MODUL 1

REVIEW ALJABAR MATRIKS dan GEOMETRI SAMPEL

1.1 Deskripsi Singkat

Aljabar adalah suatu cabang ilmu matematika yang menggunakan symbol, biasanya berupa huruf, untuk merepresentasikan bilangan atau anggota suatu himpunan tertentu dan digunakan untuk menyatakan kuantitas dan mengekspresikan hubungan antar anggota himpunan tersebut.

Matriks adalah suatu array persegi yang berisi suatu kuantitas aljabar atau numerik terkait suatu operasi matematika.

Jadi aljabar matriks merupakan Aljabar yang setiap simbol yang terlibat mewakili bukan satu angka saja , melainkan suatu matriks (atau vector).

Data multivariat merupakan data yang terdiri dari n baris dan p kolom. Data multivariat bisa direpresentasikan dalam dua cara. Cara pertama data multivariat dipandang sebagai himpunan vector baris sehingga data tersebut bisa direpresentasikan secara grafis dalam bentuk scatter plot. Sedangkan cara kedua data multivariat dipandang sebagai himpunan vector kolom, sehingga bisa direpresentasikan dalam bentuk vector di dalam ruang n dimensi. Kedua cara tersebut masing-masing memiliki interpretasi geometris terkait keragaman data.

1.2 Tujuan Praktikum

Tujuan pembelajaran pada modul 2 ini adalah mahasiswa mampu menjelaskan konsep aljabar matriks dalam metode analisis peubah ganda serta memahami geometri sampel dan *random sampling* untuk memaknai nilai tengah dan keragaman data peubah ganda dan mempraktekkannya dengan menggunakan R.

1.3 Material Praktikum

Pada kegiatan modul 1 diperlukan beberapa material berupa software, yaitu:

1) Software R (direkomendasikan versi terbaru).
2) Software RStudio.

1.4 Kegiatan Praktikum

1. Aljabar Matriks dengan R

A*B	element-wise multiplication
A%*%B	matrix multiplication
crossprod(A,B)	= A'B → cross product
t(A)	transpose of matrix
solve(A)	inverse of matrix
det(A)	determinant of matrix
diag(x)	create diagonal matrix from vector x
diag(A)	Returns a vector containing the elements of the principal diagonal
eigen(A)	returns eigen value and eigen vector of A
svd(A)	single value decomposition of A

Contoh:

```
> A <- matrix(c(3,5,2,6,4,8,7,9,5), nrow=3)
> B <- matrix(c(1,3,2,5,7,5,8,3,4), nrow = 3)
>
> #element-wise multiplication
> A*B
     [,1] [,2] [,3]
[1,]    3   30   56
[2,]   15   28   27
[3,]    4   40   20
>
> #matrix multiplication
> A%*%B
     [,1] [,2] [,3]
[1,]   35   92   70
[2,]   35   98   88
[3,]   36   91   60
>
> #cross product
> crossprod(A,B)
     [,1] [,2] [,3]
[1,]   22   60   47
[2,]   34   98   92
[3,]   44  123  103
>
> #transpose of matrix
> t(A)
     [,1] [,2] [,3]
[1,]    3    5    2
[2,]    6    4    8
[3,]    7    9    5
```

```
> t(B)
     [,1] [,2] [,3]
[1,]    1    3    2
[2,]    5    7    5
[3,]    8    3    4
>
> #inverse matrix
> solve(A)
           [,1]        [,2]       [,3]
[1,] -2.0000000  1.00000000  1.0000000
[2,] -0.2692308  0.03846154  0.3076923
[3,]  1.2307692 -0.46153846 -0.6923077
> solve(B)
           [,1]       [,2]       [,3]
[1,] -1.4444444 -2.2222222  4.5555556
[2,]  0.6666667  1.3333333 -2.3333333
[3,] -0.1111111 -0.5555556  0.8888889
>
> #Determinant matrix
> det(A)
[1] 26
> det(B)
[1] -9
>
> #identify diagonal of matrix
> diag(A)
[1] 3 4 5
> diag(B)
[1] 1 7 4
>
> #create diagonal matrix
> diag(c(2,4,3),3,3)  #diagonal matrix size 3x3
     [,1] [,2] [,3]
[1,]    2    0    0
[2,]    0    4    0
[3,]    0    0    3
> diag(2) #create identity matrix sized 2x2
     [,1] [,2]
[1,]    1    0
[2,]    0    1
>
> #eigen value square matrix
> eigen(A)
eigen() decomposition
$values
[1] 16.3243746 -3.9178477 -0.4065269

$vectors
           [,1]        [,2]       [,3]
[1,] -0.5647567 -0.03865392 -0.8292486
[2,] -0.6237354 -0.73949409 -0.1554778
[3,] -0.5403739  0.67205235  0.5368178

> eigen(B)
eigen() decomposition
$values
[1] 12.5023175 -1.1360018  0.6336843
```

```
$vectors
           [,1]        [,2]       [,3]
[1,] -0.6140401 -0.94255286  0.8083083
[2,] -0.6087174  0.33104496 -0.5151917
[3,] -0.5024121  0.04475873  0.2849828
```

2. Geometri Sampel

Diberikan data sebagai berikut dimana tiga variable diukur dalam satuan milliequivalent per 100gr pada 10 lokasi berbeda, dimana:

y1 = available soil calcium

y2 = exchangeable soil calcium

y3 = turnip green calcium

Table 3.3. Calcium in Soil and Turnip Greens

Location Number	y_1	y_2	y_3
1	35	3.5	2.80
2	35	4.9	2.70
3	40	30.0	4.38
4	10	2.8	3.21
5	6	2.7	2.73
6	20	2.8	2.81
7	35	4.6	2.88
8	35	10.9	2.90
9	35	8.0	3.28
10	30	1.6	3.20

a) Hitung matriks kovarians **S** dari matriks data Y!

b) Hitung matriks korelasi **R**!

c) Hitung generalized variance |**S**|!

Dengan menggunakan R:

a) Dengan menggunakan script R berikut:

```
y1<-c(35,35,40,10,6,20,35,35,35,30)
y2<-c(3.5,4.9,30,2.8,2.7,2.8,4.6,10.9,8,1.6)
```

Input data

```
y3<-c(2.8,2.7,4.38,3.21,2.73,2.81,2.88,2.9,3.28,3.2)


Ydat<-data.frame(y1,y2,y3)     #convert data menjadi data frame

ybar<-apply(Ydat,2,mean)       #hitung mean setiap variabel

d1<-y1-ybar[1]

d2<-y2-ybar[2]

d3<-y3-ybar[3]
```

Hitung jarak

```
dmat<-matrix(c(d1,d2,d3),ncol=3)


#Atau matriks jarak juga bisa kita peroleh sbb:

dmat<-Ydat-matrix(rep(ybar,10),nrow = 10,byrow = TRUE)


#Maka matriks kovarians Sn

Sn<-(1/10)*t(dmat)%*%dmat


print(dmat)

print(Sn)
```

Dari data dihitung vektor deviasi untuk masing-masing variabel sebagai berikut:

$$d_i = (y_i - \bar{x}_i\mathbf{1})$$

diperoleh hasil sebagai berikut (output R):

```
       [,1]  [,2]   [,3]
 [1,]   6.9 -3.68 -0.289
 [2,]   6.9 -2.28 -0.389
 [3,]  11.9 22.82  1.291
 [4,] -18.1 -4.38  0.121
 [5,] -22.1 -4.48 -0.359
 [6,]  -8.1 -4.38 -0.279
 [7,]   6.9 -2.58 -0.209
 [8,]   6.9  3.72 -0.189
 [9,]   6.9  0.82  0.191
[10,]   1.9 -5.58  0.111
```

Kemudian dari vektor deviasi dihitung matriks kovarians $\mathbf{S}_n$

$$\boldsymbol{S_n} = \frac{1}{n}\boldsymbol{d_i}'\boldsymbol{d_k}$$

diperoleh hasil berikut (output R):

```
         [,1]     [,2]     [,3]
[1,] 126.4900 44.71200 1.747100
[2,]  44.7120 65.02360 3.308480
[3,]   1.7471  3.30848 0.225109
```

Untuk unbiased estimator matriks kovarians $\mathbf{\Sigma}$ adalah

$$\boldsymbol{S_{n-1}} = \frac{n}{n-1}\boldsymbol{S_n} = \frac{10}{10-1}\boldsymbol{S_n}$$

Syntax :

```
S <- (10/9)*Sn
```

diperoleh:

```
             [,1]      [,2]      [,3]
[1,] 140.544444 49.680000 1.9412222
[2,]  49.680000 72.248444 3.6760889
[3,]   1.941222  3.676089 0.2501211
```

Catatan:

- Selanjutnya unbiased estimator $\mathbf{S}_{n-1}$ cukup ditulis dengan **S**

- Matriks kovarians **S** di atas dalam R bisa diperoleh dengan *command* : cov(Ydat)

b) Matriks korelasi. Syntax R (melanjutkan dari a):

```
#nomor b

#define Ds

Ds<-diag(c(sqrt(s11),sqrt(s22),sqrt(s33)))

#inverse matriks Ds

invDs<-solve(Ds)

#Hitung matriks korelasi R

Rmat<-invDs%*%S%*%invDs

print(Rmat)
```

Output:

```
          [,1]      [,2]     [,3]

[1,] 1.0000000 0.4930154 0.327411

[2,] 0.4930154 1.0000000 0.864762

[3,] 0.3274110 0.8647620 1.000000
```

c) Generalized variance adalah determinan dari matriks kovarian (unbiased):

```
GenVar<-det(Sunb)
```

Diperoleh generalized variance |**S**| = 459.9555

1.5 Responsi

1. Apakah matriks berikut positif definif? Periksalah!

$$\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

2. Periksalah apakah matriks berikut

$$\mathbf{Q} = \begin{bmatrix} \frac{5}{13} & \frac{12}{13} \\ -\frac{12}{13} & \frac{5}{13} \end{bmatrix}$$

adalah matriks orthogonal atau bukan.

1.6 Tugas Terstruktur

Untuk Program Studi Komputasi Statistik:

1. Buatlah fungsi R untuk menghitung:
    a. Proyeksi vector **x** terhadap vector **y** dan panjang vektornya
    b. Matriks kovarians **S** yang dihitung dari vector deviasi
2. Kerjakan Exercise 3.4 pada buku Johnson edisi 5, halaman 144!

Untuk Program Studi Statistika:

1. Kerjakan Exercise 3.1 dan 3.4 pada buku Johnson edisi 5, halaman 144!