

Laporan Tugas Besar 1 IF 2123 Aljabar Linier dan Geometri

Sistem Persamaan Linier, Determinan, dan Aplikasinya



Kelompok 32 (Grezi)

Muhammad Equilibrie Fajria - 13521047

Muhammad Farrel Danendra Rachim - 13521048

Ezra Maringan Christian Mastra Hutagaol - 13521073

Bab 1

Deskripsi Masalah

- A. Buatlah pustaka dalam Bahasa Java untuk menemukan solusi SPL dengan metode eliminasi Gauss, metode Eliminasi Gauss-Jordan, metode matriks balikan, dan kaidah Cramer (kaidah Cramer khusus untuk SPL dengan n peubah dan n persamaan), menghitung determinan matriks dengan reduksi baris dan dengan ekspansi kofaktor, dan menghitung balikan matriks.
- B. Gunakan pustaka di atas untuk membuat program penyelesaian berbagai persoalan dalam bentuk SPL, menyelesaikan persoalan interpolasi dan regresi linier, menghitung matriks balikan, menghitung determinan matriks dengan berbagai metode (reduksi baris dan ekspansi kofaktor).

Bab 2

Teori Singkat

1. Eliminasi Gauss

Metode ini ditemukan oleh matematikawan bernama Carl Friedrich Gauss. Eliminasi Gauss dapat digunakan untuk mencari solusi dalam sebuah sistem persamaan linear (SPL). Caranya yaitu dengan merepresentasikan sistem persamaan tersebut menjadi sebuah matriks augmented, mengubah matriks tersebut menjadi matriks eselon baris dengan melakukan operasi baris elementer (OBE), dan mencari solusi dengan melakukan substitusi balik (solusi bisa dalam bentuk parametrik).

Berikut contoh sebuah sistem persamaan linear:

$$\begin{aligned}x_1 - x_2 + 2x_3 &= 5 \\2x_1 - 2x_2 + 4x_3 &= 10 \\3x_1 - x_2 + 6x_3 &= 15\end{aligned}$$

Dari sistem tersebut, buat matriks augmented-nya seperti berikut:

$$\left[\begin{array}{ccc|c} 1 & -1 & 2 & 5 \\ 2 & -2 & 4 & 10 \\ 3 & -1 & 6 & 15 \end{array} \right]$$

Dengan melakukan OBE, diperoleh matriks eselon baris:

$$\left[\begin{array}{ccc|c} 1 & -1 & 2 & 5 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array} \right]$$

Diperoleh persamaan:

$$\begin{aligned}x_2 &= 0 \\x_1 - x_2 + 2x_3 &= 5\end{aligned}$$

Dengan melakukan substitusi mundur, dan misal $x_3 = r$, maka diperoleh persamaan parametrik:

$$\text{Solusi: } x_1 = 5 - 2r, x_2 = 0, x_3 = r; \quad r \in \mathbb{R}$$

2. Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan mirip seperti eliminasi Gauss, namun perbedaannya terletak pada pengubahan matriks augmented dari SPL menggunakan OBE menjadi matriks eselon baris tereduksi, sehingga substitusi mundur tidak diperlukan lagi.

Misal terdapat persamaan yang diubah menjadi matriks augmented seperti berikut::

$$\begin{array}{rrcrcl} 2x_1 + 2x_2 - x_3 & & & + x_5 = 0 \\ -x_1 - x_2 + 2x_3 - 3x_4 + x_5 & = 0 \\ x_1 + x_2 - 2x_3 & & & - x_5 = 0 \\ & x_3 + x_4 + x_5 = 0 \end{array} \quad \left[\begin{array}{cccccc} 2 & 2 & -1 & 0 & 1 & 0 \\ -1 & -1 & 2 & -3 & 1 & 0 \\ 1 & 1 & -2 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{array} \right]$$

Lalu dengan OBE, diperoleh matriks eselon baris tereduksi dengan persamaan:

$$\left[\begin{array}{cccccc} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right] \quad \begin{array}{l} x_1 + x_2 \quad \quad \quad + x_5 = 0 \\ \quad \quad \quad x_3 \quad \quad \quad + x_5 = 0 \\ \quad \quad \quad \quad \quad x_4 \quad \quad \quad = 0 \end{array}$$

Sehingga persamaan parametriknya adalah:

$$\boxed{x_1 = -s - t, \quad x_2 = s, \quad x_3 = -t, \quad x_4 = 0, \quad x_5 = t}$$

3. Determinan

Determinan merupakan sebuah fungsi yang mengasosiasi bilangan riil dengan sebuah matriks persegi. Determinan dapat ditentukan dengan dua buah metode.

- Reduksi baris. OBE diterapkan pada suatu matriks persegi sehingga diperoleh matriks segitiga atas atau bawah. Determinan dapat ditemukan dengan mengalikan setiap elemen pada diagonal utama. Tanda positif atau negatif bergantung kepada berapa banyak penukaran baris yang dilakukan pada proses OBE
- Eksansi kofaktor. Determinan suatu matriks merupakan jumlah perkalian tiap elemen matriks dengan kofaktornya dalam sebuah baris atau kolom.

Misal terdapat matriks berikut:

$$A = \begin{bmatrix} 3 & 5 & -2 & 6 \\ 1 & 2 & -1 & 1 \\ 2 & 4 & 1 & 3 \\ 3 & 7 & 5 & 3 \end{bmatrix}$$

Maka:

$$\det(A) = a_{11}C_{11} + a_{12}C_{12} + a_{13}C_{13} + a_{14}C_{14}$$

$$\det(A) = 3 \begin{vmatrix} 2 & -1 & 1 \\ 4 & 1 & 3 \\ 7 & 5 & 3 \end{vmatrix} - 5 \begin{vmatrix} 1 & -1 & 1 \\ 2 & 1 & 3 \\ 3 & 5 & 3 \end{vmatrix} + (-2) \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 3 \\ 3 & 7 & 3 \end{vmatrix} - 6 \begin{vmatrix} 1 & 2 & -1 \\ 2 & 4 & 1 \\ 3 & 7 & 5 \end{vmatrix}$$

4. Matriks balikan

Matriks balikan atau matriks invers merupakan sebuah matriks yang dilambangkan dengan A^{-1} dari sebuah matriks persegi A sehingga $A^{-1}A = AA^{-1} = I$, di mana I merupakan matriks identitas. Terdapat dua buah metode untuk memperoleh matriks balikan:

a. Eliminasi Gauss-Jordan

$$[A|I] \sim [I|A^{-1}]$$

Dari matriks augmented di ruas yang kiri, dapat dilihat matriks A di sisi kiri dan matriks identitas I di sisi kanan. Tujuan eliminasi ini adalah untuk membuat sisi kiri mengandung matriks I, sehingga diperoleh matriks A^{-1} di sisi kanan.

Misal dikasih matriks A:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 3 \\ 1 & 0 & 8 \end{bmatrix}$$

Dilakukan eliminasi dengan OBE yang memenuhi syarat di atas:

$$\left(\begin{array}{ccc|ccc} 1 & 2 & 3 & 1 & 0 & 0 \\ 2 & 5 & 3 & 0 & 1 & 0 \\ 1 & 0 & 8 & 0 & 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & -40 & 16 & 9 \\ 0 & 1 & 0 & 13 & -5 & -3 \\ 0 & 0 & 1 & 5 & -2 & -1 \end{array} \right)$$

Jadi matriks invers A:

$$A^{-1} = \begin{bmatrix} -40 & 16 & 9 \\ 13 & -5 & -3 \\ 5 & -2 & -1 \end{bmatrix}$$

b. Matriks adjoin

Balikan matriks A juga dapat ditemukan dengan mencari determinan dan matriks adjoin A (akan dijelaskan di poin no. 6) lalu menerapkannya ke rumus:

$$A^{-1} = \frac{1}{\det(A)} \text{adj}(A)$$

5. Matriks kofaktor

Misalkan A adalah matriks persegi n x n dan C_{ij} adalah kofaktor dari a_{ij} , maka matriks kofaktor A adalah

$$\begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix}$$

Adapun kofaktor $C_{ij} = (-1)^{i+j} M_{ij}$ di mana M_{ij} merupakan determinan submatriks yang elemen-elemennya tidak berada di baris i dan kolom j.

Misal terdapat matriks:

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix}$$

sehingga masing-masing kofaktornya dan matriks kofaktornya:

$$\begin{array}{lll} C_{11} = 12 & C_{12} = 6 & C_{13} = -16 \\ C_{21} = 4 & C_{22} = 2 & C_{23} = 16 \\ C_{31} = 12 & C_{32} = -10 & C_{33} = 16 \end{array} \quad \begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}$$

6. Matriks adjoin

Matriks adjoin adalah transpos dari matriks kofaktor. Mengambil contoh dari poin no. 5, berikut matriks kofaktornya:

$$\begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}$$

sehingga matriks adjoinnya adalah:

$$\text{adj}(A) = \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & 16 & 16 \end{bmatrix}$$

7. Kaidah Cramer

Jika $Ax = b$ adalah SPL yang terdiri dari n persamaan linier dengan n variabel sehingga $\det(A) \neq 0$, SPL tersebut memiliki solusi:

$$x_1 = \frac{\det(A_1)}{\det(A)}, \quad x_2 = \frac{\det(A_2)}{\det(A)}, \dots, \quad x_n = \frac{\det(A_n)}{\det(A)}$$

A_j adalah matriks yang diperoleh dengan mengganti entri pada kolom ke- j dari A dengan entri dari matriks

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

Misal, berikut matriks A dan \mathbf{b} :

$$A = \begin{bmatrix} -1 & 2 & -3 \\ 2 & 0 & 1 \\ 3 & -4 & 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

maka matriks A_j :

$$A_1 = \begin{bmatrix} \mathbf{1} & 2 & -3 \\ \mathbf{0} & 0 & 1 \\ \mathbf{2} & -4 & 4 \end{bmatrix} \quad A_2 = \begin{bmatrix} -1 & \mathbf{1} & -3 \\ 2 & \mathbf{0} & 1 \\ 3 & \mathbf{2} & 4 \end{bmatrix} \quad A_3 = \begin{bmatrix} -1 & 2 & \mathbf{1} \\ 2 & 0 & \mathbf{0} \\ 3 & -4 & \mathbf{2} \end{bmatrix}$$

8. Interpolasi polinom

Interpolasi polinomial adalah teknik interpolasi dengan mengasumsikan pola data yang dimiliki mengikuti pola polinomial baik berderajat satu (linier) maupun berderajat tinggi. Secara umum, melalui $n + 1$ pasangan titik berlainan $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ dalam arti $x_i \neq x_j$ untuk $i \neq j$, selalu dapat didefinisikan $P_n(x)$ polinomial derajat n yang melalui titik-titik tersebut, yaitu $P_n(x_k) = y_k$, $k = 0, 1, \dots, n$. Dengan menyulihkan (x_i, y_i) ke dalam persamaan polinom

$$p_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

untuk $i = 0, 1, 2, \dots, n$, akan diperoleh n buah sistem persamaan linier dalam $a_0, a_1, a_2, \dots, a_n$,

$$a_0 + a_1x_0 + a_2x_0^2 + \dots + a_nx_0^n = y_0$$

$$a_0 + a_1x_1 + a_2x_1^2 + \dots + a_nx_1^n = y_1$$

...

...

$$a_0 + a_1x_n + a_2x_n^2 + \dots + a_nx_n^n = y_n$$

Solusi sistem persamaan linier diperoleh dengan menggunakan metode eliminasi Gauss.

9. Interpolasi bicubic

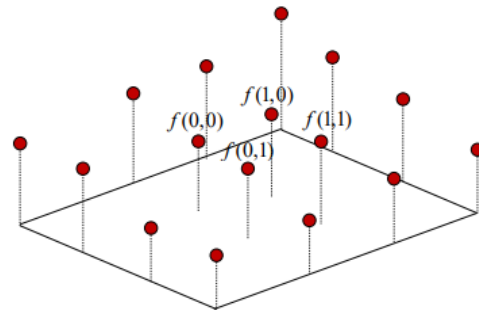
Interpolasi bicubic adalah teknik interpolasi pada data 2D umumnya digunakan dalam pembesaran citra yang merupakan pengembangan dari interpolasi linier dan cubic. Akan dicari persamaan interpolasi $f(x, y)$ dengan pemodelan sebagai berikut:

Normalization: $f(0,0), f(1,0)$

$f(0,1), f(1,1)$

$$\text{Model: } f(x, y) = \sum_{j=0}^3 \sum_{i=0}^3 a_{ij} x^i y^j$$

$$x = -1, 0, 1, 2$$



Solve: a_{ij}

Dilakukan substitusi pada matriks 4 x 4 ke persamaan $f(x, y)$, sehingga diperoleh persamaan:

$$y = Xa$$

$$\begin{bmatrix} f(-1,-1) \\ f(0,-1) \\ f(1,-1) \\ f(2,-1) \\ f(-1,0) \\ f(0,0) \\ f(1,0) \\ f(2,0) \\ f(-1,1) \\ f(0,1) \\ f(1,1) \\ f(2,1) \\ f(-1,2) \\ f(0,2) \\ f(1,2) \\ f(2,2) \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 2 & 4 & 8 & -1 & -2 & -4 & -8 & 1 & 2 & 4 & 8 & -1 & -2 & -4 & -8 \\ 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 \\ 1 & -1 & 1 & -1 & 2 & -2 & 2 & -2 & 4 & -4 & 4 & -4 & 8 & -8 & 8 & -8 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 8 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 4 & 4 & 4 & 4 & 8 & 8 & 8 & 8 \\ 1 & 2 & 4 & 8 & 2 & 4 & 8 & 16 & 4 & 8 & 16 & 32 & 8 & 16 & 32 & 64 \end{bmatrix} \begin{bmatrix} a_{00} \\ a_{10} \\ a_{20} \\ a_{30} \\ a_{01} \\ a_{11} \\ a_{21} \\ a_{31} \\ a_{02} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{03} \\ a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}$$

Vektor \mathbf{a} dapat dicari dari persamaan tersebut menggunakan inverse, dan vektor \mathbf{a} dapat digunakan sebagai nilai variabel dalam $f(x, y)$, sehingga terbentuk fungsi interpolasi bicubic. Dalam laporan ini akan ditentukan persamaan $f(x, y)$ lalu melakukan interpolasi berdasarkan $f(a, b)$ dari masukan matriks 4 x 4, dengan nilai input a dan b dalam rentang $[0, 1]$.

10. Regresi linier berganda

Regresi linier merupakan salah satu metode untuk memprediksi nilai selain menggunakan interpolasi polinom dengan banyak variabel. Terdapat rumus umum yang bisa digunakan untuk regresi linear berganda:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + \epsilon_i$$

Untuk mendapatkan nilai dari setiap β_i dapat digunakan *normal estimation equation for multiple linear regression*:

$$\begin{array}{ccccccc} nb_0 + b_1 \sum_{i=1}^n x_{1i} & + & b_2 \sum_{i=1}^n x_{2i} & + & \cdots & + & b_k \sum_{i=1}^n x_{ki} & = & \sum_{i=1}^n y_i \\ b_0 \sum_{i=1}^n x_{1i} + b_1 \sum_{i=1}^n x_{1i}^2 & + & b_2 \sum_{i=1}^n x_{1i}x_{2i} & + & \cdots & + & b_k \sum_{i=1}^n x_{1i}x_{ki} & = & \sum_{i=1}^n x_{1i}y_i \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ b_0 \sum_{i=1}^n x_{ki} + b_1 \sum_{i=1}^n x_{ki}x_{1i} & + & b_2 \sum_{i=1}^n x_{ki}x_{2i} & + & \cdots & + & b_k \sum_{i=1}^n x_{ki}^2 & = & \sum_{i=1}^n x_{ki}y_i \end{array}$$

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

Bab 3

Implementasi Pustaka dan Program

1. Class App Methods

Nama	Tipe	Parameter	Deskripsi
main	Public void	String[] args	Mengeluarkan "Hello world!"

2. Class DeterminantRowReduction Methods

Nama	Tipe	Parameter	Deskripsi
detRowReduction	Public double	double [] [] matrix	Menghitung nilai determinan m menggunakan metode reduksi baris

3. Class EkspansiKofaktor Methods

Nama	Tipe	Parameter	Deskripsi
makeNewMatrix	Protected double [] []	Double [] [] matrix Int i Int j	Membuat matriks baru tanpa elemen pada baris i dan kolom j
detKofaktor	Public double	Double [] [] matrix	Menghitung nilai determinan m menggunakan ekspansi kofaktor

4.Class GaussJordanMethod Methods

Nama	Tipe	Parameter	Deskripsi
gaussJordan	Public void	Double [] [] matrix	Melakukan eliminasi Gauss Jordan

5.Class GaussMethod Methods

Nama	Tipe	Parameter	Deskripsi
gauss	Public void	Double [] [] matrix	Melakukan eliminasi gauss

6.Class Inverse Methods

Nama	Tipe	Parameter	Deskripsi
getKofaktor	Protected double	Double [] [] matrix Int i Int j	Mendapatkan nilai dari matriks kofaktor
inverse	Public double [] []	Double [] [] matrix	Melakukan operasi invers pada matriks

7.Class Cramer Methods

Nama	Tipe	Parameter	Deskripsi
kaidahCramer	Public string []	Double matrix [] []	Mengembalikan solusi sistem persamaan linier dari suatu matriks dengan kaidah cramer

8. Class filemethods

Methods

Nama	Tipe	Parameter	Deskripsi
readMatrixFile	Public double [] []	Int row Int col String file	Membaca matriks dari suatu file dan mengembalikan matriks tersebut
banyakRowandCol	Public int []	String file	Menghitung banyak baris dan kolom pada matriks dari file yang dimasukkan
writeMatrixkeFile	Public boolean	String directory Double [] [] matrix	Mengembalikan true jika berhasil menulis matriks ke file dan mengembalikan false jika gagal
writeMatrixDeterminanfile	Public boolean	String directory Double [] [] matrix Double det	Mengembalikan true jika berhasil menulis matriks dan determinannya ke file dan mengembalikan false jika gagal
writeMatrixInverse keFile	Public boolean	String directory Double [] [] matrix Double [] [] inverse Boolean hasInverse	Mengembalikan true jika berhasil menulis matriks dan inversnya ke file dan mengembalikan false jika gagal

9. Class matrixmethods

Methods

Nama	Tipe	Parameter	Deskripsi
Print	Public void	String str	Menulis string tanpa new line

Println	Public void	-	Menulis new line
printlnstr	Public void	String str	Menulis string dengan new line
printlnDou	Public void	Double dou	Menulis string bertipe double dengan new line
printf	Public void	String str Object . . . args	Menulis string dengan beberapa format argumen
displaymatrix	Public void	Double [] [] matrix	Menampilkan matrix
isSquare	Public boolean	Double [] [] matrix	Mengembalikan nilai true jika banyak baris sama dengan banyak kolom, false jika banyak baris tidak sama dengan banyak kolom
transpose	Public double [] []	Double [] [] matrix	Mengembalikan transpose dari suatu matriks
multiplyByconst	Public double [] []	Double [] [] matrix Double x	Mengembalikan matriks yang setiap elemennya dikali dengan suatu konstanta
multiplyMatrix	Public double [] []	Double [] [] matrix1 Double [] [] matrix2	Mengembalikan matriks baru hasil perkalian dari 2 matriks
isZero	Public boolean	Double x	Mengembalikan true jika x sama dengan 0 dan false jika sebaliknya
isRowElmtZero	Public boolean	Double [] row	Mengembalikan true jika semua elemen dalam baris tersebut bernilai 0 semua

			dan false jika ada elemen yang tidak 0
doesMatrixHaveNoSolution	Public boolean	Double [] [] matrix	Mengembalikan true jika terdapat suatu baris yang elemennya 0 dan hasilnya bukan 0, dan mengembalikan false jika tidak ada
cutAugmentedToSquare	Public double [] []	Double matrix [] []	Mengembalikan matriks persegi dari suatu matriks augmented yang banyak barisnya sama dengan banyak kolom dikurangi 1
getElementDiagonal	Public double	Double matrix [] [] Int i	Mengirimkan elemen m(i,i)
checkRowZeroUntilLast	Public boolean	Double[] row	Mengembalikan true jika sebuah row memiliki semua elemennya 0 kecuali elemen terakhir
minus2barismatrix	Public void	Double [] [] matrix Int row1 Int row2 Double n	Mengurangkan row1 pada matriks dengan n*row2
searchNonZero	Public double	Double [] [] matrix Int row	Mengembalikan elemen bukan 0 pertama di rowm, jika tidak ada maka mengembalikan 0
multiplyRow	Public void	Double [] [] matrix Int i Double n	Mengalikan baris dengan konstanta n
switchRow	Public void	Double [] [] matrix	Menukar row 1

		Int row1 Int row2	dengan row 2
searchIndex	Public int	Double [] [] matrix Int row Double n	Mencari indeks kolom dari elemen n pada row
createMatrix	Public void	Double [] [] newMatrix Int m Int n	Membuat matrix baru dengan ukuran baris = m dan kolom = n
increaseMatrix	Public double [] []	Double [] [] matrix1 Double [] [] matrix2	Memperluas matriks dengan menggabungkan matrix1 dan matrix2
copyMatrix	Public double [] []	Double [] [] MIn	Mengembalikan matriks yang merupakan duplikat MIn
epsilonHandler	Public void	Double [] [] matrix	Menghandle angka kecil dan -0

10. Class menu Methods

pilihanMenu	Public int	Int i Int j	Menampilkan pilihan menu dan menerima input nomor menu
inputKeyboard	Public double [] []	Boolean harusSquare	Menerima input pembentuk matriks
inputFilename	Public string	-	Menerima input file path dokumen yang akan dibaca
matrixFile	public double [] []	Boolean harusSquare	Menerima input file berisikan matriks
inputDanBuatMatrix	Public double [] []	Boolean harusSquare	Menampilkan pilihan membuat

			matriks dari input file atau dari input keyboard
matrixBicubicFile	public double [] []	String file	Membuat matriks bicubic 4 x 4 dari file.
taksiranBicubic	public double []	String file	Mengambil nilai a dan b untuk f(a,b) pada bicubic

11. Class regression Methods

regression	Public double []	Double [] [] matrix	Mengembalikan array berisi solusi koefisien dalam persamaan regresi
------------	-------------------	-----------------------	---

12. Class SPL Methods

leadingone	Public int	Double [] [] matrix Int col	Mencari leading one dari matriks
variablemachine	Public char	Int index Int [] arrayIndex Int count	Membuat variabel dari a sampai z
parametersolving	Public string []	Double [] [] matrix Boolean dariGauss	Memberi solusi dalam bentuk parameter
SPLGauss	Public String []	Double [] [] matrix Boolean apaGaussJordan	Memberi solusi SPL dengan metode Gauss
SPLInvers	Public String []	Double [] [] matrix	Memberi solusi SPL dengan metode invers

13. Class InterpolasiPolinom Methods

polynomialInterpolation	Public double []	Double [] [] matrix	Mendapatkan koefisien polinomial dari input berupa koordinat
estimateFunction	Public double	Double [] hasilInterpolasi Double x	Menghitung f(x) dari hasil interpolasi dengan x sebagai parameter

14. Class BicubicInterpolation Methods

change16	Protected double [] []	Double [] [] matrix	Membuat matriks baru dari elemen matrix dengan ukuran 16 x 1
koefisien_ajj	Protected double	Double x Double y Int i Int j	Menghitung koefisien aij
modelSigma	Protected double	Double [] [] matrix Double x Double y	Menghitung interpolasi dengan model sigma
matrix_X	Protected double [] []	-	Membuat matrix X
bicubicInterpolation	Public double	Double [] [] matrix Double x Double y	Melakukan interpolasi bicubic

Bab 4

Eksperimen

1. Solusi SPL $Ax = b$

a. Menggunakan metode Gauss

$$A = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 2 & 5 & -7 & -5 \\ 2 & -1 & 1 & 3 \\ 5 & 2 & -4 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ -2 \\ 4 \\ 6 \end{bmatrix}$$

Matriks SPL yang diperoleh:

```
1.0 1.0 -1.0 -1.0 1.0
0.0 1.0 -1.6666666666666665 -1.0 -1.3333333333333333
0.0 0.0 1.0 -1.0 1.0
0.0 0.0 0.0 0.0 1.0
```

b. Menggunakan metode Gauss Jordan

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$$

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 0.0 -1.0 3.0
0.0 1.0 0.0 0.0 -2.0 0.0
0.0 0.0 0.0 1.0 -1.0 -1.0
0.0 0.0 0.0 0.0 0.0 0.0
```

Solusi SPL adalah:

```
x1 = 3.0 + (1.0)b
x2 = 0.0 + (2.0)b
x3 = a
x4 = -1.0 + (1.0)b
x5 = b
```

C. Menggunakan metode Gauss

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$$

Matriks SPL yang diperoleh:

```
0.0 1.0 0.0 0.0 1.0 0.0 2.0
0.0 0.0 0.0 1.0 0.0 1.0 -2.0
0.0 0.0 0.0 0.0 1.0 -1.0 1.0
```

Solusi SPL adalah:

```
x1 = a
x2 = 1.0 + (-1.0)c
x3 = b
x4 = -2.0 + (-1.0)c
x5 = 1.0 + (1.0)c
x6 = c
```

D. Menggunakan metode Gauss Jordan

$$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \dots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \dots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \dots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \dots & \frac{1}{2n+1} \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

H adalah matriks Hilbert. Cobakan untuk $n = 6$ dan $n = 10$.

$n = 6$

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 0.0 0.0 0.0 9.596501644375174
0.0 1.0 0.0 0.0 0.0 0.0 -33.54912541492277
0.0 0.0 1.0 0.0 0.0 0.0 32.425095026004215
0.0 0.0 0.0 1.0 0.0 0.0 -210.5076010966817
0.0 0.0 0.0 0.0 1.0 0.0 522.6861574407928
0.0 0.0 0.0 0.0 0.0 1.0 -327.1718915601028
```

Solusi SPL adalah:

```
x1 = 9.596501644375174
x2 = -33.54912541492277
x3 = 32.425095026004215
x4 = -210.5076010966817
x5 = 522.6861574407928
x6 = -327.1718915601028
```

n = 10

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 20.15010388877986
0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 -167.6001342183977
0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 340.91878381388966
0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 -212.2386201555614
0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 175.66458440028003
0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 -66.39838687195567
0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 -284.69176567428303
0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 -26.97145330331074
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 89.26711819053773
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 141.53389972439024
```

Solusi SPL adalah:

```
x1 = 20.15010388877986
x2 = -167.6001342183977
x3 = 340.91878381388966
x4 = -212.2386201555614
x5 = 175.66458440028003
x6 = -66.39838687195567
x7 = -284.69176567428303
x8 = -26.97145330331074
x9 = 89.26711819053773
x10 = 141.53389972439024
```

2. SPL berbentuk matriks Augmented

a. Menggunakan metode Gauss Jordan

$$\begin{bmatrix} 1 & -1 & 2 & -1 & -1 \\ 2 & 1 & -2 & -2 & -2 \\ -1 & 2 & -4 & 1 & 1 \\ 3 & 0 & 0 & -3 & -3 \end{bmatrix}.$$

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 -1.0 -1.0
0.0 1.0 -2.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0
```

Solusi SPL adalah:

```
x1 = -1.0 + (1.0)b
x2 = 0.0 + (2.0)a
x3 = a
x4 = b
```

b. Menggunakan metode Gauss Jordan

$$\begin{bmatrix} 2 & 0 & 8 & 0 & 8 \\ 0 & 1 & 0 & 4 & 6 \\ -4 & 0 & 6 & 0 & 6 \\ 0 & -2 & 0 & 3 & -1 \\ 2 & 0 & -4 & 0 & -4 \\ 0 & 1 & 0 & -2 & 0 \end{bmatrix}.$$

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 0.0 0.0
0.0 1.0 0.0 0.0 2.0
0.0 0.0 1.0 0.0 1.0
0.0 0.0 0.0 1.0 1.0
0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0
```

Solusi SPL adalah:

```
x1 = 0.0
x2 = 2.0
x3 = 1.0
x4 = 1.0
```

3. SPL berbentuk persamaan

a. Menggunakan kaidah Cramer

$$8x_1 + x_2 + 3x_3 + 2x_4 = 0$$

$$2x_1 + 9x_2 - x_3 - 2x_4 = 1$$

$$x_1 + 3x_2 + 2x_3 - x_4 = 2$$

$$x_1 + 6x_3 + 4x_4 = 3$$

Solusi SPL menggunakan matriks invers adalah:

```
x1 = -0.22432432432432434
x2 = 0.18243243243243243
x3 = 0.7094594594594594
x4 = -0.2581081081081081
```

b. Menggunakan metode Gauss Jordan

$$\begin{aligned}x_7 + x_8 + x_9 &= 13.00 \\x_4 + x_5 + x_6 &= 15.00 \\x_1 + x_2 + x_3 &= 8.00 \\0.04289(x_3 + x_5 + x_7) + 0.75(x_6 + x_8) + 0.61396x_9 &= 14.79 \\0.91421(x_3 + x_5 + x_7) + 0.25(x_2 + x_4 + x_6 + x_8) &= 14.31 \\0.04289(x_3 + x_5 + x_7) + 0.75(x_2 + x_4) + 0.61396x_1 &= 3.81 \\x_3 + x_6 + x_9 &= 18.00 \\x_2 + x_5 + x_8 &= 12.00 \\x_1 + x_4 + x_7 &= 6.00 \\0.04289(x_1 + x_5 + x_9) + 0.75(x_2 + x_6) + 0.61396x_3 &= 10.51 \\0.91421(x_1 + x_5 + x_9) + 0.25(x_2 + x_4 + x_6 + x_8) &= 16.13 \\0.04289(x_1 + x_5 + x_9) + 0.75(x_4 + x_8) + 0.61396x_7 &= 7.04\end{aligned}$$

Matriks SPL yang diperoleh:

```
1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.9999999999999999 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
```

Solusi SPL adalah:
Tidak ada solusi

4. Studi kasus interpolasi

a. Studi kasus pertama

Gunakan tabel di bawah ini untuk mencari polinom interpolasi dari pasangan titik-titik yang terdapat dalam tabel. Program menerima masukan nilai x yang akan dicari nilai fungsi $f(x)$.

x	0.4	0.7	0.11	0.14	0.17	0.2	0.23
$f(x)$	0.043	0.005	0.058	0.072	0.1	0.13	0.147

Lakukan pengujian pada nilai-nilai default berikut:

$$x = 0.2 \quad f(x) = ?$$

$$x = 0.55 \quad f(x) = ?$$

$$x = 0.85 \quad f(x) = ?$$

$$x = 1.28 \quad f(x) = ?$$

```
Persamaan polinomial dari titik-titik yang diinput:  
y = -4212.434532 x^6 + 7102.399163 x^5 - 4346.313951 x^4 + 1220.854891 x^3 -  
163.915663 x^2 + 10.276384 x^1 - 0.184559
```

$x = 0.2$:

```
Masukkan nilai yang akan ditaksir: 0.2  
Nilai taksiran fungsi saat x = 0.200000 adalah 0.12999999999979278
```

$x = 0.55$:

```
Masukkan nilai yang akan ditaksir: 0.55  
Nilai taksiran fungsi saat x = 0.550000 adalah 2.137571620904623
```

$x = 0.85$:

```
Masukkan nilai yang akan ditaksir: 0.85  
Nilai taksiran fungsi saat x = 0.850000 adalah -66.269639315218
```

$x = 1.28$:

```
Masukkan nilai yang akan ditaksir: 1.28  
Nilai taksiran fungsi saat x = 1.280000 adalah -3485.1449016089537
```


b. Studi kasus kedua

Jumlah kasus positif baru Covid-19 di Indonesia semakin fluktuatif dari hari ke hari. Di bawah ini diperlihatkan jumlah kasus baru Covid-19 di Indonesia mulai dari tanggal 17 Juni 2022 hingga 31 Agustus 2022:

Tanggal	Tanggal (desimal)	Jumlah Kasus Baru
17/06/2022	6,567	12.624
30/06/2022	7	21.807
08/07/2022	7,258	38.391
14/07/2022	7,451	54.517
17/07/2022	7,548	51.952
26/07/2022	7,839	28.228
05/08/2022	8,161	35.764
15/08/2022	8,484	20.813
22/08/2022	8,709	12.408
31/08/2022	9	10.534

Tanggal (desimal) adalah tanggal yang sudah diolah ke dalam bentuk desimal 3 angka di belakang koma dengan memanfaatkan perhitungan sebagai berikut:

$$\text{tanggal(desimal)} = \text{bulan} + (\text{tanggal} / \text{jumlah hari pada bulan tersebut})$$

Sebagai **contoh**, untuk tanggal 17/06/2022 (dibaca: 17 Juni 2022) diperoleh tanggal(desimal) sebagai berikut:

$$\text{Tanggal(desimal)} = 6 + (17/30) = 6,567$$

Gunakanlah data di atas dengan memanfaatkan **polinom interpolasi** untuk melakukan prediksi jumlah kasus baru Covid-19 pada tanggal-tanggal berikut:

- 16/07/2022
- 10/08/2022
- 05/09/2022
- beserta masukan user lainnya berupa **tanggal (desimal) yang sudah diolah** dengan asumsi prediksi selalu dilakukan untuk tahun 2022.

Persamaan polinomial dari titik-titik yang diinput:
 $y = -141086,352655 x^9 + 9373741,525539 x^8 - 275502593,064479 x^7 + 4696316966,551331 x^6 - 51137864858,132620 x^5 + 368597567443,480700 x^4 - 1757053341532,453100 x^3 + 5335014967217,772000 x^2 - 9348572690401,220000 x + 7188430348201,680000$

$$y = -141006.352655x^9 + 9373741.525539x^8 - 275502503.064479x^7 + 4696316966.551331x^6 - 51137864858.132620x^5 + 368597567443.480700x^4 - 1757053341532.453100x^3 + 5335014967217.772000x^2 - 9348572690401.220000x^1 + 7188430348201.680000$$

Taksiran jumlah kasus Covid-19 baru di tanggal:

a) 16/07/2022

$$7 + (16/31) = 7,516$$

Masukkan nilai yang akan ditaksir: 7.516

Nilai taksiran fungsi saat $x = 7.516000$ adalah 53535.1953125

b) 10/08/2022

$$8 + (10/31) = 8,323$$

Masukkan nilai yang akan ditaksir: 8.323

Nilai taksiran fungsi saat $x = 8.323000$ adalah 36291.91015625

c) 05/09/2022

$$9 + (5/30) = 9,167$$

Masukkan nilai yang akan ditaksir: 9.167

Nilai taksiran fungsi saat $x = 9.167000$ adalah -667716.0234375

d) 13/08/2022

$$8 + (13/31) = 8,419$$

Masukkan nilai yang akan ditaksir: 8.419

Nilai taksiran fungsi saat $x = 8.419000$ adalah 28495.0390625

c. Studi kasus ketiga

Sederhanakan fungsi

$$f(x) = \frac{x^2 + \sqrt{x}}{e^x + x}$$

dengan polinom interpolasi derajat n di dalam selang $[0, 2]$. Sebagai contoh, jika $n = 5$, maka titik-titik x yang diambil di dalam selang $[0, 2]$ berjarak $h = (2 - 0)/5 = 0.4$.

testcasepolinom3.txt

0.0 0.0

0.4 0.419

0.8 0.507

1.2 0.561

1.6 0.584

2.0 0.577

Persamaan polinomial dari titik-titik yang diinput:
 $y = 0.237630 x^5 - 1.429036 x^4 + 3.252604 x^3 - 3.565104 x^2 + 2.038500 x^1$

5. Studi kasus interpolasi Bicubic

Diberikan matriks input:

153	59	210	96
125	161	72	81
98	101	42	12
21	51	0	16

Tentukan nilai:

$$f(0,0) = ?$$

$$f(0.5, 0.5) = ?$$

$$f(0.25, 0.75) = ?$$

$$f(0.1, 0.9) = ?$$

$f(0,0)$:

```
testcasebicubic.txt
153 59 210 96
125 161 72 81
98 101 42 12
21 51 0 16
0 0
```

Masukkan nama file

--> testcasebicubic.txt

Nilai taksiran $f(0.000000, 0.000000)$ adalah 161.0

$f(0.5, 0.5)$:

```
testcasebicubic.txt
153 59 210 96
125 161 72 81
98 101 42 12
21 51 0 16
0.5 0.5
```

```
Masukkan nama file
--> testcasebicubic.txt
Nilai taksiran  $f(0.500000, 0.500000)$  adalah 97.7265625
```

$f(0.25, 0.75)$:

```
testcasebicubic.txt
153 59 210 96
125 161 72 81
98 101 42 12
21 51 0 16
0.25 0.75
```

```
Masukkan nama file
--> testcasebicubic.txt
Nilai taksiran  $f(0.250000, 0.750000)$  adalah 105.5147705078125
```

$f(0.1, 0.9)$:

```
testcasebicubic.txt
153 59 210 96
125 161 72 81
98 101 42 12
21 51 0 16
0.1 0.9
```

```
Masukkan nama file
--> testcasebicubic.txt
Nilai taksiran  $f(0.100000, 0.900000)$  adalah 104.22911850000001
```

6. Studi kasus Regresi Linier Berganda

Diberikan sekumpulan data sesuai pada tabel berikut ini.

Table 12.1: Data for Example 12.1

Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3	Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3
0.90	72.4	76.3	29.18	1.07	23.2	76.8	29.38
0.91	41.6	70.3	29.35	0.94	47.4	86.6	29.35
0.96	34.3	77.1	29.24	1.10	31.5	76.9	29.63
0.89	35.1	68.0	29.27	1.10	10.6	86.3	29.56
1.00	10.7	79.0	29.78	1.10	11.2	86.0	29.48
1.10	12.9	67.4	29.39	0.91	73.3	76.3	29.40
1.15	8.3	66.8	29.69	0.87	75.4	77.9	29.28
1.03	20.1	76.9	29.48	0.78	96.6	78.7	29.29
0.77	72.2	77.7	29.09	0.82	107.4	86.8	29.03
1.07	24.0	67.7	29.60	0.95	54.9	70.9	29.37

Source: Charles T. Hare, "Light-Duty Diesel Emission Correction Factors for Ambient Conditions," EPA-600/2-77-116. U.S. Environmental Protection Agency.

Gunakan *Normal Estimation Equation for Multiple Linear Regression* untuk mendapatkan regresi linear berganda dari data pada tabel di atas, kemudian

estimasi nilai Nitrous Oxide apabila Humidity bernilai 50%, temperatur 76°F, dan tekanan udara sebesar 29.30.

Dari data-data tersebut, apabila diterapkan *Normal Estimation Equation for Multiple Linear Regression*, maka diperoleh sistem persamaan linear sebagai berikut.

$$20b_0 + 863.1b_1 + 1530.4b_2 + 587.84b_3 = 19.42$$

$$863.1b_0 + 54876.89b_1 + 67000.09b_2 + 25283.395b_3 = 779.477$$

$$1530.4b_0 + 67000.09b_1 + 117912.32b_2 + 44976.867b_3 = 1483.437$$

$$587.84b_0 + 25283.395b_1 + 44976.867b_2 + 17278.5086b_3 = 571.1219$$

Persamaan regresi linear berganda adalah:

$$y = -3.507778 - 0.002625 x_1 + 0.000799 x_2 + 0.154155 x_3$$

Untuk menaksir nilai fungsi, masukkan 3 peubah yang akan ditaksir nilai fungsinya.

50 76 29.30

Nilai taksiran adalah 0.938434

Bab 5

Kesimpulan, Saran, dan Refleksi

A. Kesimpulan

Hasil program kelompok kami dapat digunakan untuk :

1. Menghitung solusi SPL dengan menggunakan metode eliminasi Gauss, eliminasi Gauss - Jordan, matriks balikan, dan kaidah Cramer
2. Menghitung determinan matriks dengan metode eliminasi Gauss dan ekspansi Kofaktor
3. Menghitung matriks balikan.
4. Menyelesaikan persoalan interpolasi polinom dan regresi linier berganda.

B. Saran

Kami memiliki saran untuk memberi contoh percobaan pada studi kasus yang dilampirkan di file instruksi tubes, khususnya untuk file yang sangat besar. Karena kami mengalami kendala dalam meng-handle matriks yang sangat besar.

C. Refleksi

Selama pengerjaan tubes ini, kami mempelajari hal-hal penting seperti betapa pentingnya koordinasi antar anggota kelompok sehingga tugas terbagi secara merata, lalu kami juga mempelajari lebih banyak mengenai cara menulis kode yang elegan mengenai matriks serta implementasinya ke masalah interpolasi dan regresi.

Referensi

- <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-03-Sistem-Persamaan-Linier.pdf>
- <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-05-Sistem-Persamaan-Linier-2.pdf>
- <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-08-Determinan-bagian1.pdf>
- <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-09-Determinan-bagian2.pdf>
- Anton, Howard and Chris Rorres. 2006. *Elementary Linear Algebra*. New Jersey: Wiley.
- <https://rpubs.com/ummiku/841844>
- https://www.mssc.mu.edu/~daniel/pubs/RoweTalkMSCS_BiCubic.pdf

Link Repository : <https://github.com/BreezyDR/Algeo01-21047>