# HW1-SOLUTION

TA: Ao Sun

3/16/2022

## 1. Prove that the sample . . . .

Note $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^\top$ and $\bar{\mathbf{y}} = \frac{1}{n}\mathbf{Y}^\top\mathbf{1}$

$$
\begin{aligned}
\mathbf{S} &= \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{y}_i - \bar{\mathbf{y}})(\mathbf{y}_i - \bar{\mathbf{y}})^\top \\
&= \frac{1}{n-1}(\sum_{i=1}^{n}\mathbf{y}_i\mathbf{y}_i^\top - n\bar{\mathbf{y}}\bar{\mathbf{y}}^\top) \\
&= \frac{1}{n-1}(\mathbf{Y}^\top\mathbf{Y} - \frac{1}{n}\mathbf{Y}^\top\mathbf{1}\mathbf{1}^\top\mathbf{Y}) \\
&= \frac{1}{n-1}(\mathbf{Y}^\top\mathbf{Y} - \frac{1}{n}\mathbf{Y}^\top\mathbf{J}\mathbf{Y}) \\
&= \frac{1}{n-1}\mathbf{Y}^\top(\mathbf{I} - \frac{1}{n}\mathbf{J})\mathbf{Y}
\end{aligned}
$$

Similarly, note that $\bar{\mathbf{Y}} = \bar{\mathbf{y}}\mathbf{1}^\top = \frac{1}{n}\mathbf{Y}^\top\mathbf{1}\mathbf{1}^\top$, which suffice to show second equality.

## 2. Let matrix $A$ be. . .

### (a) $\mathbf{A}\mathbf{A}^\top$. . .

```
A <- matrix(c(4,8,8,3,6,-9), ncol = 3, nrow = 2, byrow = T)
A %*% t(A)
```

```
##      [,1] [,2]
## [1,]  144  -12
## [2,]  -12  126
```

the eigenvalues are 150, 120, the corresponding eigenvectors are $[2/\sqrt{5}, -1/\sqrt{5}]$ and $[/\sqrt{5}, 2/\sqrt{5}]$.

### (b) $\mathbf{A}^\top\mathbf{A}$. . .

```
t(A) %*% A
```

```
##      [,1] [,2] [,3]
## [1,]   25   50    5
## [2,]   50  100   10
## [3,]    5   10  145
```

the eigenvalues are 150, 120 and 0, the corresponding eigenvectors are $[1/\sqrt{30}, 2/\sqrt{30}, 5/\sqrt{30}]$, $[1/\sqrt{6}, 2/\sqrt{6}, -1/\sqrt{6}]$ and $[-2/\sqrt{5}, 1/\sqrt{5}, 0]$.

**(c) Obtain the spectral decomposition**

$$\mathbf{A}\mathbf{A}^\top = \begin{bmatrix} \frac{2}{\sqrt{5}}, & \frac{1}{\sqrt{5}} \\ -\frac{1}{\sqrt{5}}, & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} 150, & 0 \\ 0, & 120 \end{bmatrix} \begin{bmatrix} \frac{2}{\sqrt{5}}, & -\frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}}, & \frac{2}{\sqrt{5}} \end{bmatrix}$$

### (d) Self-study the definition… Since $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, where $\mathbf{U}^\top\mathbf{U} = \mathbf{I}$ and $\mathbf{V}^\top\mathbf{V} = \mathbf{I}$, we have

$$A = \begin{pmatrix} \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{pmatrix} \begin{pmatrix} 5\sqrt{6} & 0 & 0 \\ 0 & 2\sqrt{30} & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{5}{\sqrt{30}} \\ \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{6}} \\ -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} & 0 \end{pmatrix}$$

## 3. Suppose the random vector…

$$\begin{aligned}
E\mathbf{y}^{(1)} &= (4,3)', \\
E\mathbf{A}\mathbf{y}^{(1)} &= (1,2)\begin{pmatrix}4\\3\end{pmatrix} = 10, \\
COV\mathbf{y}^{(1)} &= \begin{pmatrix}3 & 0\\0 & 1\end{pmatrix}, \\
COV\mathbf{A}\mathbf{y}^{(1)} &= (1,2)\begin{pmatrix}3 & 0\\0 & 1\end{pmatrix}\begin{pmatrix}1\\2\end{pmatrix} = 7, \\
E\mathbf{B}\mathbf{y}^{(2)} &= \begin{pmatrix}1 & -2\\2 & -1\end{pmatrix}\begin{pmatrix}2\\1\end{pmatrix} = \begin{pmatrix}0\\3\end{pmatrix}, \\
COV\mathbf{B}\mathbf{y}^{(2)} &= \begin{pmatrix}1 & -2\\2 & -1\end{pmatrix}\begin{pmatrix}9 & -2\\-2 & 4\end{pmatrix}\begin{pmatrix}1 & 2\\-2 & -1\end{pmatrix} = \begin{pmatrix}13 & -10\\20 & -8\end{pmatrix}\begin{pmatrix}1 & 2\\-2 & -1\end{pmatrix} = \begin{pmatrix}33 & 36\\36 & 48\end{pmatrix}, \\
COV(\mathbf{y}^{(1)}, \mathbf{y}^{(2)}) &= \begin{pmatrix}2 & 2\\1 & 0\end{pmatrix}, \\
COV(\mathbf{A}\mathbf{y}^{(1)}, \mathbf{B}\mathbf{y}^{(2)}) &= (1,2)\begin{pmatrix}2 & 2\\1 & 0\end{pmatrix}\begin{pmatrix}1 & 2\\-2 & -1\end{pmatrix} = (4,2)\begin{pmatrix}1 & 2\\-2 & -1\end{pmatrix} = (0,6).
\end{aligned}$$

## 4. (R exercise.) The following table (data attached) gives…

```r
rm(list = ls())
y <- read.table('./CALCIUM.DAT')
y
```

```
##    V1 V2   V3   V4
## 1   1 35  3.5 2.80
## 2   2 35  4.9 2.70
## 3   3 40 30.0 4.38
## 4   4 10  2.8 3.21
## 5   5  6  2.7 2.73
## 6   6 20  2.8 2.81
## 7   7 35  4.6 2.88
## 8   8 35 10.9 2.90
## 9   9 35  8.0 3.28
## 10 10 30  1.6 3.20
```

```r
y1 <- y$V2
y2 <- y$V3
y3 <- y$V4
z1 <- y1 + y2 + y3
```

```r
z2 <- 2 * y1 - 3 * y2 + 2 * y3
z3 <- -1 * y1 - 2 * y2 - 3 * y3
z <- data.frame(z1, z2, z3)
```

#### (a) Find the sample mean vector…

```r
colMeans(z)
```

```
##      z1      z2      z3
##  38.369  40.838 -51.727
```

#### …sample covariance matrix…

```r
S <- cov(z)
S
```

```
##           z1       z2        z3
## z1  323.6376  19.2526 -460.9770
## z2   19.2526 588.6710  104.0717
## z3 -460.9770 104.0717  686.2697
```

#### (b) Find the sample correlation matrix…

```r
D <- diag(1/sqrt(diag(S)))
R <- D %*% S %*% D
R
```

```
##              [,1]       [,2]       [,3]
## [1,]  1.00000000 0.04410862 -0.9781430
## [2,]  0.04410862 1.00000000  0.1637378
## [3,] -0.97814302 0.16373782  1.0000000
```

#### (c) Find the generalized variance…

```r
det(S)
```

```
## [1] 45995.55
```

#### …and total variance of…

```r
sum(diag(S))
```

```
## [1] 1598.578
```

#### (d) The spectral decomposition of $\mathbf{S}_z$,

```r
S.eig <- eigen(S)
S.eig
```

```
## eigen() decomposition
## $values
## [1] 1.013775e+03 5.847259e+02 7.759291e-02
##
## $vectors
```

```
##             [,1]        [,2]       [,3]
## [1,] -0.5433288  0.20391455  0.8143787
## [2,]  0.1763352  0.97613268 -0.1267711
## [3,]  0.8207921 -0.07472522  0.5663183
```

```
S.eig$vectors %*% diag(S.eig$values) %*% t(S.eig$vectors) # double check
```

```
##           [,1]     [,2]      [,3]
## [1,]  323.6376  19.2526 -460.9770
## [2,]   19.2526 588.6710  104.0717
## [3,] -460.9770 104.0717  686.2697
```

**the sqare root matrix of $\mathbf{S}_z$,**

```
S.eig$vectors %*% diag(sqrt(S.eig$values)) %*% t(S.eig$vectors)
```

```
##            [,1]      [,2]       [,3]
## [1,]  10.589534  1.733925 -14.439283
## [2,]   1.733925 24.035112   2.824513
## [3,] -14.439283  2.824513  21.674846
```

```
library(pracma)
sqrtm(S)$B # alternatively
```

```
##            z1        z2         z3
## z1  10.589534  1.733925 -14.439283
## z2   1.733925 24.035112   2.824513
## z3 -14.439283  2.824513  21.674846
```

```
sqrtm(S)$B %*% sqrtm(S)$B # double check
```

```
##           z1       z2        z3
## z1  323.6376  19.2526 -460.9770
## z2   19.2526 588.6710  104.0717
## z3 -460.9770 104.0717  686.2697
```

**the Cholesky decomposition (also the square root matrix) of $\mathbf{S}_z$,**

```
chol(S)
```

```
##          z1        z2         z3
## z1 17.98993  1.070187 -25.624168
## z2  0.00000 24.238929   5.424925
## z3  0.00000  0.000000   0.491830
```

```
t(chol(S)) %*% chol(S) # double check
```

```
##           z1       z2        z3
## z1  323.6376  19.2526 -460.9770
## z2   19.2526 588.6710  104.0717
## z3 -460.9770 104.0717  686.2697
```

**the spectral decomposition of $\mathbf{R}_z$,**

```
R.eig <- eigen(R)
R.eig
```

```
## eigen() decomposition
## $values
## [1] 1.9854859438 1.0143393778 0.0001746784
##
## $vectors
##             [,1]       [,2]       [,3]
## [1,] -0.69986611 0.16435410  0.6951080
## [2,]  0.08647836 0.98550551 -0.1459465
## [3,]  0.70901969 0.04203123  0.7039350
```

```
R.eig$vectors %*% diag(R.eig$values) %*% t(R.eig$vectors) # double check
```

```
##             [,1]       [,2]       [,3]
## [1,]  1.00000000 0.04410862 -0.9781430
## [2,]  0.04410862 1.00000000  0.1637378
## [3,] -0.97814302 0.16373782  1.0000000
```

**the sqare root matrix of $\mathbf{R}_z$,**

```
R.eig$vectors %*% diag(sqrt(R.eig$values)) %*% t(R.eig$vectors)
```

```
##             [,1]       [,2]       [,3]
## [1,]  0.72377273 0.07650653 -0.6857841
## [2,]  0.07650653 0.98897895  0.1267572
## [3,] -0.68578407 0.12675720  0.7166818
```

```
sqrtm(R)$B # alternatively
```

```
##             [,1]       [,2]       [,3]
## [1,]  0.72377274 0.07650653 -0.6857841
## [2,]  0.07650653 0.98897895  0.1267572
## [3,] -0.68578405 0.12675719  0.7166818
```

```
sqrtm(R)$B %*% sqrtm(R)$B # double check
```

```
##             [,1]       [,2]       [,3]
## [1,]  1.00000000 0.04410862 -0.9781430
## [2,]  0.04410862 1.00000000  0.1637378
## [3,] -0.97814302 0.16373782  1.0000000
```

**the Cholesky decomposition (also the square root matrix) of $\mathbf{R}_z$.**

```
chol(R)
```

```
##      [,1]       [,2]        [,3]
## [1,]    1 0.04410862 -0.97814302
## [2,]    0 0.99902674  0.20708391
## [3,]    0 0.00000000  0.01877447
```

```
t(chol(R)) %*% chol(R) # double check
```

```
##             [,1]       [,2]       [,3]
## [1,]  1.00000000 0.04410862 -0.9781430
## [2,]  0.04410862 1.00000000  0.1637378
## [3,] -0.97814302 0.16373782  1.0000000
```

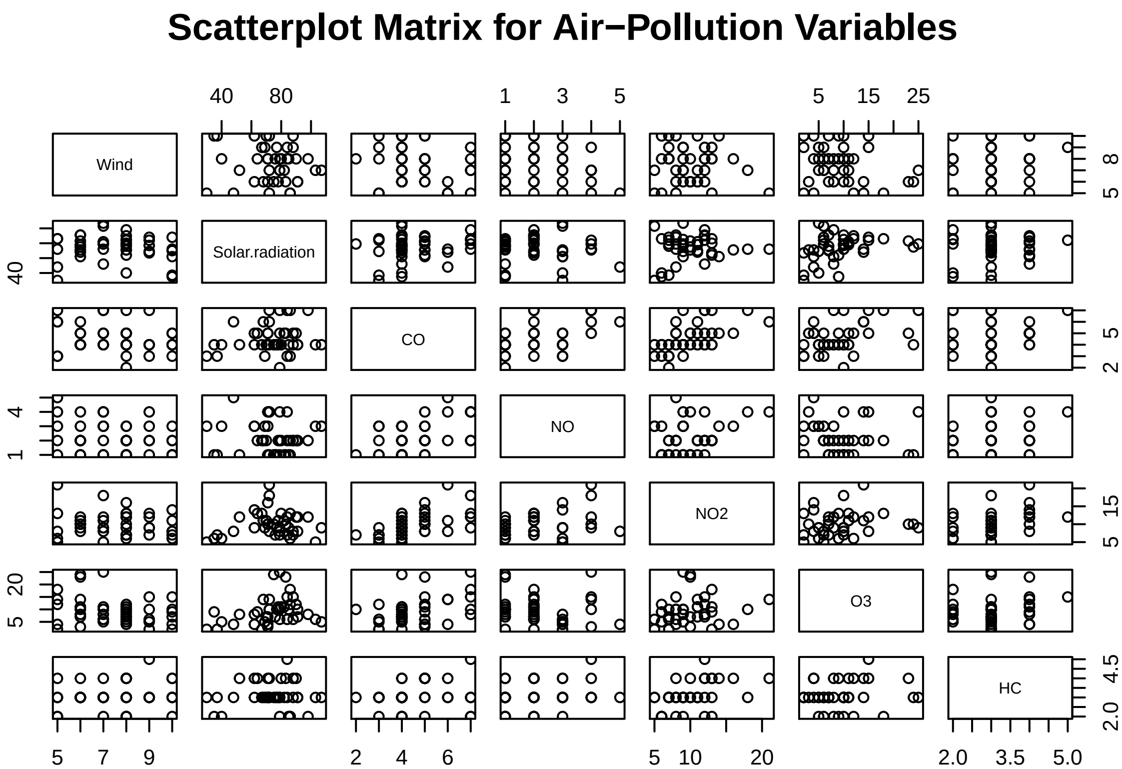
**5. (R exercise.) The attached data are 42 measurements on air-pollution variables recorded at 12:00 noon in the Los Angeles area on different days.**

```r
rm(list = ls())
y <- read.table(paste('./DATA_pollution.txt'), header = TRUE)
y
```

```
##    Wind Solar.radiation CO NO NO2 O3 HC
## 1     8              98  7  2  12  8  2
## 2     7             107  4  3   9  5  3
## 3     7             103  4  3   5  6  3
## 4    10              88  5  2   8 15  4
## 5     6              91  4  2   8 10  3
## 6     8              90  5  2  12 12  4
## 7     9              84  7  4  12 15  5
## 8     5              72  6  4  21 14  4
## 9     7              82  5  1  11 11  3
## 10    8              64  5  2  13  9  4
## 11    6              71  5  4  10  3  3
## 12    6              91  4  2  12  7  3
## 13    7              72  7  4  18 10  3
## 14   10              70  4  2  11  7  3
## 15   10              72  4  1   8 10  3
## 16    9              77  4  1   9 10  3
## 17    8              76  4  1   7  7  3
## 18    8              71  5  3  16  4  4
## 19    9              67  4  2  13  2  3
## 20    9              69  3  3   9  5  3
## 21   10              62  5  3  14  4  4
## 22    9              88  4  2   7  6  3
## 23    8              80  4  2  13 11  4
## 24    5              30  3  3   5  2  3
## 25    6              83  5  1  10 23  4
## 26    8              84  3  2   7  6  3
## 27    6              78  4  2  11 11  3
## 28    8              79  2  1   7 10  3
## 29    6              62  4  3   9  8  3
## 30   10              37  3  1   7  2  3
## 31    8              71  4  1  10  7  3
## 32    7              52  4  1  12  8  4
## 33    5              48  6  5   8  4  3
## 34    6              75  4  1  10 24  3
## 35   10              35  4  1   6  9  2
## 36    8              85  4  1   9 10  2
## 37    5              86  3  1   6 12  2
## 38    5              86  7  2  13 18  2
## 39    7              79  7  4   9 25  3
## 40    7              79  5  2   8  6  2
## 41    6              68  6  2  11 14  3
## 42    8              40  4  3   6  5  2
```

**(a) Plot the pairwise scatter plot matrix for all the variables in R. And comment on the output.**

```r
pairs(y, main = "Scatterplot Matrix for Air-Pollution Variables")
```




As illustrated in the picture above, data on most graphs scattered on the grids, which indicates the discontinuity of most variables. Note that a week relationship appeared between gases (CO, NO, etc.) and other variables (wind and solar radiation).

**(b) Construct the sample mean vector, sample covariance matrix and sample correlation matrix. Interpret the entries in the sample correlation matrix.**

```r
colMeans(y)
```

```
##            Wind Solar.radiation              CO              NO             NO2 
##        7.500000       73.857143        4.547619        2.190476       10.047619 
##              O3              HC 
##        9.404762        3.095238
```

```r
cov(y)
```

```
##                       Wind Solar.radiation         CO         NO        NO2
## Wind             2.5000000      -2.7804878 -0.3780488 -0.4634146 -0.5853659
## Solar.radiation -2.7804878     300.5156794  3.9094077 -1.3867596  6.7630662
## CO              -0.3780488       3.9094077  1.5220674  0.6736353  2.3147503
## NO              -0.4634146      -1.3867596  0.6736353  1.1823461  1.0882695
## NO2             -0.5853659       6.7630662  2.3147503  1.0882695 11.3635308
## O3              -2.2317073      30.7909408  2.8217189 -0.8106852  3.1265970
## HC               0.1707317       0.6236934  0.1416957  0.1765389  1.0441347
##                         O3        HC
## Wind            -2.2317073 0.1707317
## Solar.radiation 30.7909408 0.6236934
```

```
## CO                2.8217189 0.1416957
## NO               -0.8106852 0.1765389
## NO2               3.1265970 1.0441347
## O3               30.9785134 0.5946574
## HC                0.5946574 0.4785134
```

```r
cor(y)
```

```
##                       Wind Solar.radiation         CO          NO         NO2
## Wind             1.0000000     -0.10144191 -0.1938032 -0.26954261 -0.1098249
## Solar.radiation -0.1014419      1.00000000  0.1827934 -0.07356907  0.1157320
## CO              -0.1938032      0.18279338  1.0000000  0.50215246  0.5565838
## NO              -0.2695426     -0.07356907  0.5021525  1.00000000  0.2968981
## NO2             -0.1098249      0.11573199  0.5565838  0.29689814  1.0000000
## O3              -0.2535928      0.31912373  0.4109288 -0.13395214  0.1666422
## HC               0.1560979      0.05201044  0.1660323  0.23470432  0.4477678
##                         O3         HC
## Wind            -0.2535928 0.15609793
## Solar.radiation  0.3191237 0.05201044
## CO               0.4109288 0.16603235
## NO              -0.1339521 0.23470432
## NO2              0.1666422 0.44776780
## O3               1.0000000 0.15445056
## HC               0.1544506 1.00000000
```

Most pairs of variables appeared weak correlations, few pairs (e.g., CO and NO, CO and NO2) appeared moderate correlations, and no pairs appeared a strong correlation, which coincides with the pairwise scatterplot.

### (c) Compute the Eucleadian distance matrix and the Mahalanobis/statistical distance matrix among the first five days. Explain the advantage of the Mahalanobis distance.

```r
dist(y[1:5,]) # the Eucleadian distance matrix
```

```
##           1         2         3         4
## 2 10.535654
## 3  9.486833  5.744563
## 4 13.304135 21.771541 18.083141
## 5  9.110434 16.852300 13.076697  7.211103
```

```r
library(pracma)
S <- cov(y)
S # the sample covariance matrix
```

```
##                       Wind Solar.radiation         CO         NO        NO2
## Wind             2.5000000      -2.7804878 -0.3780488 -0.4634146 -0.5853659
## Solar.radiation -2.7804878     300.5156794  3.9094077 -1.3867596  6.7630662
## CO              -0.3780488       3.9094077  1.5220674  0.6736353  2.3147503
## NO              -0.4634146      -1.3867596  0.6736353  1.1823461  1.0882695
## NO2             -0.5853659       6.7630662  2.3147503  1.0882695 11.3635308
## O3              -2.2317073      30.7909408  2.8217189 -0.8106852  3.1265970
## HC               0.1707317       0.6236934  0.1416957  0.1765389  1.0441347
##                         O3        HC
## Wind            -2.2317073 0.1707317
## Solar.radiation 30.7909408 0.6236934
## CO               2.8217189 0.1416957
## NO              -0.8106852 0.1765389
```

```
## NO2              3.1265970 1.0441347
## O3              30.9785134 0.5946574
## HC               0.5946574 0.4785134
```

```r
y.tsf <- as.matrix(y) %*% sqrtm(S)$Binv
dist(y.tsf[1:5,]) # the Mahalanobis/statistical distance matrix
```

```
##          1        2        3        4
## 2 4.221941
## 3 4.518621 1.626539
## 4 4.694563 3.811112 3.402224
## 5 4.097358 2.063497 2.099450 3.313883
```

Comparing to the Eucleadian distance, the Mahalanobis distance refined the distance information leaving out variation of some variables or correlation between some pairs of variables.

### (d) Describe the overall variability of the data.

```r
det(S) # generalized sample variance
```

```
## [1] 35307.53
```

```r
sum(diag(S)) # total sample variance
```

```
## [1] 348.5407
```

### (e) Get the Spectral decomposition and Cholesky decomposition of the sample covariance matrix. Observe the difference between the two decompositions.

```r
S.eig <- eigen(S)
S.eig # the spectral decomposition
```

```
## eigen() decomposition
## $values
## [1] 304.2578640  28.2761046  11.4644830   2.5243296   1.2795247   0.5287288
## [7]   0.2096157
##
## $vectors
##              [,1]        [,2]        [,3]          [,4]          [,5]
## [1,]  0.010039244  0.07622439  0.03087761  0.9203045748  0.3423859285
## [2,] -0.993199405  0.11615518  0.00659069 -0.0002118679  0.0022391022
## [3,] -0.014062314 -0.09956775 -0.18282641 -0.1382922410  0.6500776063
## [4,]  0.004710175  0.01320423 -0.13021553 -0.3277842624  0.6431560485
## [5,] -0.024255644 -0.15038113 -0.95526318  0.1023719020 -0.2065840405
## [6,] -0.112429558 -0.97335904  0.16981025  0.0632480276 -0.0002935726
## [7,] -0.002340785 -0.02382046 -0.08519558  0.1095073458  0.0619613872
##              [,6]         [,7]
## [1,]  0.011779079 -0.169729925
## [2,]  0.003353218 -0.001781987
## [3,] -0.563893916  0.443577538
## [4,]  0.497513370 -0.462855916
## [5,] -0.009009299 -0.105029951
## [6,]  0.051067254 -0.066992404
## [7,]  0.657012233  0.738019426
```

```r
S.eig$vectors %*% diag(S.eig$values) %*% t(S.eig$vectors) # double check
```

```
##            [,1]        [,2]       [,3]       [,4]       [,5]       [,6]
## [1,]  2.5000000  -2.7804878 -0.3780488 -0.4634146 -0.5853659 -2.2317073
## [2,] -2.7804878 300.5156794  3.9094077 -1.3867596  6.7630662 30.7909408
## [3,] -0.3780488   3.9094077  1.5220674  0.6736353  2.3147503  2.8217189
## [4,] -0.4634146  -1.3867596  0.6736353  1.1823461  1.0882695 -0.8106852
## [5,] -0.5853659   6.7630662  2.3147503  1.0882695 11.3635308  3.1265970
## [6,] -2.2317073  30.7909408  2.8217189 -0.8106852  3.1265970 30.9785134
## [7,]  0.1707317   0.6236934  0.1416957  0.1765389  1.0441347  0.5946574
##           [,7]
## [1,] 0.1707317
## [2,] 0.6236934
## [3,] 0.1416957
## [4,] 0.1765389
## [5,] 1.0441347
## [6,] 0.5946574
## [7,] 0.4785134
```

```r
chol(S) # the Cholesky decomposition
```

```
##                     Wind Solar.radiation        CO         NO         NO2
## Wind            1.581139       -1.758535 -0.239099 -0.2930891 -0.37021787
## Solar.radiation 0.000000       17.245963  0.202305 -0.1102964  0.35440324
## CO              0.000000        0.000000  1.193303  0.5244867  1.80552154
## NO              0.000000        0.000000  0.000000  0.8995517  0.07990644
## NO2             0.000000        0.000000  0.000000  0.0000000  2.79903104
## O3              0.000000        0.000000  0.000000  0.0000000  0.00000000
## HC              0.000000        0.000000  0.000000  0.0000000  0.00000000
##                         O3         HC
## Wind            -1.4114556 0.10798021
## Solar.radiation  1.6414767 0.04717512
## CO               1.8035341 0.13238040
## NO              -2.2113774 0.16003329
## NO2             -0.3777415 0.29138247
## O3               4.2433768 0.21087886
## HC               0.0000000 0.54048060
```

```r
t(chol(S)) %*% chol(S) # double check
```

```
##                       Wind Solar.radiation         CO         NO        NO2
## Wind             2.5000000      -2.7804878 -0.3780488 -0.4634146 -0.5853659
## Solar.radiation -2.7804878     300.5156794  3.9094077 -1.3867596  6.7630662
## CO              -0.3780488       3.9094077  1.5220674  0.6736353  2.3147503
## NO              -0.4634146      -1.3867596  0.6736353  1.1823461  1.0882695
## NO2             -0.5853659       6.7630662  2.3147503  1.0882695 11.3635308
## O3              -2.2317073      30.7909408  2.8217189 -0.8106852  3.1265970
## HC               0.1707317       0.6236934  0.1416957  0.1765389  1.0441347
##                         O3        HC
## Wind            -2.2317073 0.1707317
## Solar.radiation 30.7909408 0.6236934
## CO               2.8217189 0.1416957
## NO              -0.8106852 0.1765389
## NO2              3.1265970 1.0441347
## O3              30.9785134 0.5946574
```

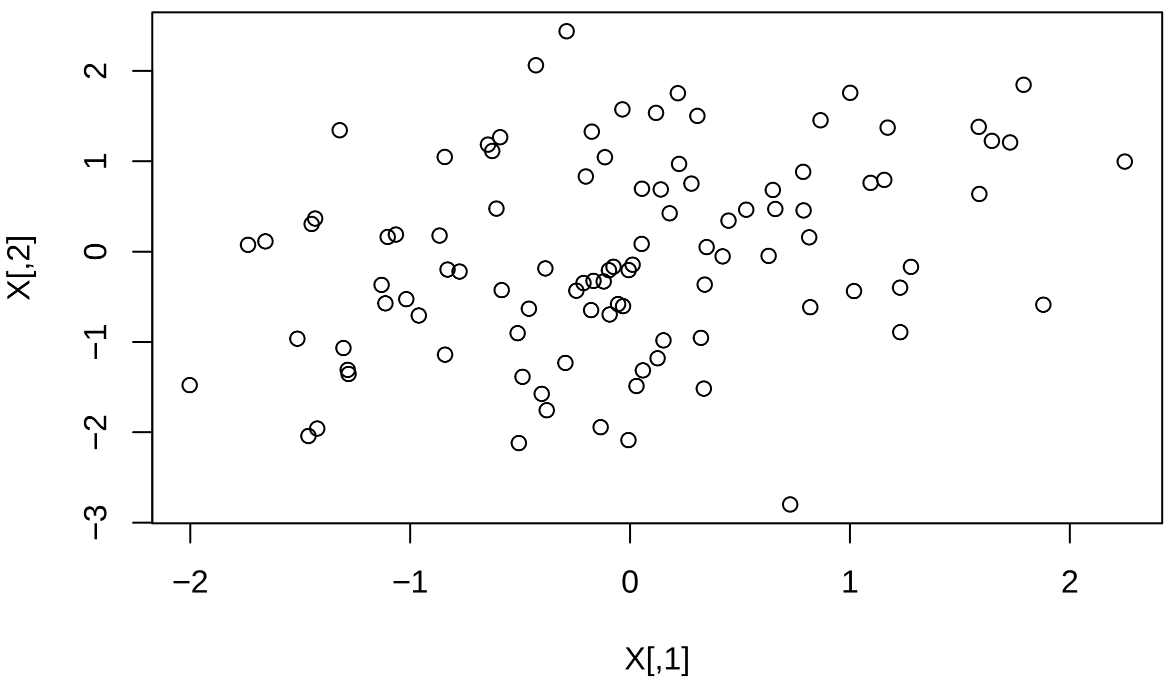
```
## HC                0.5946574 0.4785134
```

The spectral decomposition decomposed the sample covariance matrix into two orthogonal matrices and one diagonal matrix, whereas the Cholesky decomposition decomposed it into an upper triangular matrix with a transpose.

## 6. Generate 100 random pairs...

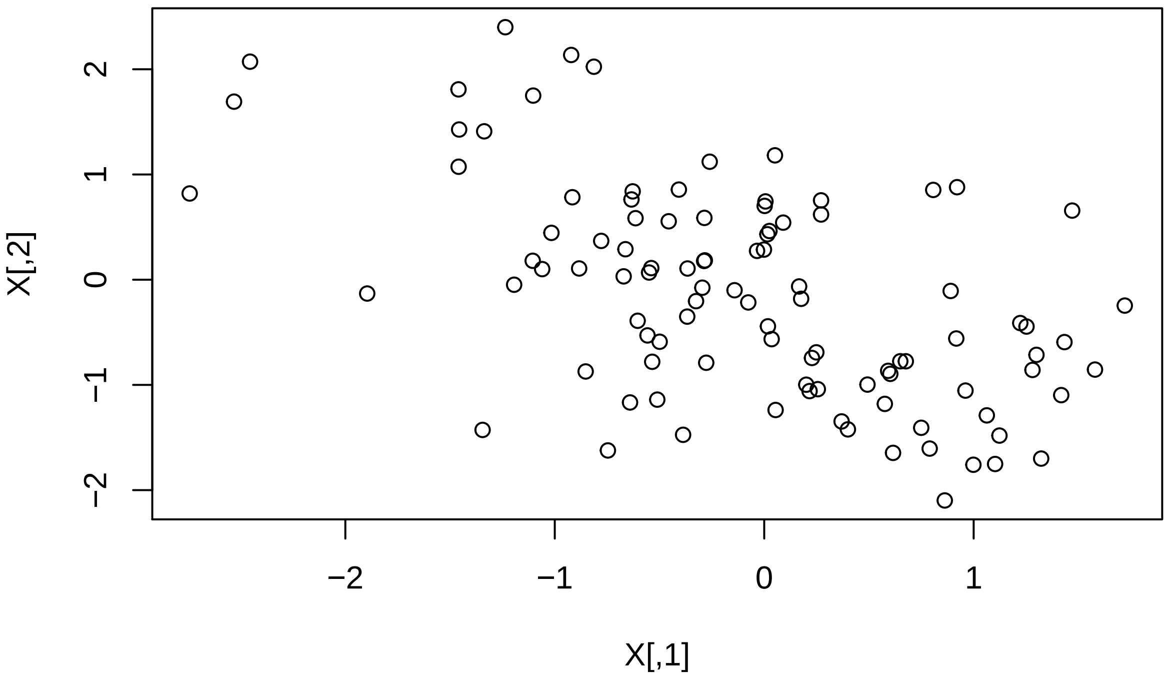
### (a) X and Y are positively correlated

```
n = 100
p = 2
Sigma = matrix(c(1,0.5,0.5,1),ncol = 2)
X = matrix(rnorm(n*p),n) %*% chol(Sigma)
plot(X)
```



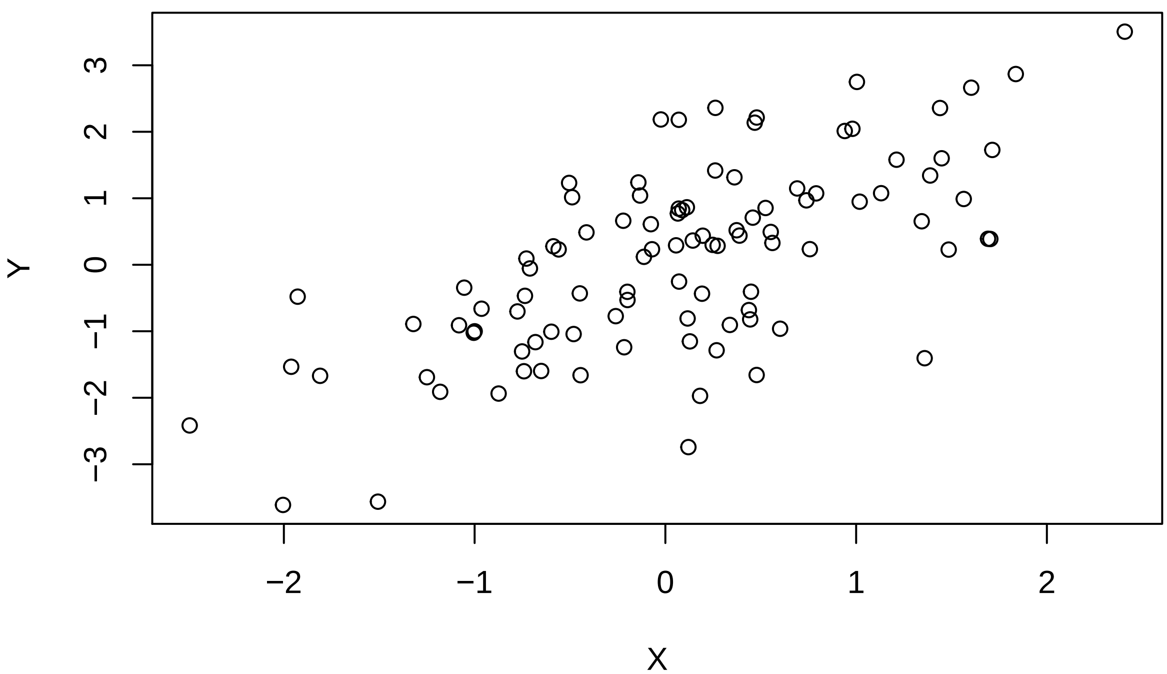

### (b) X and Y are negatively correlated;

```
Sigma = matrix(c(1,-0.5,-0.5,1),ncol = 2)
X = matrix(rnorm(n*p),n) %*% chol(Sigma)
plot(X)
```




**(c) X and Y are perfectly positive-correlated;**

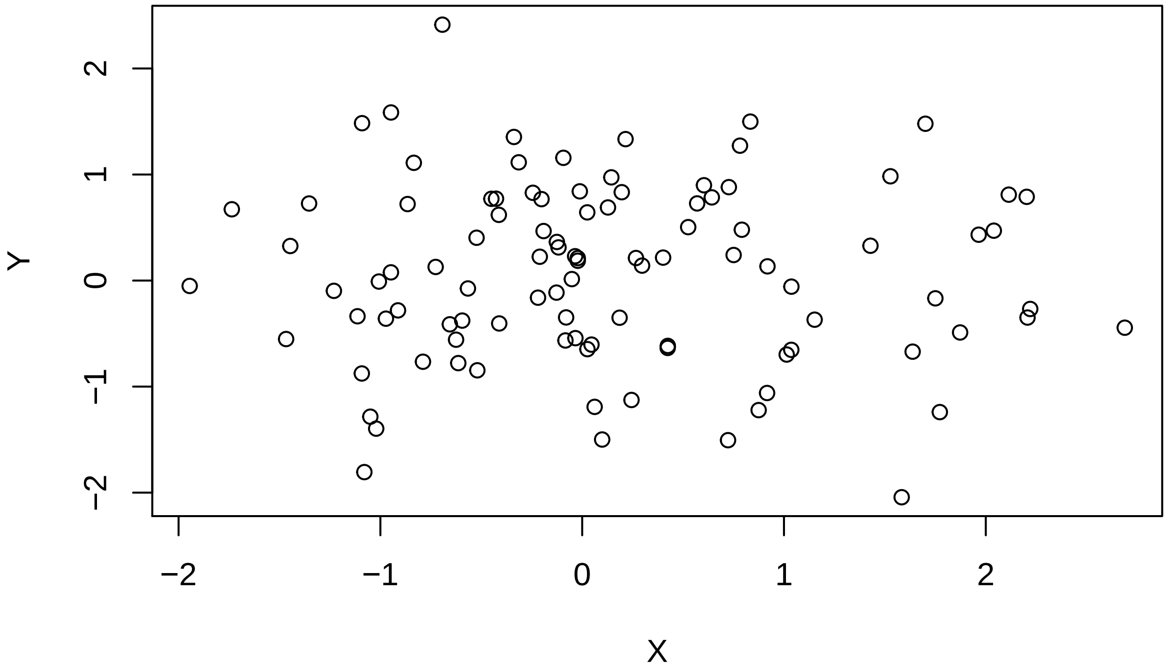
```
X =rnorm(n)
Y = X + rnorm(n)
plot(X,Y)
```




**(d) X and Y are uncorrelated;**

```
X =rnorm(n)
Y =rnorm(n)
```

```
plot(X,Y)
```




**{e} X and Y are nonlinearly correlated.**

```
X =rnorm(n)
Y = X^2 + X + rnorm(n)
plot(X,Y)
```