119"
## [7,] "   7806"
## [8,] "    724"
## [9,] "    -0"
## [10,] "   -995"
## [11,] "  -5886"
## [12,] " -13245"
## [13,] " -21336"
## [14,] " -37055"
## [15,] " -56468"

fiscal$points[, 1:2]

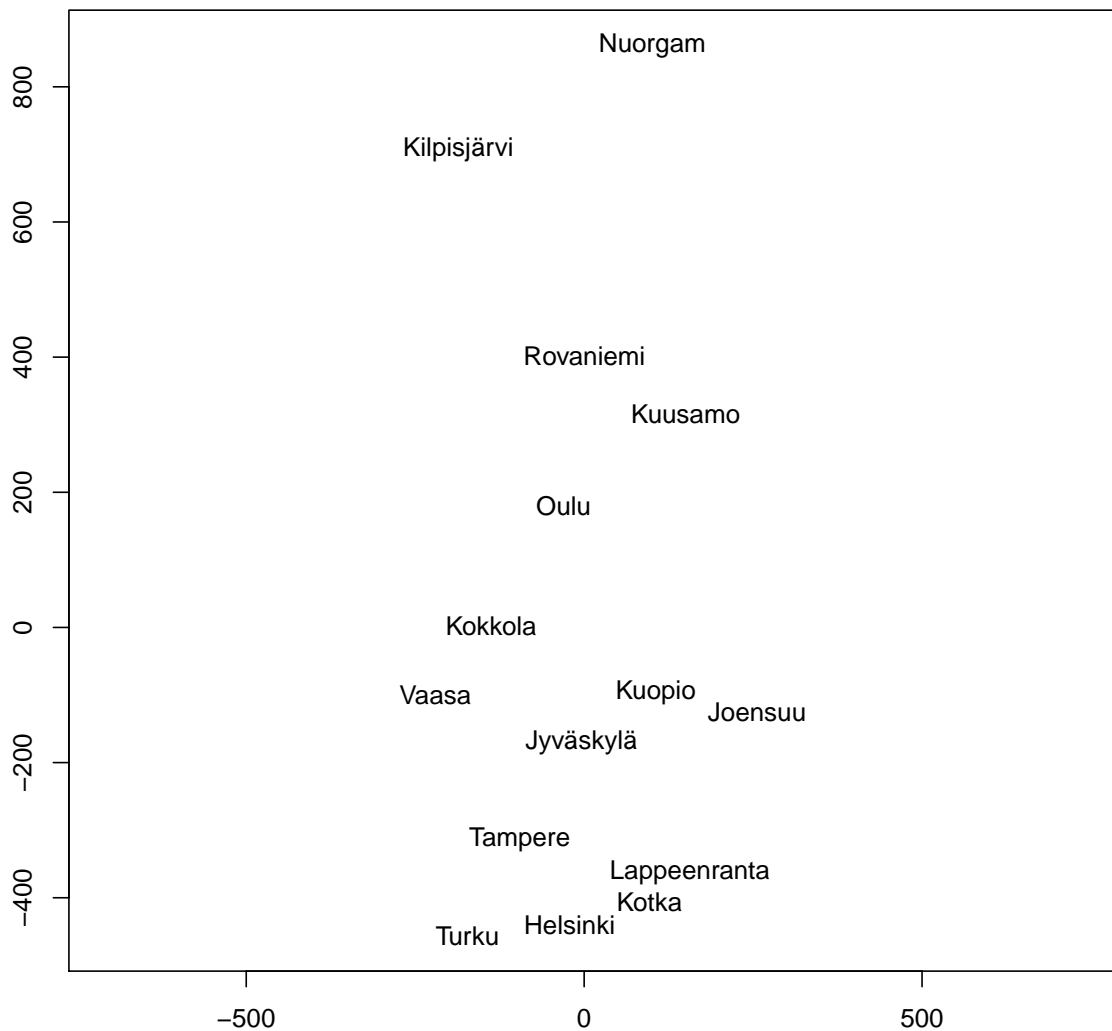
##      [,1]      [,2]
## Helsinki -440.7 -21.46
## Joensuu -124.5 255.86
## Jyväskylä -169.6 -3.75
## Kilpisjärvi 707.1 -185.96
## Kokkola 2.6 -137.40
## Kotka -405.8 96.89
## Kuopio -95.7 106.04
## Kuusamo 316.6 150.25
## Lappeenranta -362.9 156.52
## Nuorgam 860.8 100.99
## Oulu 180.5 -30.44
## Rovaniemi 402.0 0.45
## Tampere -314.1 -95.33
## Turku -455.9 -172.48
## Vaasa -100.4 -220.18
```

Figure 14.2

```
# draft plot:

y <- fiscal$points[, 1] # exchange x and y to show the map
x <- fiscal$points[, 2] # so that Helsinki is in the South

plot(x, y, type = "n", xlab = "", ylab = "", asp = 1, axes = TRUE)
text(x, y, rownames(fiscal$points), cex = 1.0)
```



```
#install.packages("vegan") # by Jari Oksanen, Finland
library(vegan)
```

```
## Loading required package: permute
```

```

## Loading required package: lattice
## This is vegan 2.5-2
finland <- as.data.frame(scores(fiscal$points[, 1:2])) # adds names Dim1, Dim2

library(ggplot2)

p1 <- ggplot(finland, aes(x = Dim2, y = Dim1)) # OBS! rotation
p2 <- p1 + geom_point() + geom_text(aes(label = fitowns),
                                   position = position_nudge(y = +40), size=4)
p3 <- p2 + scale_x_continuous(name = "Dimension 2", limits = c(-500, +500))
p4 <- p3 + scale_y_continuous(name = "Dimension 1", limits = c(-500, +1000))
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p7 <- p6 + coord_fixed(ratio = 1)
p7

```

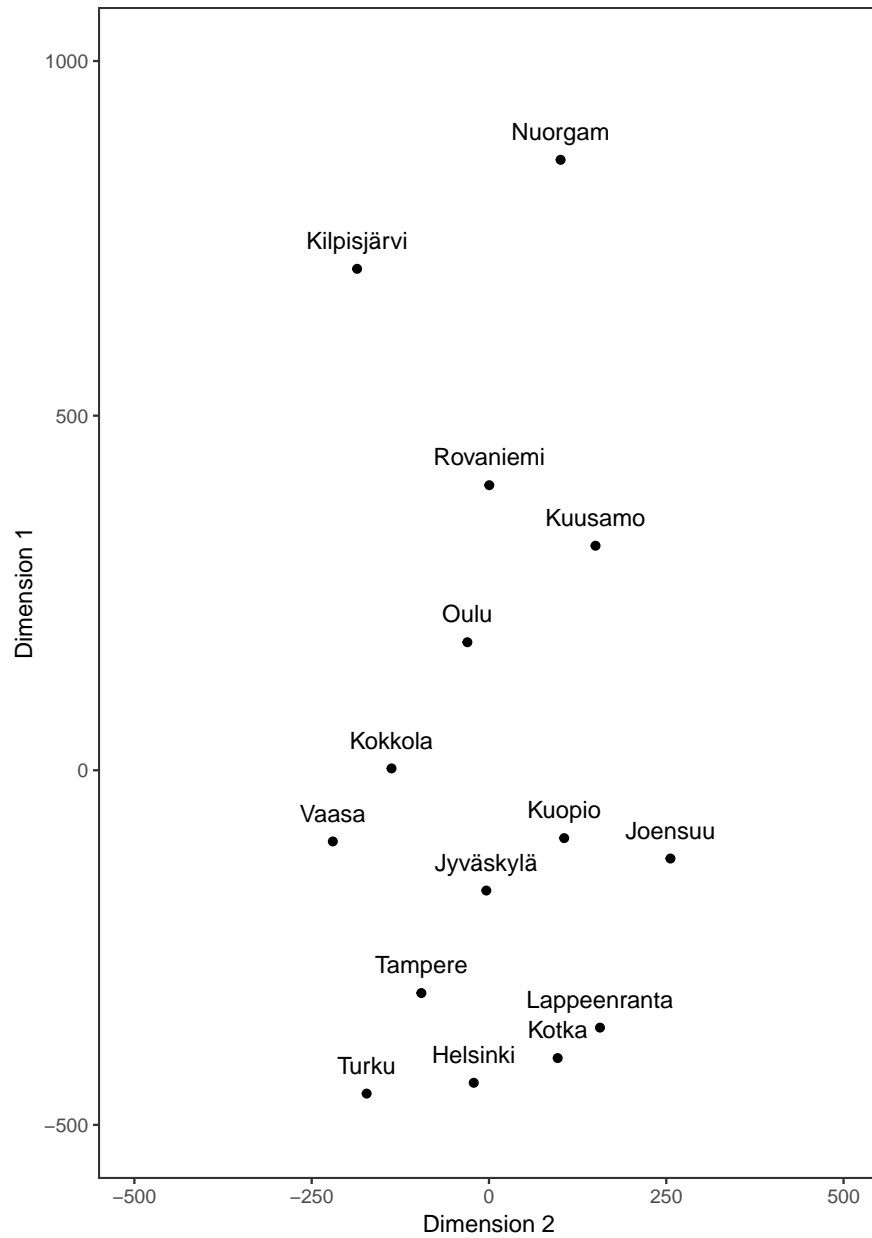
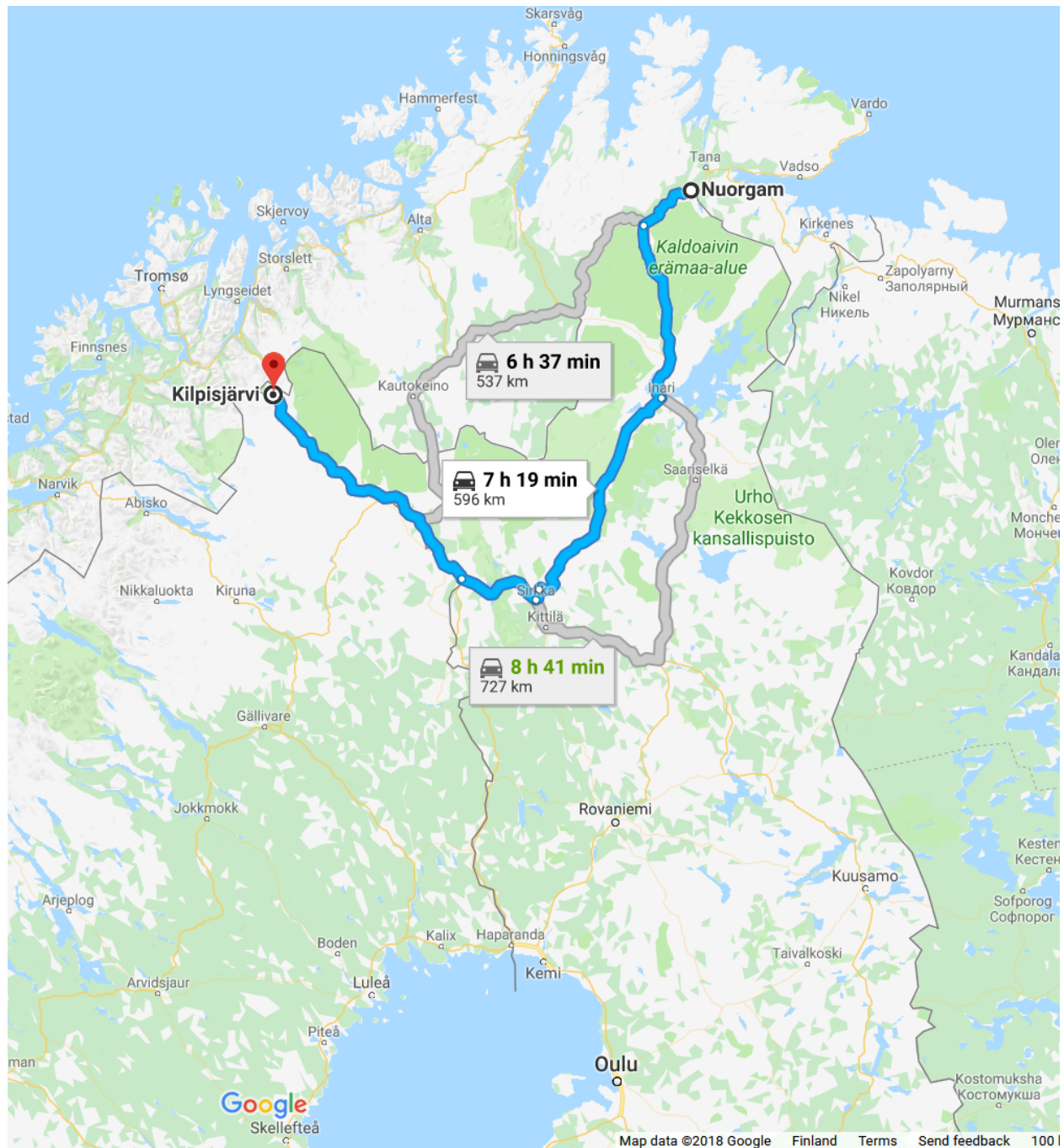


Figure 14.3

For an interactive map, see <https://goo.gl/maps/5Yp6nG31HeQ2>



For an interactive map, see <https://goo.gl/maps/5Yp6nG31HeQ2>

**Table 14.7: Proximity Matrix of Ten Remarkable Classical Music Composers Selected and Compared by Olli Mustonen**

```
composers <- c("Bach", "Haydn", "Mozart", "Beethoven", "Schubert", "Brahms",
              "Sibelius", "Debussy", "Bartok", "Sostakovits")
OMD <- matrix(
c( 0,  50,  30,  20,  40,  40,  40,  50,  30,  30,
   50,  0,  10,  15,  30,  70,  90,  50,  80,  40,
   30,  10,  0,  20,  25,  40,  70,  50,  80,  50,
   20,  15,  20,  0,  10,  20,  25,  80,  60,  40,
   40,  30,  25,  10,  0,  15,  60,  50,  70,  60,
   40,  70,  40,  20,  15,  0,  20,  70,  70,  70,
   40,  90,  70,  25,  60,  20,  0,  35,  35,  20,
   50,  50,  50,  80,  50,  70,  35,  0,  15,  40,
   30,  80,  80,  60,  70,  70,  35,  15,  0,  20,
   30,  40,  50,  40,  60,  70,  20,  40,  20,  0
), nrow = 10, ncol = 10, byrow = TRUE, dimnames = list(composers, composers))

n <- dim(OMD)[1]
OMDS <- cmdscale(d = OMD, k = n-1, eig = TRUE, list. = TRUE)

## Warning in cmdscale(d = OMD, k = n - 1, eig = TRUE, list. = TRUE): only 5
## of the first 9 eigenvalues are > 0

as.matrix(format(OMDS$eig, scientific = FALSE, justify = "right", nsmall = 0L, digits = 0))

##      [,1]
## [1,] " 7459"
## [2,] " 4830"
## [3,] " 2288"
## [4,] "  752"
## [5,] "  514"
## [6,] "    0"
## [7,] " -661"
## [8,] " -906"
## [9,] " -937"
## [10,] "-2912"

pk1 <- cumsum(abs(OMDS$eig))/sum(abs(OMDS$eig))
pk2 <- cumsum(OMDS$eig^2)/sum(OMDS$eig^2)
pk1

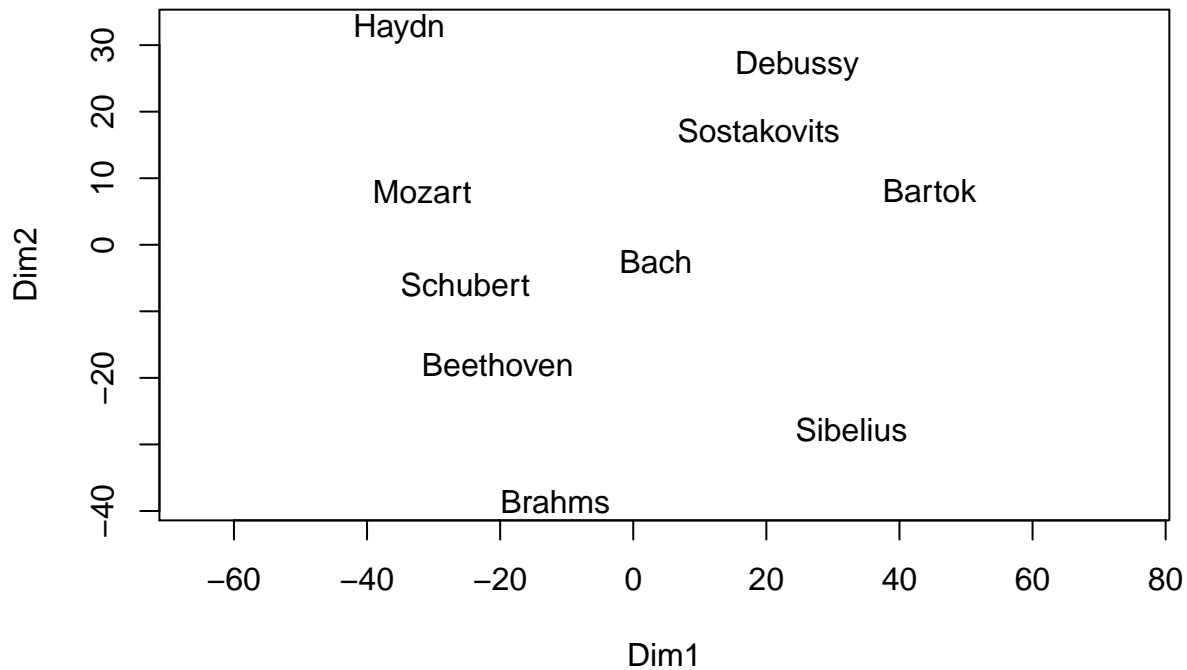
## [1] 0.35 0.58 0.69 0.72 0.75 0.75 0.78 0.82 0.86 1.00

pk2

## [1] 0.58 0.83 0.88 0.89 0.89 0.89 0.89 0.90 0.91 1.00
```

Figure 14.4

```
OwMDS <- wcmdscale(d = OMD, k = n-1, eig = TRUE)
plot(OwMDS, cex = 1.0)
```



```
Owscal <- as.data.frame(scores(OwMDS$points[, 1:2]))

library(ggplot2)

p1 <- ggplot(Owscal, aes(x = Dim1, y = Dim2))
p2 <- p1 + geom_point() + geom_text(aes(label = composers),
                                     position = position_nudge(y = +4), size=4)
p3 <- p2 + scale_x_continuous(name = "Dimension 1", limits = c(-40, +50))
p4 <- p3 + scale_y_continuous(name = "Dimension 2", limits = c(-50, +40))
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p7 <- p6 + coord_fixed(ratio = 1)
p7
```

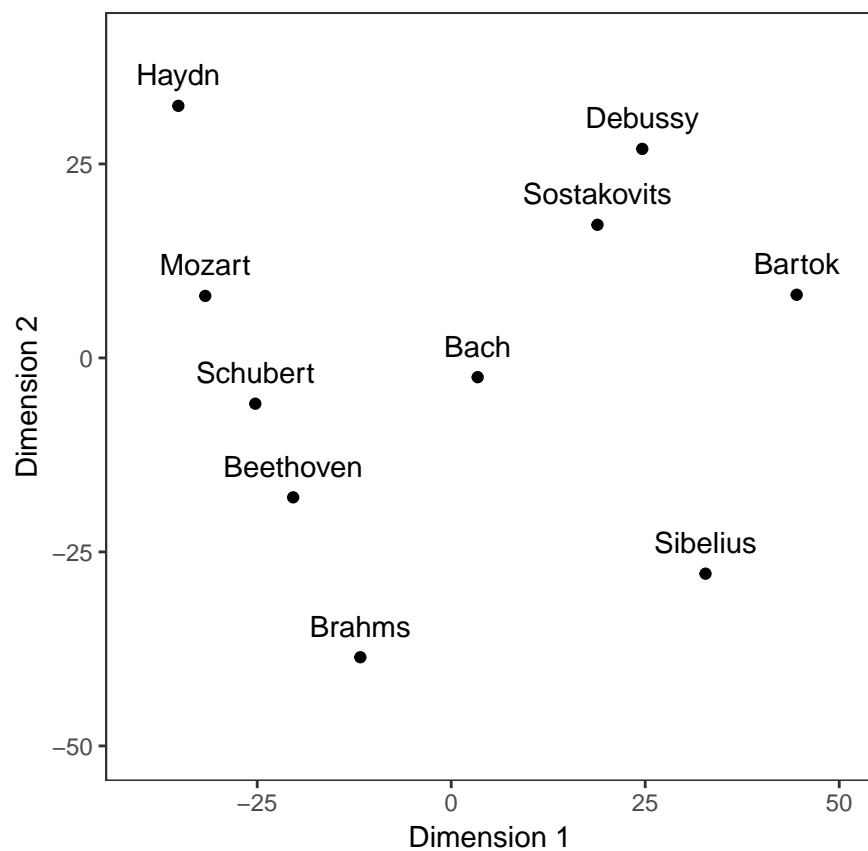
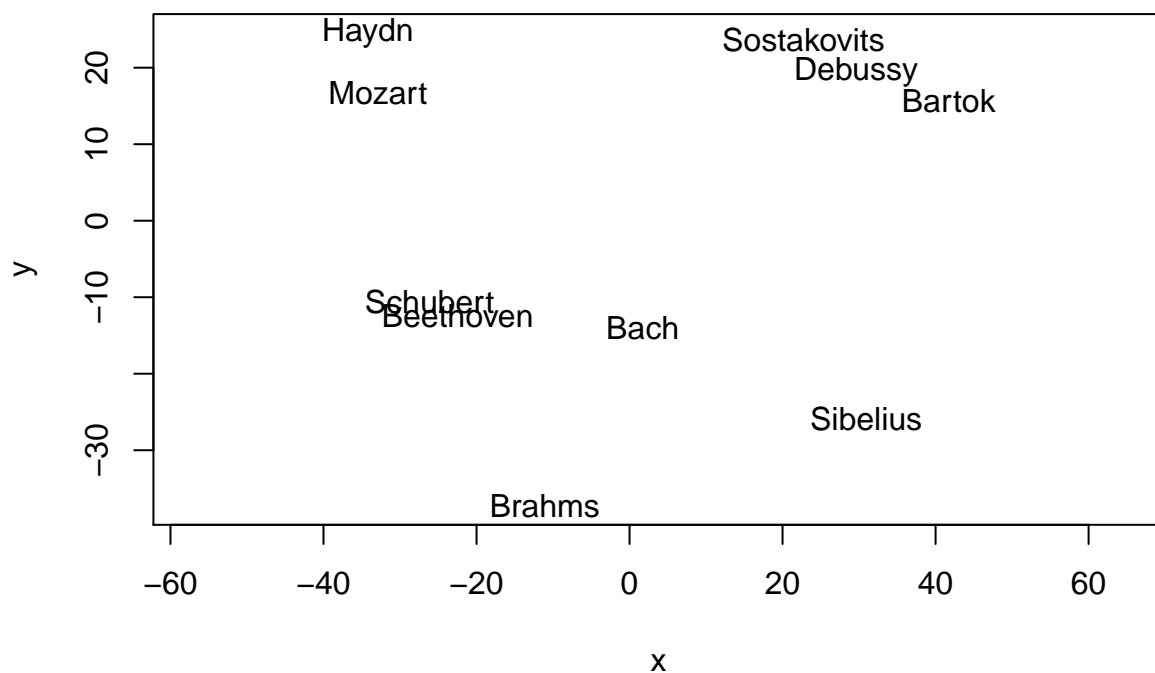




Figure 14.5

```
library(vegan)
OmonoMDS <- monoMDS(dist = OMD, y = cmdscale(OMD, 3), k = 3,
                    model = "global", scaling = FALSE, pc = FALSE)

x <- OmonoMDS$points[, 1]
y <- OmonoMDS$points[, 2]
plot(x, y, type = "n", asp = 1, axes = TRUE)
text(x, y, composers, cex = 1.0)
```



```
Omono <- as.data.frame(scores(OmonoMDS$points[, 1:3])) # adds names MDS1,2,3

p1 <- ggplot(Omono, aes(x = MDS1, y = MDS2))
p2 <- p1 + geom_point() + geom_text(aes(label = composers),
                                   position = position_nudge(y = +4), size=4)
p3 <- p2 + scale_x_continuous(name = "Dimension 1", limits = c(-40, +50))
p4 <- p3 + scale_y_continuous(name = "Dimension 2", limits = c(-50, +40))
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p7 <- p6 + coord_fixed(ratio = 1)
p7
```

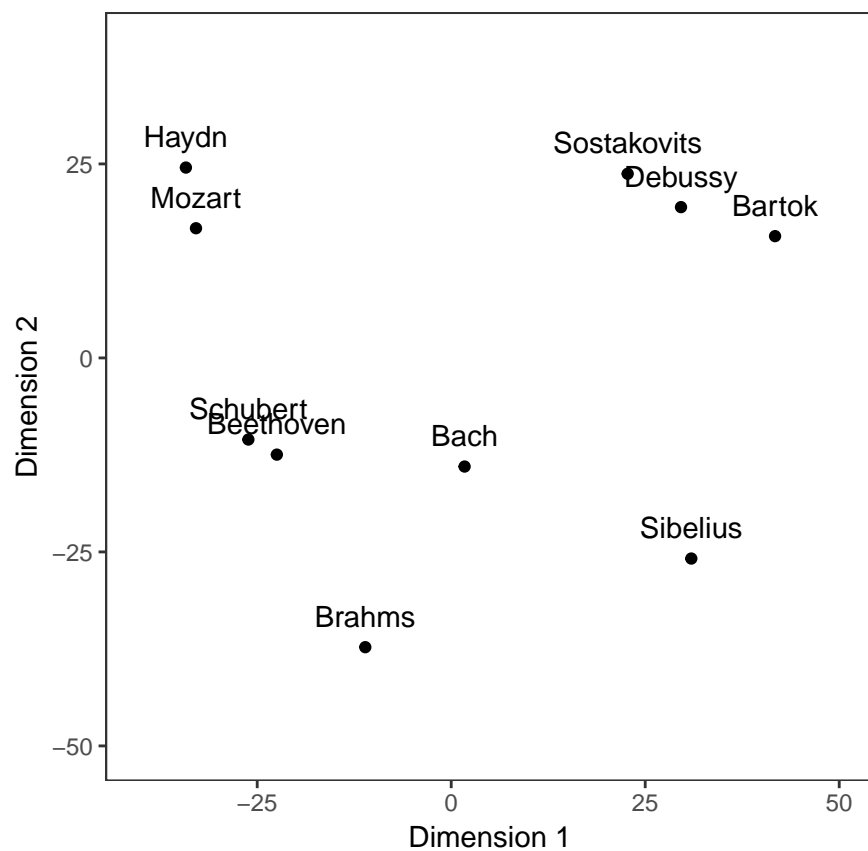


Figure 14.6

```
p1 <- ggplot(Omono, aes(x = MDS1, y = MDS3))
p2 <- p1 + geom_point() + geom_text(aes(label = composers),
                                   position = position_nudge(y = +4.1), size=4)
p3 <- p2 + scale_x_continuous(name = "Dimension 1", limits = c(-40, +50))
p4 <- p3 + scale_y_continuous(name = "Dimension 3", limits = c(-50, +40))
p5 <- p4 + theme_bw()
p6 <- p5 + theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
p7 <- p6 + coord_fixed(ratio = 1)
p7
```

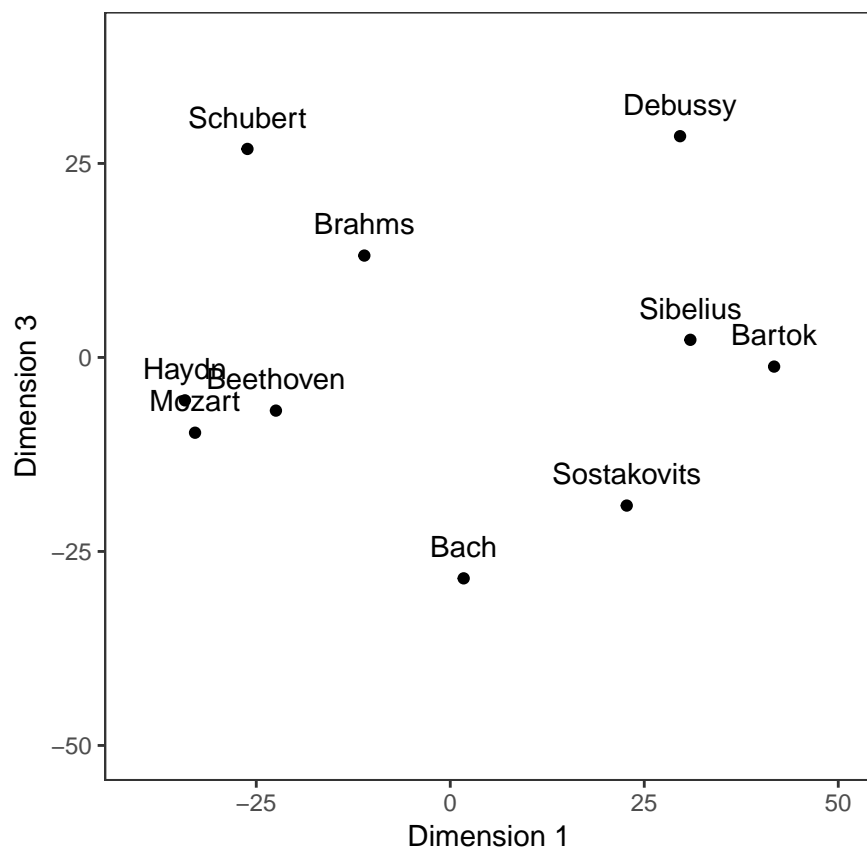
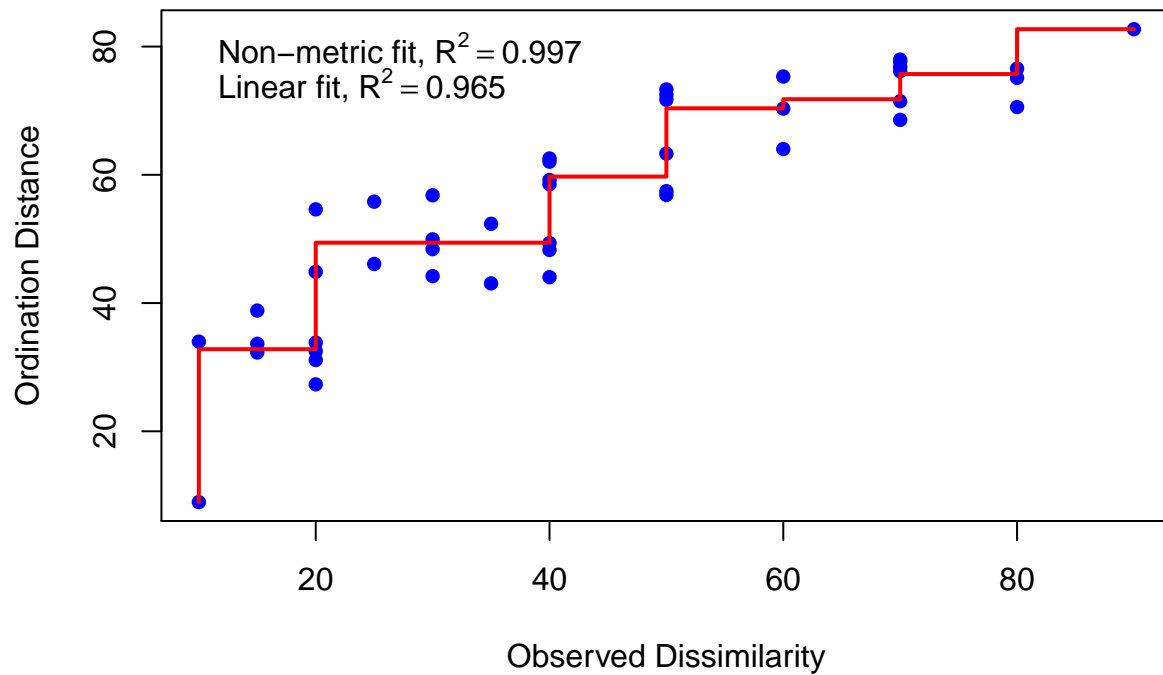
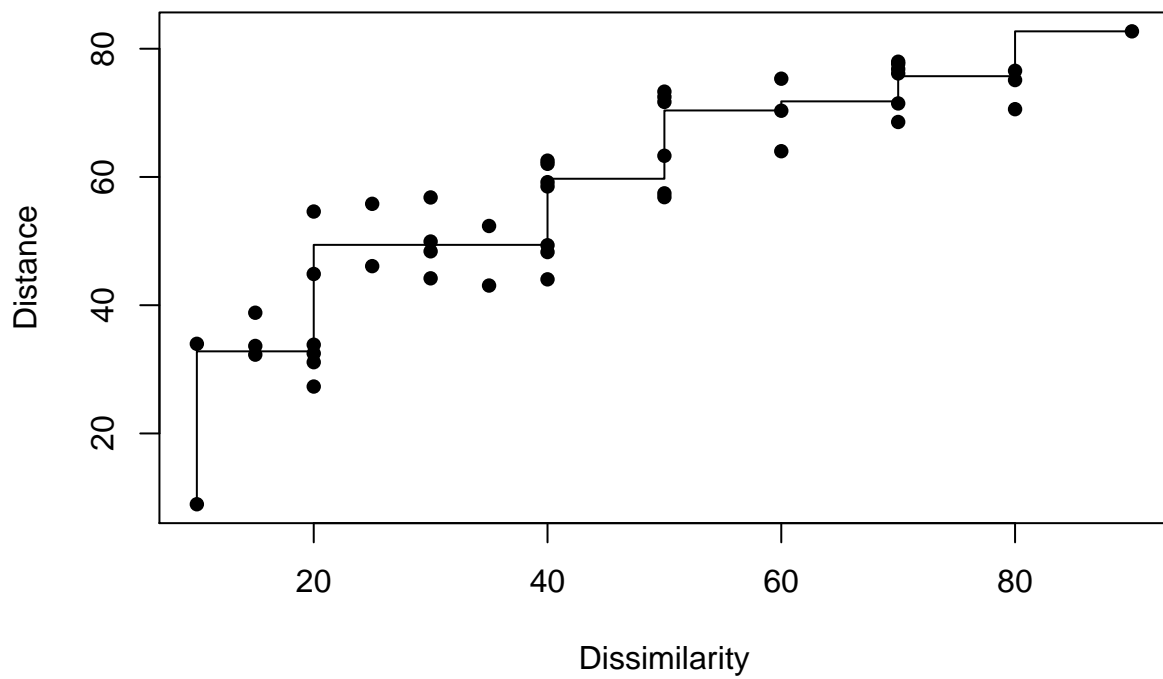


Figure 14.7

```
# instead of Shepard function, use stressplot from vegan package:  
strpl <- stressplot(OmonoMDS, pch = 16)
```



```
plot(strpl, pch = 16, xlab = "Dissimilarity", ylab = "Distance")  
lines(sort(strpl$x), sort(strpl$yf), type = "S")
```



**Table 14.8: The Influence of Age of Relationship Status with Boyfriends**

```
Girls <- as.table(rbind(
  c(21, 21, 14, 13, 8),
  c(8, 9, 6, 8, 2),
  c(2, 3, 4, 10, 10)))

dimnames(Girls) <- list(relship = c("No boyfriend",
  "Boyfriend: No sex",
  "Boyfriend: Sex"),
  agegroup = c("< 16", "16-17", "17-18", "18-19", "19-20"))
```

Girls

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend      21   21   14   13    8
## Boyfriend: No sex   8    9    6    8    2
## Boyfriend: Sex      2    3    4   10   10
```

*# observed and expected frequencies, chi-square test, percentages etc.*

```
Xsq <- chisq.test(Girls) # Prints test summary
```

## Warning in chisq.test(Girls): Chi-squared approximation may be incorrect

```
Xsq$observed # observed counts (same as Girls)
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend      21   21   14   13    8
## Boyfriend: No sex   8    9    6    8    2
## Boyfriend: Sex      2    3    4   10   10
```

```
Xsq$expected # expected counts under the null
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend      17.2 18.3 13.3 17.2 11.1
## Boyfriend: No sex   7.4  7.8  5.7  7.4  4.7
## Boyfriend: Sex      6.5  6.9  5.0  6.5  4.2
```

Xsq

```
##
## Pearson's Chi-squared test
##
## data: Girls
## X-squared = 20, df = 8, p-value = 0.008
```

```
100*prop.table(Girls, margin = 1)
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend      27.3 27.3 18.2 16.9 10.4
## Boyfriend: No sex 24.2 27.3 18.2 24.2  6.1
## Boyfriend: Sex     6.9 10.3 13.8 34.5 34.5
```

```
100*prop.table(Girls, margin = 2)
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend  67.7 63.6 58.3 41.9 40.0
## Boyfriend: No sex 25.8 27.3 25.0 25.8 10.0
## Boyfriend: Sex   6.5  9.1 16.7 32.3 50.0
```

```
100*prop.table(Girls)
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20
## No boyfriend  15.1 15.1 10.1  9.4  5.8
## Boyfriend: No sex 5.8  6.5  4.3  5.8  1.4
## Boyfriend: Sex   1.4  2.2  2.9  7.2  7.2
```

```
addmargins(100*prop.table(Girls, margin = 1), margin = c(1,2), FUN = list(Sum = mean, Sum = sum) )
```

```
## Margins computed over dimensions
## in the following order:
## 1: relship
## 2: agegroup
```

```
##               agegroup
## relship      < 16 16-17 17-18 18-19 19-20    Sum
## No boyfriend  27.3 27.3 18.2 16.9 10.4 100.0
## Boyfriend: No sex 24.2 27.3 18.2 24.2  6.1 100.0
## Boyfriend: Sex   6.9 10.3 13.8 34.5 34.5 100.0
## Sum             19.5 21.6 16.7 25.2 17.0 100.0
```

Figure 14.8

```
#install.packages("ca")
library(ca)
caGirls <- ca(Girls)
caGirls
```

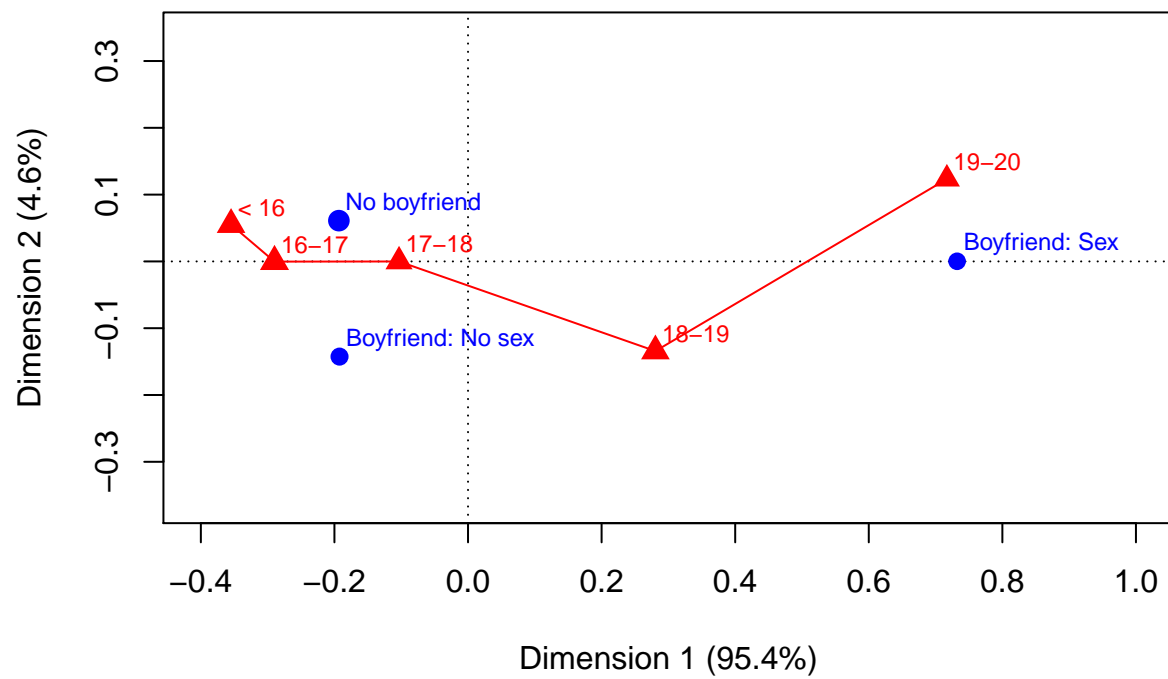
```
##
## Principal inertias (eigenvalues):
##      1      2
## Value  0.141348 0.006884
## Percentage 95.36%  4.64%
##
##
## Rows:
##      No boyfriend Boyfriend: No sex Boyfriend: Sex
## Mass      0.554      0.237      0.209
## ChiDist    0.203      0.239      0.732
## Inertia    0.023      0.014      0.112
## Dim. 1     -0.514     -0.512      1.948
## Dim. 2      0.735     -1.718      0.002
##
##
## Columns:
##      < 16   16-17   17-18   18-19 19-20
## Mass    0.223 0.2374 0.1727 0.223 0.144
## ChiDist 0.359 0.2897 0.1033 0.311 0.727
## Inertia 0.029 0.0199 0.0018 0.022 0.076
## Dim. 1  -0.944 -0.7706 -0.2747 0.746 1.907
## Dim. 2   0.663 -0.0036 -0.0013 -1.618 1.487
```



```
summary(caGirls)
```

```
##
## Principal inertias (eigenvalues):
##
## dim    value      %   cum%   scree plot
## 1      0.141348  95.4  95.4   *****
## 2      0.006884   4.6 100.0   *
##
## -----
## Total: 0.148232 100.0
##
##
## Rows:
##   name  mass  qlt  inr   k=1  cor  ctr   k=2  cor  ctr
## 1 | Nbyf |  554 1000 154 | -193 909 146 |   61  91 300 |
## 2 | ByfN |  237 1000  92 | -192 646  62 | -143 354 700 |
## 3 | ByfS |  209 1000 755 |  732 1000 791 |    0   0   0 |
##
## Columns:
##   name  mass  qlt  inr   k=1  cor  ctr   k=2  cor  ctr
## 1 |  16 |  223 1000 194 | -355 977 199 |   55  23  98 |
## 2 | 1617 |  237 1000 134 | -290 1000 141 |    0   0   0 |
## 3 | 1718 |  173 1000  12 | -103 1000  13 |    0   0   0 |
## 4 | 1819 |  223 1000 146 |  281  814 124 | -134 186 584 |
## 5 | 1920 |  144 1000 514 |  717  971 523 |  123  29 318 |
```

```
plot(caGirls, lines=c(FALSE, TRUE), mass=c(TRUE,TRUE), xlim = c(-0.4,1))
```



**Table 14.9: Types of Work Activities and Main Advantages of Job from 6933 Survey Respondents**

**\*\*Source of data:\*\*** Lebart, L., Morineau, A. and Warwick, K. (1984). *Multivariate Descriptive Statistical Analysis: Correspondence Analysis and Related Techniques for Large Matrices*. John Wiley & Sons, Chichester, U.K.

```
jobs <- read.csv(file = "data/jobs.csv", row.names = 1)
str(jobs)
```

```
## 'data.frame':    26 obs. of  23 variables:
## $ WORKTYPE: Factor w/ 26 levels "Administrative services  ",...: 8 7 6 21 3 26 2 24 16 12 ...
## $ VARIETY  : int   4 1 1 5 2 2 2 3 3 0 ...
## $ FREEDOM  : int  189 13 9 5 7 5 3 18 7 18 ...
## $ HUMAN    : int   0 3 1 2 1 0 1 0 3 1 ...
## $ SCHEDULE: int   3 10 0 9 4 4 8 6 6 12 ...
## $ SALARIES: int   2 17 4 18 15 1 16 16 6 31 ...
## $ SECURITY: int   2 12 13 5 5 0 17 5 0 7 ...
## $ FAMILY   : int   9 4 0 3 2 3 1 4 0 0 ...
## $ INTEREST: int   3 1 2 2 1 0 8 4 2 8 ...
## $ NEARHOME: int  12 8 2 6 6 2 7 13 6 19 ...
## $ ATMOSPHE: int   2 3 0 5 1 1 2 4 3 11 ...
## $ SOC_ADV  : int   1 5 2 5 2 1 4 2 3 3 ...
## $ OWN_BOSS: int   4 1 1 0 2 1 3 3 0 2 ...
## $ LIKE_IT  : int  11 9 4 2 3 1 6 6 2 10 ...
## $ OTHER    : int  15 5 3 3 0 0 1 2 1 4 ...
## $ NONE     : int  12 11 6 22 5 3 24 26 8 26 ...
## $ OUTDOORS: int   8 0 1 0 0 0 0 0 0 0 ...
## $ NO_ANS   : int   1 0 0 0 1 2 1 2 0 6 ...
## $ NO_DIPL  : int  93 32 9 29 13 10 25 28 11 51 ...
## $ HIGH_SCH: int  18 15 8 4 13 5 4 4 11 26 ...
## $ FIRM10   : int  187 12 6 4 3 6 7 24 6 32 ...
## $ FIRM500  : int   2 40 17 30 14 11 19 38 16 40 ...
## $ FIRM1000: int   0 11 6 27 25 0 61 6 9 30 ...
```

```
row.names(jobs) <- jobs$WORKTYPE
jobs <- jobs[, 2:23]
# for simplicity, withOUT supplementary variables:
jobs2 <- jobs[, 1:17]
head(jobs2[, c(1:3, 16:17)])
```

	VARIETY	FREEDOM	HUMAN	OUTDOORS	NO_ANS
## Farming-fishing	4	189	0	8	1
## Farm-food industry	1	13	3	0	0
## Energy-mines	1	9	1	1	0
## Steel	5	5	2	0	0
## Chemical-glass-oil	2	7	1	0	1
## Wood-paper	2	5	0	0	2

```
chisq.test(jobs2)
```

```
## Warning in chisq.test(jobs2): Chi-squared approximation may be incorrect
##
## Pearson's Chi-squared test
##
```

```
## data: jobs2
## X-squared = 2000, df = 400, p-value <2e-16
```

Figure 14.9

```
lebart <- ca(jobs2)
plot(lebart, xlim = c(-1.1, 1.1), ylim = c(-1.1, 0.5), mass=c(TRUE,TRUE))
```

