Solution: Math524 Midterm Exam by Dr. Samuel Shen, FA2023

October 11, Wednesday, due 11:59PM the next day

This is an open-book exam. You can use Google or ChatGPT to search for any helpful information. You can use a calculator, R, computer, or experiment. However, you CANNOT ask natural person to help you. When done, please submit **a single pdf file and a single R file** *online via Canvas.*

Student Name:

Student ID Number:

1. [20 $points$] **R for space-time data matrix**
   Use R to read the `NOAAGlobalT.csv` data file, and Select the four grid boxes that cover the following four locations: San Diego (USA), New York (USA), Paris (France), and Tokyo (Japan). Generate a 4×7 space-time data matrix for the January mean surface air temperature anomaly data of the four selected grid boxes and the seven years from1981 to 1987.
   Hint: You can download the NOAA Global Surface Temperature (NOAAGlobalTemp.csv) dataset from this midterm site on Canvas. Use the homework solution as a reference.
   Solution:

```
#Problem 1: R for space-time data matrix
setwd('/Users/sshen/Desktop/teach/524LinAlg2023Fall')
dat = read.csv('data/NOAAGlobalT.csv',
               header = TRUE)
dat[1:2, 1:8]
#  X   LAT LON X1880.1 X1880.2 X1880.3 X1880.4 X1880.5
#1 1 -87.5 2.5  -999.9  -999.9  -999.9  -999.9  -999.9
#2 2 -87.5 7.5  -999.9  -999.9  -999.9  -999.9  -999.9
dim(dat)
#[1] 2592 1648
#I would like to select the following four locations:
#San Diego (32.7157 N, 117.1611 W),
#New York (40.7128 N, 74.0060 W)
#Paris (48.8566 N, 2.3522 E)
#Tokyo (35.6764 N, 139.6500 E)
#The dataset uses longitude 0 to 360.
#For San Diego, 117W corresponds to 243 (=360-117).
#Search for San Diego (32.7157 N, 117.1611 W)
nsd = which(dat$LAT > 30 & dat$LAT < 35 &
              dat$LON > 240 & dat$LON < 245)
nsd
#[1] 1777 #Row 1777 is the grid box that covers San Diego
dat[nsd, 1:6]
#        X  LAT   LON X1880.1 X1880.2 X1880.3
#1777 1777 32.5 242.5  -1.984 -2.0391 -1.9442

#Search for New York (40.7128 N, 74.0060 W)
nny = which(dat$LAT > 40 & dat$LAT < 45 &
              dat$LON > 280 & dat$LON < 285)
nny
#[1] 1929
dat[nny, 1:6]
#        X  LAT   LON X1880.1 X1880.2 X1880.3
#1929 1929 42.5 282.5  3.7122  0.9924  -1.673
```

```
#Search for Paris (48.8566 N, 2.3522 E)
npa = which(dat$LAT > 45 & dat$LAT < 50 &
              dat$LON > 0 & dat$LON < 5)
npa
#[1] 1945
dat[npa, 1:6]
#        X  LAT   LON X1880.1 X1880.2 X1880.3
#1945 1945 47.5 2.5 -5.1221  0.0683  1.9522

#Search for Tokyo (35.6764 N, 139.6500 E)
nto = which(dat$LAT > 35 & dat$LAT < 40 &
              dat$LON > 135 & dat$LON < 140)
nto
#[1] 1828
dat[nto, 1:6]
#        X  LAT   LON X1880.1 X1880.2 X1880.3
#1828 1828 37.5 137.5 -0.2221  0.1473  0.2916

#Extract data for those four grid boxes
dat4 = dat[c(nsd, nny, npa, nto),]
dat4[, 1:6] #the first 6 columns of data
#        X  LAT   LON X1880.1 X1880.2 X1880.3
#1777 1777 32.5 242.5 -1.9840 -2.0391 -1.9442
#1929 1929 42.5 282.5  3.7122  0.9924 -1.6730
#1945 1945 47.5   2.5 -5.1221  0.0683  1.9522
#1828 1828 37.5 137.5 -0.2221  0.1473  0.2916

#I will use the Jan in 1981-1987 data
dat4[,4] # corresponds to Jan 1880
#For Jan 1981, 4 + 12*101 = 1216
dat4[,1216:1218] #Verify that Jan 1981 is the first column
#     X1981.1 X1981.2 X1981.3
#1777  0.9260  0.5872  0.1366
#1929 -3.1299  2.4882 -0.1373
dat47 = dat4[, 1216:1300]
dat47[,80:84] #Verify the data to Dec 1987
#     X1987.8 X1987.9 X1987.10 X1987.11 X1987.12
#1777 -0.4747  0.0816   1.0341   0.2833  -0.5720
#1929 -0.0293  0.2338  -1.6259   0.2618   1.4610

#Extract only the Jan data by an index
JanIndex = seq(1, 83, by = 12)
JanIndex #The seven columns for DJan
#[1] 1 13 25 37 49 61 73
datJan = dat47[,JanIndex]
datJan
#     X1981.1 X1982.1 X1983.1 X1984.1 X1985.1 X1986.1 X1987.1
#1777  0.9260 -0.7491  1.2609  1.1079 -0.8644  0.9502 -0.3180
#1929 -3.1299 -3.0661  1.5059 -1.7508 -1.5179  0.3121  0.2546
#1945 -0.5337  1.1086  1.7997  0.5446 -5.3362  0.1774 -5.6329
#1828 -1.1684 -0.3927  0.1208 -1.0764 -1.1135 -1.2139 -0.1822

#Put proper row and column names
rownames(datJan) <-c('San Diego', 'New York', 'Paris', 'Tokyo')
colnames(datJan) <-seq(1981, 1987)

datJan #This is the 4-by-7 space-time data matrix (Units: deg C)
```

```
#              1981    1982   1983    1984    1985    1986    1987
#San Diego  0.9260 -0.7491 1.2609  1.1079 -0.8644  0.9502 -0.3180
#New York  -3.1299 -3.0661 1.5059 -1.7508 -1.5179  0.3121  0.2546
#Paris     -0.5337  1.1086 1.7997  0.5446 -5.3362  0.1774 -5.6329
#Tokyo     -1.1684 -0.3927 0.1208 -1.0764 -1.1135 -1.2139 -0.1822
```

2. [20 $points$] **R programming: Change a time-time data matrix to a data sequence, and then plot the data sequence**
(a) [10 points] The data file `EarthTemperatureData.csv` contains the monthly and annual global average temperature anomalies from 1880 to 2015. Extract the monthly data from this data matrix and convert the monthly data into a sequence, i.e., a vector, according to time order: Jan 1880, Feb 1880, Mar 1880, ... , Dec 1880, Jan 1881, Feb 1881, ..., Nov 2015, Dec 2015.
(b) [10 points] Plot the temperature data sequence from Jan 1880 to Dec 2015. The horizontal axis should be marked as Year from 1880 to 2015. The vertical axis is for the temperature anomaly data.
Solution:

```
#Problem 2: R programming: Change a time-time data matrix to
#a data sequence, and then plot the data sequence

#(a) Convert the matrix data into a sequence data
dat = read.csv('data/EarthTemperatureData.csv', header = TRUE)
dim(dat)
#[1] 166  14  It is a 166-by-14 matrix
dat[1,]
#  YEAR    JAN    FEB    MAR   APR    MAY    JUN    JUL    AUG    SEP    OCT
#1 1850 -0.702 -0.284 -0.732 -0.57 -0.325 -0.213 -0.128 -0.233 -0.444 -0.452
#NOV    DEC ANNUAL
#1 -0.19 -0.268 -0.375
dat[1:2, 1:5]
#  YEAR    JAN    FEB    MAR    APR
#1 1850 -0.702 -0.284 -0.732 -0.570
#2 1851 -0.303 -0.362 -0.485 -0.445

#The above shows that the monthly data are from columns 2 to 13.
#Column 1 is Year and 14 is Annual Data
mondat = dat[, 2:13]
mondat[1:2, 1:4]
#     JAN    FEB    MAR    APR
#1 -0.702 -0.284 -0.732 -0.570
#2 -0.303 -0.362 -0.485 -0.445
mondat[1, 8:12]
#     AUG    SEP    OCT   NOV    DEC
#1 -0.233 -0.444 -0.452 -0.19 -0.268

#Thus, mondat is the monthly data matrix
#Each row has 12 months
#Convert the matrix data into a vector or sequence
#We transpose the matrix first since the time goes
#according to row from Jan to Dec in a year
#Then we convert the transposed matrix into a vector
monseq = c(t(mondat))
monseq[1:24]
# [1] -0.702 -0.284 -0.732 -0.570 -0.325 -0.213 -0.128 -0.233 -0.444 -0.452
#[11] -0.190 -0.268 -0.303 -0.362 -0.485 -0.445 -0.302 -0.189 -0.215 -0.153
#[21] -0.108 -0.063 -0.030 -0.067
```

```
#Compare these data with the output of dat[1:2,]
dat[1:2,]
#YEAR    JAN    FEB    MAR    APR    MAY    JUN    JUL    AUG    SEP
#1 1850 -0.702 -0.284 -0.732 -0.570 -0.325 -0.213 -0.128 -0.233 -0.444
#2 1851 -0.303 -0.362 -0.485 -0.445 -0.302 -0.189 -0.215 -0.153 -0.108
#OCT   NOV    DEC ANNUAL
#1 -0.452 -0.19 -0.268 -0.375
#2 -0.063 -0.03 -0.067 -0.223

#This has verified that monseq is the monthly data sequence.

#(b) Plot the data sequence
time = seq(1880, 2015, len = length(monseq))
plot(time, monseq, type = 'l',
     xlab = 'Time', ylab = 'Temp anomalies [deg C]',
     main = 'Monthly Global Average Temp Anomalies')
```

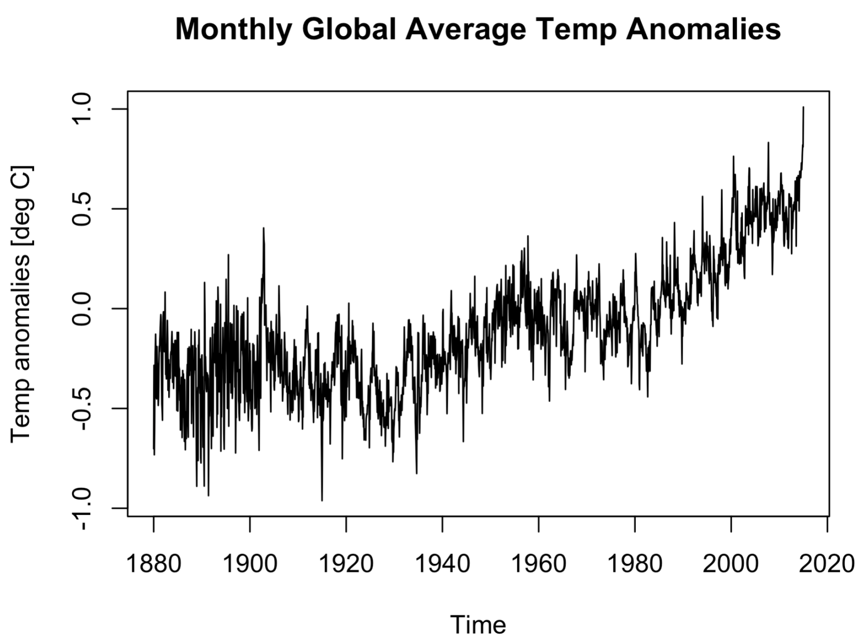

Figure 1 The time series of the monthly global average temperature anomalies.

3. [20 $points$] **SVD calculation**
The R command `svd(A)` gives the following results

```
svd(A)
$d
[1] 3 1

$u
           [,1]       [,2]
[1,] -0.7071068 -0.7071068
[2,] -0.7071068  0.7071068

$v
     [,1] [,2]
[1,]   -1    0
[2,]    0   -1
```

(a) [10 points] What are A's SVD matrices U, D and V?
(b) [10 points] Use hand-calculations for matrix multiplication: DV^t, then $U(DV^t)$ to approximately recover the original matrix A. Your final result should be a 2-by-2 matrix A.
Solution:
(a)

$$U = \begin{bmatrix} -\sqrt{2}/2 & -\sqrt{2}/2 \\ -\sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} \tag{0.1}$$

$$D = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \tag{0.2}$$

$$V = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \tag{0.3}$$

(b)

$$DV^t = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 0 & -1 \end{bmatrix} \tag{0.4}$$

$$U(DV^t) = \begin{bmatrix} -\sqrt{2}/2 & -\sqrt{2}/2 \\ -\sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} \begin{bmatrix} -3 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 3\sqrt{2}/2 & \sqrt{2}/2 \\ 3\sqrt{2}/2 & -\sqrt{2}/2 \end{bmatrix} \tag{0.5}$$

4. [20 $points$] **SVD theory**
If A is an m-by-n matrix with $m > n$, and if

$$A = UDV^t \tag{0.6}$$

is the SVD of A and

$$C = A^t A = QHQ^t \tag{0.7}$$

is the SVD of C, show that

$$D^2 = H \tag{0.8}$$

and

$$Q = V \tag{0.9}$$

assuming the SVD result for a matrix is unique.
Proof:

$$C = A^t A = (UDV^t)^t UDV^t = VDU^t UDV^t. \tag{0.10}$$

Since U is an orthogonal matrix, we have

$$U^t U = I \tag{0.11}$$

being an identity matrix.
Thus,

$$C = A^t A = (UDV^t)^t UDV^t = VDU^t UDV^t = VDIDV^t = V(D^2)V^t. \tag{0.12}$$

Comparing this with

$$C = A^t A = QHQ^t \tag{0.13}$$

and using the SVD's uniqueness assumption, we have

$$D^2 = H, \quad Q = V. \tag{0.14}$$

5. [20 $points$] **Concept of independent vectors**
(a) [8 points] Show that the three column vectors of the following matrix A are not linearly independent

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}. \tag{0.15}$$

(b) [4 points] Use R to compute the determinant of this matrix A. Does this matrix A have an inverse?
(c) [4 points] Use R to compute the inverse matrix of the following matrix B:

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 1 & 9 \end{bmatrix}. \tag{0.16}$$

(d) [4 points] Use R to compute the determinant of the matrix B.
Solution:
(a)
When looking at the columns, we see a pattern: The first column plus 1 is the second column, and plus 2 is the third column. This observation allows to try different kinds of relationships. One of them is that the first column is equal to 2 times the second column and then minus the third column, i.e.,

$$\mathbf{c}_1 = 2\mathbf{c}_2 - \mathbf{c}_3, \tag{0.17}$$

where **c** stands for a column vector, and the subscript indicates what column.

Thus, column 1 is a linear combination of columns 2 and 3. Thus, the column vectors are not linearly independent.

(b)

```
A = matrix(1:9, nrow = 3, byrow = TRUE)
A #Check if the matrix is correct
det(A) # Compute determinant of A
#[1] 6.661338e-16 This is considered zero
```

Thus, the determinant of A is zero.

Matrix A has no inverse since $\det(A) = 0$.

(c)

```
#(c)
B = matrix(c(1,2,3,4,5,6,7,1,9),
           nrow = 3, byrow = TRUE)
B #Check if the matrix is correct
solve(B)
#           [,1]       [,2]        [,3]
#[1,] -0.9285714  0.3571429  0.07142857
#[2,] -0.1428571  0.2857143 -0.14285714
#[3,]  0.7380952 -0.3095238  0.07142857
#This is the inverse matrix of B

#(d)
det(B)
#[1] -42 is the determinant of B and is not zero
```