

Multidimensional Scaling

Blake Pappas

December 17, 2023

Classical Multidimensional Scaling

Check $B = -\frac{1}{2}CD^2C$

```
n <- 100
library(MASS)
sigma <- c(1, 0.8, 0.64, 0.8, 1, 0.8, 0.64, 0.8, 1)
Sigma <- matrix(sigma, 3, 3)
set.seed(123)
X <- mvrnorm(n, rep(0, 3), Sigma)
X <- scale(X, center = T, scale = F)
B <- X %*% t(X)
C <- diag(n) - (outer(rep(1, n), rep(1, n))) / n
library(fields)

## Loading required package: spam

## Spam version 2.9-1 (2022-08-07) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.

##
## Attaching package: 'spam'

## The following objects are masked from 'package:base':
##
##      backsolve, forwardsolve

## Loading required package: viridis

## Loading required package: viridisLite

##
## Try help(fields) to get started.
```

```
D <- rdist(X)
tmp <- (-1 / 2) * C %*% D^(2) %*% C
```

Distances Between US Cities

```
loc <- cmdscale(UScitiesD)
x <- loc[, 1]; y <- loc[, 2]
plot(x, y, type = "n", xlab = "", ylab = "", asp = 1,
      axes = FALSE, main = "cmdscale(UScitiesD)")
text(x, y, rownames(loc), cex = 0.8)
```

cmdscale(UScitiesD)



```
# Flip Axes
x1 <- -loc[, 1]; y1 <- -loc[, 2]
plot(x1, y1, type = "n", xlab = "", ylab = "", asp = 1,
      axes = FALSE, main = "cmdscale(UScitiesD)")
text(x1, y1, rownames(loc), cex = 0.8)
```

cmdscale(UScitiesD)



Air Pollution in US Cities

```
library(HSAUR3)
```

```
## Loading required package: tools
```

```
data(USairpollution)
dat <- USairpollution
head(dat)
```

```
##           SO2 temp manu popul wind precip predays
## Albany      46 47.6   44   116  8.8  33.36      135
## Albuquerque  11 56.8   46   244  8.9   7.77       58
## Atlanta     24 61.5  368   497  9.1  48.34      115
## Baltimore   47 55.0  625   905  9.6  41.31      111
## Buffalo     11 47.1  391   463 12.4  36.11      166
## Charleston  31 55.2   35    71  6.5  40.75      148
```

```
summary(dat)
```

```
##          SO2              temp              manu              popul
## Min.      : 8.00    Min.      :43.50    Min.      : 35.0    Min.      : 71.0
## 1st Qu.: 13.00    1st Qu.:50.60    1st Qu.: 181.0    1st Qu.: 299.0
## Median : 26.00    Median :54.60    Median : 347.0    Median : 515.0
## Mean     : 30.05    Mean     :55.76    Mean     : 463.1    Mean     : 608.6
## 3rd Qu.: 35.00    3rd Qu.:59.30    3rd Qu.: 462.0    3rd Qu.: 717.0
## Max.     :110.00    Max.     :75.50    Max.     :3344.0    Max.     :3369.0
##          wind          precip          predays
## Min.      : 6.000    Min.      : 7.05    Min.      : 36.0
## 1st Qu.: 8.700    1st Qu.:30.96    1st Qu.:103.0
## Median : 9.300    Median :38.74    Median :115.0
## Mean     : 9.444    Mean     :36.77    Mean     :113.9
## 3rd Qu.:10.600    3rd Qu.:43.11    3rd Qu.:128.0
## Max.     :12.700    Max.     :59.80    Max.     :166.0
```

```
xs <- apply(dat, 2, function(x) (x - min(x)) / (diff(range(x))))
summary(xs)
```

```
##          SO2              temp              manu              popul
## Min.      :0.00000    Min.      :0.0000    Min.      :0.00000    Min.      :0.00000
## 1st Qu.:0.04902    1st Qu.:0.2219    1st Qu.:0.04412    1st Qu.:0.06913
## Median :0.17647    Median :0.3469    Median :0.09429    Median :0.13463
## Mean     :0.21616    Mean     :0.3832    Mean     :0.12937    Mean     :0.16301
## 3rd Qu.:0.26471    3rd Qu.:0.4938    3rd Qu.:0.12904    3rd Qu.:0.19588
## Max.     :1.00000    Max.     :1.0000    Max.     :1.00000    Max.     :1.00000
##          wind          precip          predays
## Min.      :0.0000    Min.      :0.0000    Min.      :0.0000
## 1st Qu.:0.4030    1st Qu.:0.4533    1st Qu.:0.5154
## Median :0.4925    Median :0.6008    Median :0.6077
## Mean     :0.5140    Mean     :0.5634    Mean     :0.5992
## 3rd Qu.:0.6866    3rd Qu.:0.6836    3rd Qu.:0.7077
## Max.     :1.0000    Max.     :1.0000    Max.     :1.0000
```

```
# Compute Distance Matrix
```

```
poldist <- dist(xs)
```

```
# Reduce to 2 Dimensions
```

```
(pol.mds <- cmdscale(poldist, k = 2, eig = TRUE))
```

```
## $points
##          [,1]      [,2]
## Albany      0.140558172 -0.046859954
## Albuquerque -0.364824787 -0.636602091
## Atlanta     -0.155922591  0.244511276
## Baltimore    0.153189990  0.067519907
## Buffalo      0.256244063  0.003022604
## Charleston  -0.128730958  0.215783429
## Chicago      1.197000315  0.009638168
## Cincinnati  -0.084166097  0.106828800
## Cleveland    0.531787447  0.056305378
```

```

## Columbus      0.025412911  0.033574934
## Dallas        -0.258008194 -0.062640448
## Denver        -0.110682033 -0.510378502
## Des Moines    -0.007603614 -0.244344703
## Detroit       0.341537781 -0.105917971
## Hartford      0.206766531  0.105259858
## Houston       -0.188167760  0.243707765
## Indianapolis   0.069589745  0.010565926
## Jacksonville  -0.349520267  0.412490203
## Kansas City   -0.106424371 -0.085118726
## Little Rock   -0.355970056  0.194004542
## Louisville    -0.046780470  0.144850917
## Memphis       -0.249259311  0.208737990
## Miami         -0.449823739  0.604996816
## Milwaukee     0.217298744 -0.249612250
## Minneapolis   0.326439578 -0.242858309
## Nashville     -0.215002650  0.211835269
## New Orleans   -0.410715158  0.438263300
## Norfolk       -0.066285208  0.149134571
## Omaha         -0.063335982 -0.241936316
## Philadelphia   0.521031706  0.081089446
## Phoenix       -0.695773353 -0.527859295
## Pittsburgh    0.314965899  0.074640031
## Providence    0.466505620  0.110503750
## Richmond      -0.191967563  0.140461889
## Salt Lake City -0.111111665 -0.461383196
## San Francisco -0.253430076 -0.401897024
## Seattle       0.170829143  0.147411289
## St. Louis     0.162208664 -0.016576959
## Washington    -0.031338057  0.041417952
## Wichita       -0.149744969 -0.268806546
## Wilmington    -0.056777379  0.046236280
##
## $eig
## [1] 4.456648e+00 2.819944e+00 2.256196e+00 1.651762e+00 6.199354e-01
## [6] 1.904906e-01 3.068220e-02 1.558353e-15 9.406328e-16 2.494225e-16
## [11] 1.736021e-16 1.471280e-16 1.356518e-16 8.017147e-17 7.511957e-17
## [16] 6.686099e-17 5.684599e-17 5.034791e-17 4.025565e-17 3.312471e-17
## [21] 2.974204e-17 1.555983e-17 1.132251e-17 3.668800e-18 -5.206488e-18
## [26] -8.948794e-18 -9.519928e-18 -1.506805e-17 -1.853275e-17 -2.314710e-17
## [31] -2.858271e-17 -3.093804e-17 -3.151435e-17 -3.396470e-17 -7.209856e-17
## [36] -7.714641e-17 -1.524915e-16 -2.390840e-16 -2.833661e-16 -3.238640e-16
## [41] -1.263609e-15
##
## $x
## NULL
##
## $ac
## [1] 0
##
## $GOF
## [1] 0.6050889 0.6050889

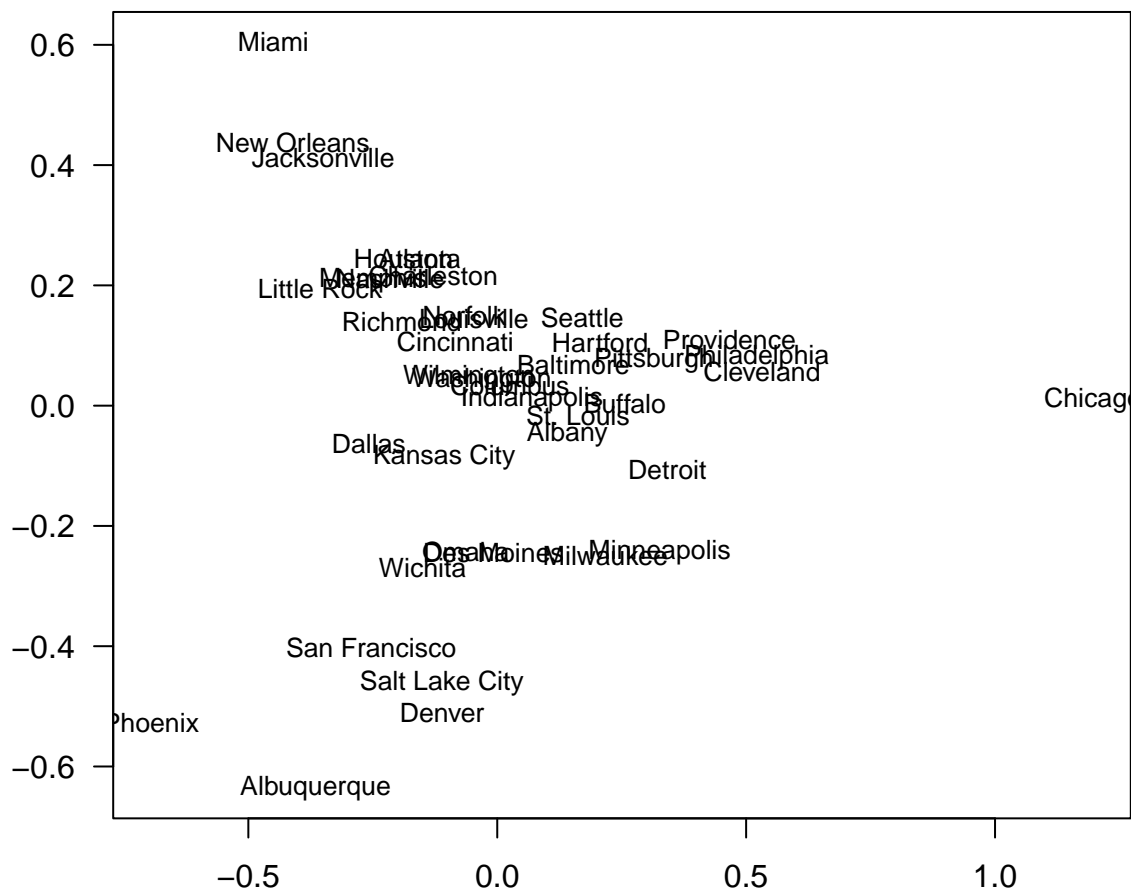
```

```
# Reduce to 3 Dimensions
(pol.mds3 <- cmdscale(poldist, k = 3, eig = TRUE))
```

```
## $points
##           [,1]      [,2]      [,3]
## Albany      0.140558172 -0.046859954  0.267632311
## Albuquerque -0.364824787 -0.636602091 -0.102087912
## Atlanta     -0.155922591  0.244511276 -0.046494117
## Baltimore   0.153189990  0.067519907 -0.089541686
## Buffalo     0.256244063  0.003022604  0.495974986
## Charleston  -0.128730958  0.215783429  0.150107702
## Chicago     1.197000315  0.009638168 -0.824422653
## Cincinnati  -0.084166097  0.106828800  0.039366111
## Cleveland   0.531787447  0.056305378  0.120608309
## Columbus    0.025412911  0.033574934  0.147295326
## Dallas      -0.258008194 -0.062640448 -0.244388882
## Denver      -0.110682033 -0.510378502 -0.029698665
## Des Moines  -0.007603614 -0.244344703  0.268379415
## Detroit     0.341537781 -0.105917971 -0.064362771
## Hartford    0.206766531  0.105259858  0.158528863
## Houston     -0.188167760  0.243707765 -0.266857401
## Indianapolis 0.069589745  0.010565926  0.086925283
## Jacksonville -0.349520267  0.412490203 -0.110633886
## Kansas City -0.106424371 -0.085118726  0.051579961
## Little Rock -0.355970056  0.194004542 -0.008211894
## Louisville  -0.046780470  0.144850917  0.020360112
## Memphis     -0.249259311  0.208737990 -0.062727649
## Miami       -0.449823739  0.604996816 -0.140722158
## Milwaukee   0.217298744 -0.249612250  0.271313047
## Minneapolis 0.326439578 -0.242858309  0.255203375
## Nashville   -0.215002650  0.211835269 -0.015978258
## New Orleans -0.410715158  0.438263300 -0.105127995
## Norfolk     -0.066285208  0.149134571  0.095300814
## Omaha       -0.063335982 -0.241936316  0.174099908
## Philadelphia 0.521031706  0.081089446 -0.416784375
## Phoenix     -0.695773353 -0.527859295 -0.571707666
## Pittsburgh   0.314965899  0.074640031  0.156669771
## Providence  0.466505620  0.110503750  0.124107741
## Richmond    -0.191967563  0.140461889 -0.002992969
## Salt Lake City -0.111111665 -0.461383196  0.055129138
## San Francisco -0.253430076 -0.401897024 -0.193036023
## Seattle     0.170829143  0.147411289  0.266888111
## St. Louis    0.162208664 -0.016576959 -0.128585601
## Washington  -0.031338057  0.041417952 -0.053942965
## Wichita     -0.149744969 -0.268806546  0.140296711
## Wilmington  -0.056777379  0.046236280  0.132538529
##
## $eig
## [1] 4.456648e+00 2.819944e+00 2.256196e+00 1.651762e+00 6.199354e-01
## [6] 1.904906e-01 3.068220e-02 1.558353e-15 9.406328e-16 2.494225e-16
## [11] 1.736021e-16 1.471280e-16 1.356518e-16 8.017147e-17 7.511957e-17
## [16] 6.686099e-17 5.684599e-17 5.034791e-17 4.025565e-17 3.312471e-17
## [21] 2.974204e-17 1.555983e-17 1.132251e-17 3.668800e-18 -5.206488e-18
```

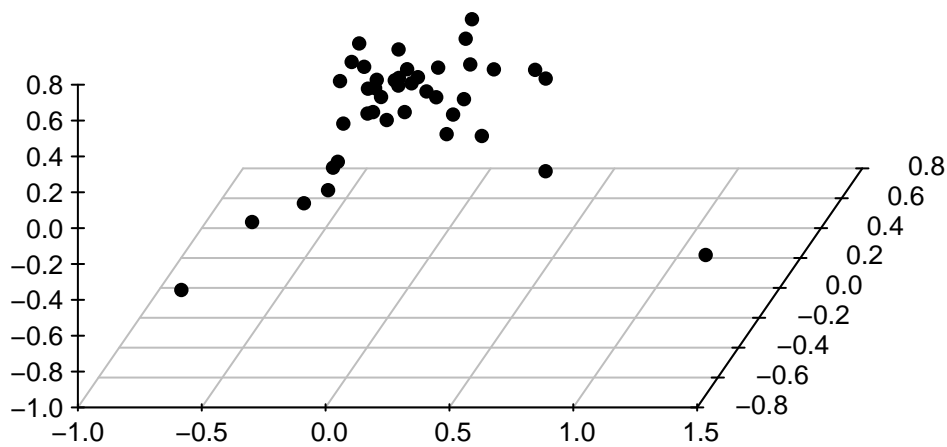
```
## [26] -8.948794e-18 -9.519928e-18 -1.506805e-17 -1.853275e-17 -2.314710e-17
## [31] -2.858271e-17 -3.093804e-17 -3.151435e-17 -3.396470e-17 -7.209856e-17
## [36] -7.714641e-17 -1.524915e-16 -2.390840e-16 -2.833661e-16 -3.238640e-16
## [41] -1.263609e-15
##
## $x
## NULL
##
## $ac
## [1] 0
##
## $GOF
## [1] 0.792704 0.792704
```

```
# Plot
par(las = 1, mgp = c(2, 1, 0), mar = c(3, 3, 1, 0.5))
x <- pol.mds$points
plot(x[, 1], x[, 2], type = "n", xlab = "", ylab = "")
text(x[, 1], x[, 2], labels = rownames(x), cex = 0.8)
```



```
library(scatterplot3d)
par(las = 1)
```

```
scatterplot3d(pol.mds3$points, pch = 16, angle = 75,
              box = F, xlab = "", ylab = "",
              zlab = "")
```



Non-Metric Multidimensional Scaling

House of Representatives Voting Data

```
# Matrix
data("voting", package = "HSAUR2")
voting
```

```
##           Hunt(R) Sandman(R) Howard(D) Thompson(D) Freylinghuysen(R)
## Hunt(R)           0         8        15          15             10
## Sandman(R)         8         0        17          12             13
## Howard(D)        15        17         0           9             16
## Thompson(D)       15        12         9           0             14
## Freylinghuysen(R) 10        13        16          14              0
## Forsythe(R)         9        13        12          12              8
## Widnall(R)          7        12        15          13              9
```


## Roe(D)	15	16	5	10	13	
## Heltoski(D)	16	17	5	8	14	
## Rodino(D)	14	15	6	8	12	
## Minish(D)	15	16	5	8	12	
## Rinaldo(R)	16	17	4	6	12	
## Maraziti(R)	7	13	11	15	10	
## Daniels(D)	11	12	10	10	11	
## Patten(D)	13	16	7	7	11	
##	Forsythe(R)	Widnall(R)	Roe(D)	Heltoski(D)	Rodino(D)	Minish(D)
## Hunt(R)	9	7	15	16	14	15
## Sandman(R)	13	12	16	17	15	16
## Howard(D)	12	15	5	5	6	5
## Thompson(D)	12	13	10	8	8	8
## Freylinghuysen(R)	8	9	13	14	12	12
## Forsythe(R)	0	7	12	11	10	9
## Widnall(R)	7	0	17	16	15	14
## Roe(D)	12	17	0	4	5	5
## Heltoski(D)	11	16	4	0	3	2
## Rodino(D)	10	15	5	3	0	1
## Minish(D)	9	14	5	2	1	0
## Rinaldo(R)	10	15	3	1	2	1
## Maraziti(R)	6	10	12	13	11	12
## Daniels(D)	6	11	7	7	4	5
## Patten(D)	10	13	6	5	6	5
##	Rinaldo(R)	Maraziti(R)	Daniels(D)	Patten(D)		
## Hunt(R)	16	7	11	13		
## Sandman(R)	17	13	12	16		
## Howard(D)	4	11	10	7		
## Thompson(D)	6	15	10	7		
## Freylinghuysen(R)	12	10	11	11		
## Forsythe(R)	10	6	6	10		
## Widnall(R)	15	10	11	13		
## Roe(D)	3	12	7	6		
## Heltoski(D)	1	13	7	5		
## Rodino(D)	2	11	4	6		
## Minish(D)	1	12	5	5		
## Rinaldo(R)	0	12	6	4		
## Maraziti(R)	12	0	9	13		
## Daniels(D)	6	9	0	9		
## Patten(D)	4	13	9	0		

```
names <- rownames(voting)
party <- gsub("[\\(\\)]", "", regmatches(names, gregexpr("\\(.*?\\)", names)))
col <- ifelse(party == "R", "red", "blue")

library(MASS)
voting_mds <- isoMDS(voting, k = 2)
```

```
## initial value 15.268246
## iter 5 value 10.264075
## final value 9.879047
## converged
```

```
str(voting_mds)
```

```
## List of 2
## $ points: num [1:15, 1:2] -8.44 -7.41 6.09 3.52 -7.25 ...
## ..- attr(*, "dimnames")=List of 2
## .. ..$ : chr [1:15] "Hunt(R)" "Sandman(R)" "Howard(D)" "Thompson(D)" ...
## .. ..$ : NULL
## $ stress: num 9.88
```

```
par(las = 1, mar = c(2, 2, 0.5, 0.5))
plot(voting_mds$points, type = "n", xlim = c(-12, 8),
      xlab = "", ylab = "")
text(voting_mds$points, labels = rownames(voting_mds$points),
      cex = 0.7, col = col)
```

