Tutorial - Week 2

Please read the related material and attempt these questions before attending your allocated tutorial. Solutions are released on Friday 4pm.

Question 1

Consider the matrix

$$\begin{pmatrix} -1 & 3 & -2 \\ 2 & 4 & 2 \\ 5 & 2 & 3 \end{pmatrix}.$$

(a) Calculate the matrix of deviations (residuals), given by

$$\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}'$$

where $\bar{\mathbf{x}}$ is the mean vector of observations and $\mathbf{1}$ is a vector of 1's. Is this matrix of full rank? Explain.

Solution:

```
Ones = c(1,1,1)
xbar = apply(X, 2, mean)
R = X - Ones %*% t(xbar)
R
```

```
##      [,1] [,2] [,3]
## [1,]   -3    0   -3
## [2,]    0    1    1
## [3,]    3   -1    2
```

The *rank* of a matrix is the number of linearly independent rows or columns. The residual matrix is of *full rank* if the rank of `R` is equal to 3. Notice that we can obtain the third row `R[3,]` by

```
-1 * R[1,] - 1 * R[2,]
```

```
## [1]  3 -1  2
```

Therefore, the matrix `R` is not of full rank.
Numerically, it's quite easy to spot a matrix is not of full rank as the inverse cannot be computed.

```
solve(R)
```

```
## Error in solve.default(R): Lapack routine dgesv: system is exactly singular: U[3,3] = 0
```

(b) Determine the sample covariance matrix $\mathbf{S}$ and calculate the sample generalised variance $|\mathbf{S}|$. Interpret the latter geometrically.

Solution: The sample covariance $\mathbf{S}$ is calculated as:

```
ix = c(1,1,1) %*% t(apply(X, 2, mean))
S = t(X - ix) %*% (X - ix) / 2
S
```

```
##      [,1] [,2] [,3]
## [1,]  9.0 -1.5  7.5
## [2,] -1.5  1.0 -0.5
```

```
## [3,]  7.5 -0.5  7.0
```

Alternatively, we obtain the same result using the inbuilt R function:

```
S = var(X)
S
```

```
##      [,1] [,2] [,3]
## [1,]  9.0 -1.5  7.5
## [2,] -1.5  1.0 -0.5
## [3,]  7.5 -0.5  7.0
```

The *sample generalised variance* $|\mathbf{S}|$ is given by:

```
det(S)
```

```
## [1] 5.995204e-15
```

Notice it is a single number (not a matrix).
A geometric interpretation of the (sample) generalised variance is that it is the volume of the parallelpiped defined by the p deviation vectors.

(c) Calculate the *sample total variance* which is given by the sum of the diagonal elements of the sample covariance matrix $\mathbf{S}$. The total variance is another measure of variance (i.e., a number to describe the covariance of the data).

Solution: The diagonal of a matrix can be extracted using the `diag` function.

```
diag(S)
```

```
## [1] 9 1 7
```

Combined with the `sum` function we can calculate very easily the sample total variance.

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

```
## [1] 17
```

Question 2

Consider the matrices

$$A = \begin{pmatrix} 4.000 & 4.001 \\ 4.001 & 4.002 \end{pmatrix}, \qquad B = \begin{pmatrix} 4 & 4.001 \\ 4.001 & 4.002001 \end{pmatrix}.$$

These matrices are identical except for a small difference in the (2,2) position. Moreover, the columns of $\mathbf{A}$ (and $\mathbf{B}$) are nearly linearly dependent. Show that

$$\mathbf{A}^{-1} = (-3)\mathbf{B}^{-1}.$$

Consequently, small changes – perhaps caused by rounding – can give substantially different inverses.

Solution: Using the formula for matrix inverse for 2x2 matrix:

$$\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \Longrightarrow \mathbf{A}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

We apply this to matrices **A** and **B**

$$
\begin{aligned}
\mathbf{A} &= \begin{pmatrix} 4 & 4.001 \\ 4.001 & 4.002 \end{pmatrix} \\
\implies \mathbf{A}^{-1} &= \frac{1}{16.008 - 16.008001} \begin{pmatrix} 4.002 & -4.001 \\ -4.001 & 4 \end{pmatrix} \\
&= \frac{1}{-0.000001} \begin{pmatrix} 4.002 & -4.001 \\ -4.001 & 4 \end{pmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{B} &= \begin{pmatrix} 4 & 4.001 \\ 4.001 & 4.002001 \end{pmatrix} \\
\implies \mathbf{B}^{-1} &= \frac{1}{16.008004 - 16.008001} \begin{pmatrix} 4.002001 & -4.001 \\ -4.001 & 4 \end{pmatrix} \\
&= \frac{1}{+0.000003} \begin{pmatrix} 4.002001 & -4.001 \\ -4.001 & 4 \end{pmatrix}
\end{aligned}
$$

So we find that a small difference between **A** and **B** led to $\approx -3\times$ difference in the inverse.
The inverse of a matrix in R can be found using the `solve` function. Notice how the inverses of A and B are very different:

```
solve(A)
```

```
##          [,1]     [,2]
## [1,] -4002000  4001000
## [2,]  4001000 -4000000
```

```
solve(B)
```

```
##          [,1]     [,2]
## [1,]  1334000 -1333667
## [2,] -1333667  1333333
```

Question 3

In this question, we consider the Fashion MNIST dataset and use it as an example data set for calculating the sample total variation. Please refer to Workshop 1 for examples of how to load and manipulate the Fashion MNIST dataset. Details on the Fashion MNIST dataset is found here: https://github.com/zalandoresearch/fashion-mnist.

(a) Turn each (image) 28×28-sized observation of Fashion MNIST data set into a vector observation of size $p = 784$.

Solution:

```
source('read_idx.r')
```

We download the (smaller) data set and decompress it.

```
URL = "http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/"
fn = "t10k-images-idx3-ubyte.gz"
download.file(paste0(URL, fn), fn)
gunzip(fn, overwrite=TRUE)
```

We load the image data into the variable x.

```r
X = read_idx(gsub(pattern = "\\.gz", "", fn))
```

The dimensionality is 10000 observations, 28 rows, 28 columns.

```r
dim(X)
```

```
## [1] 10000    28    28
```

We can force it into vectors by doing:

```r
dim(X) = c(10000, 28*28)
```

Checking:

```r
dim(X)
```

```
## [1] 10000   784
```

(b) Download the Fashion MNIST labels and use them to calculate the sample total variation of each clothing class.

Solution: We download the labels and store them in `y`

```r
URL = "http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/"
fn = "t10k-labels-idx1-ubyte.gz"
download.file(paste0(URL, fn), fn)
gunzip(fn, overwrite=TRUE)
# Read labels
y = read_idx(gsub(pattern = "\\.gz", "", fn))
```

The vector `y` contains the values of the labels. It has length 10,000:

```r
length(y)
```

```
## [1] 10000
```

The unique class values range from 0 to 9.

```r
unique(y)
```

```
##  [1] 9 2 1 6 4 5 7 3 8 0
```

We can extract the observations related to class 0 with `X[y == 0,]`, we get a vector of dimension 1000 x 784.

```r
dim(X[y == 0,])
```

```
## [1] 1000  784
```

For example, the mean vector of class 0 is given by:

```r
xbar = colMeans(X[y == 0,])
```

It has the correct length:

```r
length(xbar)
```

```
## [1] 784
```

We now construct a function to calculate the generalised variance:

```r
TV = function(x) sum(diag(var(x)))
```

We can test it:

```r
TV(X[y==0,])
```

```
## [1] 2657619
```

We calculate the total variation for each clothing class.

```r
sapply(1:9, function(i) TV(X[y==i,]))
```

```
## [1] 1683904 3038923 2397784 2802977 2596807 3182151 1584657 4044971 2700858
```