

LAPORAN TUGAS BESAR I
IF2123 ALJABAR LINEAR DAN GEOMETRI

Sistem Persamaan Linear, Determinan, dan Aplikasinya



Disusun oleh:

Kenneth Ezekiel Suprantonio (13521089)

Noel Christoffel Simbolon (13521096)

Vanessa Rebecca Wiyono (13521151)

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

2022

DAFTAR ISI

DAFTAR ISI	i
BAB I DESKRIPSI MASALAH.....	1
I. Tujuan.....	1
II. Spesifikasi	1
BAB II TEORI SINGKAT.....	4
I. Determinan Matriks.....	4
II. Metode Eliminasi Gauss	6
III. Metode Eliminasi Gauss-Jordan.....	8
IV. Matriks Balikan (<i>Inverse</i>).....	9
V. Matriks Kofaktor	10
VI. Matriks Adjoint.....	10
VII. Kaidah Cramer	10
VIII. Interpolasi Polinom	11
IX. Regresi Linear Berganda.....	12
X. Interpolasi Bikubik	14
BAB III IMPLEMENTASI.....	16
I. Class Matrix.....	16
II. Class InputOutput.....	18
III. Class InterpolasiBikubik	19
IV. Class PolynomialInterpolation.....	20
V. Class SPL	21
VI. Class UserInterface	21
VII. Class RegresiLinear	22
BAB IV EKSPERIMEN.....	23
I. Studi Kasus Sistem Persamaan Linear	23
II. Studi Kasus Interpolasi Polinom	26
III. Studi Kasus Interpolasi Bikubik	28
IV. Studi Kasus Regresi Linear Berganda.....	29
BAB V PENUTUP	31
I. Simpulan	31
II. Saran	31
III. Refleksi.....	31
DAFTAR PUSTAKA	32

BAB I

DESKRIPSI MASALAH

I. Tujuan

Tujuan dari tugas besar ini adalah sebagai berikut.

1. Membuat satu atau lebih *library* dalam bahasa Java untuk:
 - Membuat fungsi-fungsi untuk menemukan solusi Sistem Persamaan Linear (Metode Eliminasi Gauss, Metode Eliminasi Gauss-Jordan, Metode Matriks Balikan, dan Kaidah Cramer (Khusus untuk SPL untuk N peubah dan N persamaan)).
 - Menghitung Determinan matriks dengan Metode Reduksi Baris dan Ekspansi Kofaktor.
 - Menghitung Balikan dari sebuah Matriks Persegi.
2. Selanjutnya, menggunakan *library* tersebut untuk menyelesaikan persoalan yang dimodelkan dalam SPL seperti:
 - Persoalan Interpolasi Polinom
 - Persoalan Interpolasi Bikubik
 - Persoalan Regresi Linier Berganda

II. Spesifikasi

1. Program dapat menerima masukan (input) baik dari keyboard maupun membaca masukan dari file text. Untuk SPL, masukan dari keyboard adalah m, n, koefisien a_{ij} , dan b_i . Masukan dari file berbentuk matriks augmented tanpa tanda kurung, setiap elemen matriks dipisah oleh spasi. Misalnya,

```
3 4.5 2.8 10 12
5 -3 7 8.3 11 -4
0.5 -10 -9 12 0
```

2. Untuk persoalan menghitung determinan dan matriks balikan, masukan dari keyboard adalah n dan koefisien a_{ij} . Masukan dari file berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

```
3 4.5 2.8
-3 7 8.3
0.5 -10 -9
```

3. Untuk persoalan interpolasi, masukannya jika dari keyboard adalah n, (x_0, y_0) , (x_1, y_1) , ..., (x_n, y_n) , dan nilai x yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. Misalnya jika titik-titik datanya adalah $(8.0, 2.0794)$, $(9.0, 2.1972)$, dan $(9.5, 2.2513)$, maka di dalam file text ditulis sebagai berikut.

```
8.0 2.0794
```

9.0 2.1972

9.5 2.2513

4. Untuk persoalan regresi, masukannya jika dari keyboard adalah n (jumlah peubah x), m (jumlah sampel), semua nilai-nilai x_1, x_2, \dots, x_n , nilai y_i , dan nilai-nilai x_k yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.
5. Untuk persoalan SPL, luaran (output) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya $x_4 = -2$, $x_3 = 2s - t$, $x_2 = s$, dan $x_1 = t$.)
6. Untuk persoalan determinan dan matriks balikan, maka luarannya sesuai dengan persoalan masing-masing

-3 7 8.3 11 -4

3 4.5 2.8 10 12

0.5 -10 -9 12 0

0.1 0.2

7. Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada x yang diberikan. Contoh luaran untuk interpolasi adalah $f(x) = -0.0064x + 0.2266 + 0.6762x^2$, $f(5) = \dots$ dan untuk regresi adalah $f(x) = -9.5872 + 1.0732x$, $f(x_k) = \dots$
8. Untuk persoalan interpolasi bicubik, masukan dari file text (.txt) yang berisi matriks berukuran 4×4 yang berisi nilai $f(i,j)$ dengan i dan j adalah indeks matriks diikuti dengan nilai a dan b untuk mencari nilai $f(a,b)$. misalnya jika 6 nilai dari $f(-1,-1)$, $f(-1,0)$, $f(-1,1)$, $f(-1,2)$, $f(0,-1)$, $f(0,0)$, $f(0,1)$, $f(0,2)$, $f(1,-1)$, $f(1,0)$, $f(1,1)$, $f(1,2)$, $f(2,-1)$, $f(2,0)$, $f(2,1)$, $f(2,2)$ berturut-turut adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 serta nilai a dan b yang dicari berturut-turut adalah 0.5 dan 0.5 maka isi file text ditulis sebagai berikut.

1 2 3 4

5 6 7 8

9 10 11 12

13 14 15 16

0.5 0.5

luaran yang dihasilkan adalah nilai dari $f(0.5,0.5)$. masukannya adalah matriks 4×4 , diikuti oleh nilai a dan b , maka luarannya adalah nilai $f(a,b)$.

9. Luaran program harus dapat ditampilkan pada layar komputer dan dapat disimpan ke dalam file.
10. Bahasa program yang digunakan adalah Java.

11. Program tidak harus berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas Eclipse misalnya).
12. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

MENU

1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar

Untuk pilihan menu nomor 1 ada sub-menu lagi yaitu pilihan metode:

1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer

Begitu juga untuk pilihan menu nomor 2 dan 3.

BAB II TEORI SINGKAT

I. Determinan Matriks

Matrix persegi merupakan matrix dengan jumlah baris dan kolom yang sama banyaknya, sehingga hanya pada matrix persegi lah determinan dapat dihitung. Namun, hal ini tidak berarti bahwa seluruh matrix persegi pasti memiliki determinan. Dalam menghitung determinan sebuah matrix terdapat dua metode yang dapat digunakan dan antara lain adalah dengan ekspansi kofaktor dan dengan eliminasi gauss.

1.1 METODE EKSPANSI KOFAKTOR

Jika A adalah sebuah matriks persegi, maka minor entri a_{ij} dinyatakan oleh M_{ij} dan didefinisikan menjadi determinan submatriks yang tetap setelah baris ke i dan kolom ke j dicoret menjadi determinan submatriks yang tetap setelah baris ke i dan kolom ke j dicoret dari A . Bilangan $(-1)^{i+j}M_{ij}$ dinyatakan sebagai C_{ij} dan dinamakan kofaktor entri atau a_{ij} . Berikut contoh dari minor entri dan kofaktor.

$$\text{Contoh: } A = \begin{bmatrix} 3 & 1 & -4 \\ 2 & 5 & 6 \\ 1 & 4 & 8 \end{bmatrix}$$

$$M_{11} = \begin{vmatrix} 5 & 6 \\ 4 & 8 \end{vmatrix} = 16, C_{11} = (-1)^{1+1} \cdot M_{11} = 16$$

$$M_{32} = \begin{vmatrix} 3 & -4 \\ 2 & 6 \end{vmatrix} = 26, C_{32} = (-1)^{3+2} \cdot M_{32} = -26$$

2.1 Kofaktor dari suatu matriks

Berikut tabel yang merepresentasikan tanda positif dan negatif pada kofaktor.

$$\begin{vmatrix} + & - & + & \dots \\ - & + & - & \dots \\ + & - & + & \dots \\ \dots & \dots & \dots & \dots \end{vmatrix}$$

2.2 Tabel tanda positif dan negatif dari kofaktor

Berikut contoh perhitungan determinan matriks dengan menggunakan ekspansi kofaktor.

$$A = \begin{bmatrix} 3 & 1 & -4 \\ 2 & 5 & 6 \\ 1 & 4 & 8 \end{bmatrix}$$

$$\det(A) = 3 \begin{vmatrix} 5 & 6 \\ 4 & 8 \end{vmatrix} - 1 \begin{vmatrix} 2 & 6 \\ 1 & 8 \end{vmatrix} + (-4) \begin{vmatrix} 2 & 5 \\ 1 & 4 \end{vmatrix}$$

$$\det(A) = 48 - 10 - 12 = 26$$

2.3 Pencarian determinan menggunakan ekspansi kofaktor

1.2 METODE REDUKSI BARIS

Determinan juga dapat dicari dengan menggunakan Operasi Baris Elementer untuk mendapatkan matriks segitiga atas maupun bawah. Berikut contoh mencari determinan dengan menggunakan matriks segitiga.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{33} & a_{34} & a_{35} \\ 0 & 0 & 0 & a_{44} & a_{45} \\ 0 & 0 & 0 & 0 & a_{55} \end{bmatrix}, \det(A) = a_{11}a_{22}a_{33}a_{44}a_{55}$$

2.4 Mencari determinan dengan menggunakan matriks segitiga

Poin utama dari metode ini adalah penggunaan OBE dalam membuat sebuah matriks menjadi matriks segitiga atas atau matriks segitiga bawah.

$$[A] \sim [\textit{segitiga atas atau bawah}]$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1k} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2k} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & a_{k3} & \cdots & a_{kk} \end{bmatrix} \sim \begin{bmatrix} a'_{11} & a'_{12} & a'_{13} & \cdots & a'_{1k} \\ 0 & a'_{22} & a'_{23} & \cdots & a'_{2k} \\ 0 & 0 & a_{33} & \cdots & a'_{3k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a'_{kk} \end{bmatrix}$$

2.5 Pencarian determinan menggunakan matriks segitiga dan operasi baris elementer

(Nilai p menyatakan banyaknya pertukaran baris dalam Operasi Baris Elementer)

II. Metode Eliminasi Gauss

Metode eliminasi Gauss sangat erat kaitannya dengan Matriks Eselon baris yang merupakan matriks dengan 1 utama pada setiap baris, kecuali baris yang seluruhnya adalah 0. Berikut contoh matriks eselon baris.

$$\begin{bmatrix} 1 & * & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & * & * & * \\ 0 & 0 & 1 & * \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & * & * & * \\ 0 & 0 & 0 & 1 & * \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

2.6 Contoh Matriks eselon baris

(* = nilai sembarang)

Matriks Eselon Baris memiliki sifat-sifat yang antara lain adalah:

1. Jika sebuah baris tidak terdiri dari seluruhnya 0, maka bulangan bukan 0 pertama dalam baris tersebut adalah 1 (merupakan 1 utama)
2. Jika ada baris yang seluruhnya 0, maka baris tersebut harus diposisikan pada bagian bawah matriks
3. Pada dua baris berurutan yang tidak seluruhnya 0, maka 1 utama pada baris yang lebih rendah terletak lebih ke kanan daripada 1 utama pada baris yang lebih tinggi.

Langkah-langkah mencari solusi SPL dengan metode Eliminasi Gauss:

1. Nyatakan Sistem Persamaan Linear dalam bentuk matriks *augmented*
2. Terapkan Operasi Baris Elementer agar terbentuk matriks eselon baris
3. Terapkan *backward substitution* untuk mencari solusi dari tiap variabel.

Berikut contoh penyelesaian Sistem Persamaan Linear dengan menggunakan Metode Eliminasi Gauss

$$a + 2b + c + 2d = 2$$

$$2a - b + c + d = 0$$

$$3a + 2b - c + d = 1$$

$$a + b + c - d = 9.$$

Penyelesaian :

Matriks perluasan dari SPL di atas adalah

$$\left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 2 & -1 & 1 & 1 & 0 \\ 3 & 2 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 & 9 \end{array} \right]$$

Selanjutnya diselesaikan dengan menggunakan metode eliminasi Gauss.

$$\begin{aligned} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 2 & -1 & 1 & 1 & 0 \\ 3 & 2 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 & 9 \end{array} \right] & \xrightarrow{b_2-2b_1, b_3-3b_1, b_4-b_1} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 0 & -5 & -1 & -3 & -4 \\ 0 & -4 & -4 & -5 & -5 \\ 0 & -1 & 0 & -3 & 7 \end{array} \right] \\ & \xrightarrow{b_2 \times (-1), b_3 \times (-1), b_4 \times (-1)} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 0 & 5 & 1 & 3 & 4 \\ 0 & 4 & 4 & 5 & 5 \\ 0 & 1 & 0 & 3 & -7 \end{array} \right] \\ & \xrightarrow{b_2 \leftrightarrow b_4} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 0 & 1 & 0 & 3 & -7 \\ 0 & 4 & 4 & 5 & 5 \\ 0 & 5 & 1 & 3 & 4 \end{array} \right] \\ & \xrightarrow{b_4 \leftrightarrow 4b_3} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 0 & 1 & 0 & 3 & -7 \\ 0 & 0 & 1 & -12 & 39 \\ 0 & 0 & 0 & -41 & 123 \end{array} \right] \\ & \xrightarrow{b_4 \times \frac{-1}{41}} \left[\begin{array}{cccc|c} 1 & 2 & 1 & 2 & 2 \\ 0 & 1 & 0 & 3 & -7 \\ 0 & 0 & 1 & -12 & 39 \\ 0 & 0 & 0 & 1 & -3 \end{array} \right] \end{aligned}$$

Diperoleh sistem persamaan baru

$$\begin{aligned} a + 2b + c + 2d &= 2 \\ b + 3d &= -7 \\ c - 12d &= 39 \\ d &= -3. \end{aligned}$$

Dengan mensubstitusikan nilai $d = -3$ ke persamaan ke-2 dan ke-3,

diperoleh nilai $b = 2$ dan $c = 3$.

Selanjutnya, dengan mensubstitusikan nilai $b = 2$, $c = 3$ dan $d = -3$ ke persamaan ke-1,

diperoleh nilai $a = 1$.

Jadi penyelesaian dari SPL

$$\begin{aligned} a + 2b + c + 2d &= 2 \\ 2a - b + c + d &= 0 \\ 3a + 2b - c + d &= 1 \\ a + b + c - d &= 9. \end{aligned}$$

adalah $a = 1$, $b = 2$, $c = 3$ dan $d = -3$.

2.7 Contoh pencarian solusi Sistem Persamaan Linear dengan menggunakan Metode Gauss

III. Metode Eliminasi Gauss-Jordan

Memiliki prinsip utama dengan metode Gauss, metode Eliminasi Gauss Jordan juga dapat digunakan untuk mencari solusi suatu Sistem Persamaan Linear. Secara umum, kedua metode tersebut menggunakan Operasi Baris Elementer untuk mengubah bentuk matriks menjadi *augmented*. Namun, hal yang membedakan keduanya adalah tidak diterapkannya *backward substitution* pada Metode Eliminasi Gauss-Jordan karena bentuk akhir matriksnya merupakan Matriks Eselon Baris Tereduksi (*reduced row echelon form*).

Sifat Matriks Eselon Baris Tereduksi mirip dengan sifat dari Matriks Eselon Baris, hal yang membedakan adalah bahwa pada Matriks Eselon Baris Tereduksi setiap kolom yang memiliki 1 utama juga memiliki 0 di kolom lainnya. Berikut contoh dari Matriks Eselon Baris Tereduksi.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & * \\ 0 & 1 & 0 & * \\ 0 & 0 & 1 & * \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

2.8 Contoh Matriks Eselon Baris Tereduksi

(* = nilai sembarang)

Langkah-langkah untuk mencari solusi dari Sistem Persamaan Linear dengan Metode Eliminasi Gauss-Jordan, antara lain adalah:

1. Fase maju atau fase eliminasi Gauss (menghasilkan nilai 0 dibawah 1 utama).
2. Fase mundur atau *backward phase* (menghasilkan 0 diatas 1 utama).

Berikut contoh penyelesaian Sistem Persamaan Linear dengan metode Eliminasi Gauss-Jordan

$$\begin{aligned}2x + 3y - z &= 5 \\4x + 4y - 3z &= 3 \\-2x + 3y - z &= 1\end{aligned}$$

Dari fase eliminasi Gauss:

$$\begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Eliminasi fase mundur

$$\begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \begin{matrix} R_1 - 3/2 R_2 \\ \sim \\ R_2 - 1/2 R_3 \end{matrix} \begin{bmatrix} 1 & 0 & -5/4 & -11/4 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \begin{matrix} R_1 + 5/4 R_3 \\ \sim \\ R_2 - 1/2 R_3 \end{matrix}$$

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

Diperoleh solusi dari Sistem Persamaan linear adalah sebagai berikut:

$$\begin{aligned}x &= 1 \\y &= 2 \\z &= 3\end{aligned}$$

2.9 Pencarian solusi Sistem Persamaan Linear dengan Metode Gauss-Jordan

IV. Matriks Balikan (*Inverse*)

Balikan atau invers dari suatu matriks akan menghasilkan matriks identitas jika dilakukan perkalian dengan matriks awal ($A^{-1}A = I$). Umumnya, penggunaan matriks balikan bertujuan untuk mencari solusi dari Sistem Persamaan Linear yang memiliki solusi tunggal

$$Ax = B$$

$$A^{-1}Ax = A^{-1}B$$

$$Ix = A^{-1}B$$

$$x = A^{-1}B$$

(I adalah matriks identitas)

Terdapat dua metode untuk mencari matriks balikan, yaitu metode kofaktor dan metode eliminasi Gauss-Jordan

4.1 METODE KOFAKTOR

Pada metode kofaktor determinan harus dicari terlebih dahulu. Apabila suatu matriks tidak memiliki determinan, maka matriks tersebut juga tidak memiliki matriks balikan. Namun, jika matriks memiliki determinan maka dapat dilanjutkan pada langkah selanjutnya yaitu mencari matriks kofaktor yang kemudian akan ditranspose sehingga akan menghasilkan matriks adjoint. Berikut rumus dari metode kofaktor dalam mencari matriks balikan.

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

4.2 METODE GAUSS-JORDAN

Matriks balikan pada metode Gauss-Jordan diperoleh dengan mengubah matriks *augmented* $[A/I] \sim \text{Gauss Jordan} \sim [I|A^{-1}]$. Dalam hal ini, I merupakan matriks identitas dengan ukuran yang sama dengan ukuran matriks A.

V. Matriks Kofaktor

Matriks kofaktor merupakan matriks yang terbentuk oleh kofaktor-kofaktor. Susunan elemen pada matriks ini juga sesuai dengan susunan pada matriksnya. Berikut contoh penulisan matriks kofaktor

$$\begin{bmatrix} C_{11} & C_{12} & C_{13} & \cdots & C_{1k} \\ C_{21} & C_{22} & C_{23} & \cdots & C_{2k} \\ C_{31} & C_{32} & C_{33} & \cdots & C_{3k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{k1} & C_{k2} & C_{k3} & \cdots & C_{kk} \end{bmatrix}$$

2.10 Contoh matriks kofaktor

VI. Matriks Adjoint

Adjoint dari suatu matriks persegi dapat didefinisikan sebagai transpose dari matriks kofaktornya, dengan penulisan $\text{adj}(\text{Matriks})$. Adjoint dari suatu matriks akan digunakan dalam mencari matriks balikan dari suatu matriks (jika menggunakan metode kofaktor)

VII. Kaidah Cramer

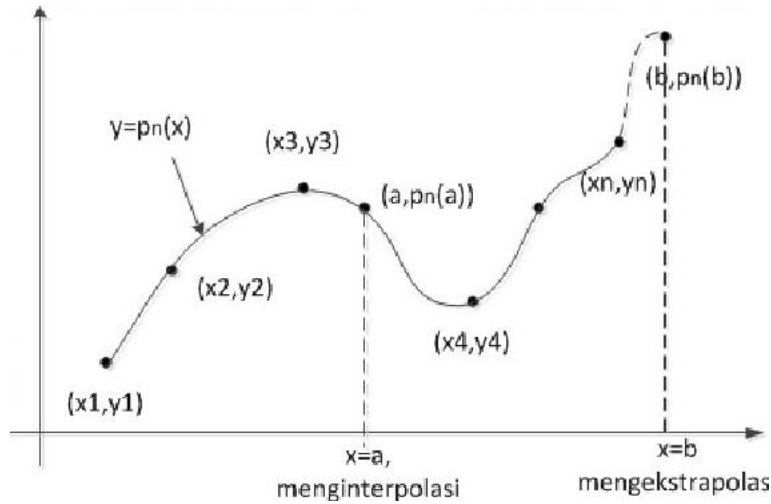
Kaidah Cramer digunakan untuk mencari solusi dari suatu Sistem Persamaan Linear yang memiliki banyak variabel dan berlaku ketika sistem memiliki solusi tunggal. Pencarian solusi dilakukan dengan menggunakan determinan matriks koefisien (dari sistem persamaan) dan

determinan matriks lain yang diperoleh melalui penggantian salah satu kolom matriks koefisien dengan vektor yang terdapat pada sebelah kanan persamaan. Misal $Ax = b$ merupakan SPL yang terdiri dari k persamaan linear dengan k variabel. Jika $\det(A) \neq 0$, maka setiap variabel akan memiliki solusi unik, yaitu:

$$x_1 = \frac{\det(A_1)}{\det(A)}, x_2 = \frac{\det(A_2)}{\det(A)}, \dots, x_k = \frac{\det(A_k)}{\det(A)}$$

(Matriks A_i didapat dengan mengubah entri pada kolom ke- i dengan matriks b)

VIII. Interpolasi Polinom



2.11. Kurva interpolasi polinom

Interpolasi Polinom merupakan suatu metode analisis numerik yang mengaproksimasi suatu nilai fungsi polinom berdasarkan beberapa titik data yang diketahui. Interpolasi Polinom dapat digunakan untuk modelkan bagaimana data berubah-ubah terhadap variabel independennya. Selain membuat prediksi, Interpolasi Polinom juga dapat digunakan untuk menganalisis hubungan antara variabel-variabel data (berbanding terbalik, tidak berhubungan, berbanding lurus / linear, berbanding kuadratik, dll).

Berikut langkah-langkahnya:

1. Misal diketahui $(n+1)$ buah titik data yang berbeda

$$(x_0, y_0), (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$$

2. Akan terdapat suatu polinom $P_n(x)$ yang memenuhi:

$$y_i = p_n(x_i) \quad \forall i = 0, 1, 2, \dots, n$$

Polinom tersebut dapat memiliki bentuk:

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

3. Buat Matriks *augmented* dari persamaan yang didapat

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n & y_0 \\ 1 & x_1 & x_1^2 & \dots & x_1^n & y_1 \\ 1 & x_2 & x_2^2 & \dots & x_2^n & y_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n & y_n \end{bmatrix}$$

4. Lakukan Eliminasi Gauss atau Gauss-Jordan pada matriks *augmented* untuk mendapatkan nilai koefisien dari setiap variabel. contoh menggunakan metode eliminasi Gauss-Jordan:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & k_0 \\ 0 & 1 & 0 & \dots & 0 & k_1 \\ 0 & 0 & 1 & \dots & 0 & k_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & k_n \end{bmatrix}$$

Solusi:

$$a_0 = k_0$$

$$a_1 = k_1$$

$$a_2 = k_2$$

$$\vdots$$

$$a_n = k_n$$

IX. Regresi Linear Berganda

Regresi Linear Berganda merupakan metode statistika yang digunakan dalam pemodelan hubungan antara beberapa variabel independen terhadap suatu variabel dependen. Sesuai dengan namanya, Regresi Linear Berganda hanya dapat digunakan untuk membuat hampiran linear dari beberapa titik data yang berbeda. Tidak seperti Interpolasi Polinom yang dapat digunakan untuk membuat polinomial derajat n . Walau demikian, Regresi Linear Berganda dapat digunakan untuk memodelkan banyak variabel independen secara bersamaan, tidak seperti Interpolasi Polinom yang hanya dapat memodelkan satu variabel independen saja. Namun, keduanya memiliki persamaan dimana Regresi Linear Berganda juga dapat digunakan untuk memperkirakan hubungan antara variabel data independen (hubungan yang dapat ditinjau hanya yang merupakan hubungan berbanding terbalik, berhubungan, atau berbanding lurus / linear).

Berikut langkah-langkah pencarian persamaan Regresi Linear Berganda:

1. Misal terdapat k peubah $(x_1, x_2, x_3, \dots, x_k)$
2. Dibutuhkan k buah titik data yang berbeda untuk melakukan regresi
3. Akan terdapat fungsi $f(x_1, x_2, x_3, \dots, x_k)$ yang memenuhi $y_i = f(x_{1i}, x_{2i}, x_{3i}, \dots, x_{ki})$

Fungsi tersebut kemudian dapat diasumsikan memiliki bentuk:

$$f(x_1, x_2, x_3, \dots, x_k) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \dots + \beta_k x_k + \epsilon_i$$

4. Buat matriks *augmented* dari persamaan tersebut dengan rumus *Normal Estimation Equation for Multiple Linear Regression*

$$\begin{bmatrix} k & \sum_{i=1}^k x_{1i} & \sum_{i=1}^k x_{2i} & \dots & \sum_{i=1}^k x_{ki} & \sum_{i=1}^k y_i \\ \sum_{i=1}^k x_{1i} & \sum_{i=1}^k x_{1i}^2 & \sum_{i=1}^k x_{1i}x_{2i} & \dots & \sum_{i=1}^k x_{1i}x_{ki} & \sum_{i=1}^k x_{1i}y_i \\ \sum_{i=1}^k x_{2i} & \sum_{i=1}^k x_{2i}x_{1i} & \sum_{i=1}^k x_{2i}^2 & \dots & \sum_{i=1}^k x_{2i}x_{ki} & \sum_{i=1}^k x_{2i}y_i \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \sum_{i=1}^k x_{ki} & \sum_{i=1}^k x_{ki}x_{1i} & \sum_{i=1}^k x_{ki}x_{2i} & \dots & \sum_{i=1}^k x_{ki}^2 & \sum_{i=1}^k x_{ki}y_i \end{bmatrix}$$

5. Lakukan Eliminasi Gauss atau Gauss-Jordan pada matriks *augmented* untuk mendapat nilai koefisien dari setiap variabel

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & n_0 \\ 0 & 1 & 0 & \dots & 0 & n_1 \\ 0 & 0 & 1 & \dots & 0 & n_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & n_k \end{bmatrix}$$

Solusi:

$$\beta_0 = n_0$$

$$\beta_1 = n_1$$

$$\beta_2 = n_2$$

$$\vdots$$

$$\beta_k = n_k$$

X. Interpolasi Bikubik

Interpolasi Bikubik adalah ekstensi dari interpolasi kubik untuk menginterpolasikan poin-poin data dalam kisi-kisi regular. Interpolasi Bikubik juga merupakan Teknik interpolasi pada data 2D umumnya digunakan untuk pembesaran citra.

Model interpolasi bikubik adalah sebagai berikut:

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

Dengan persamaan matriks:

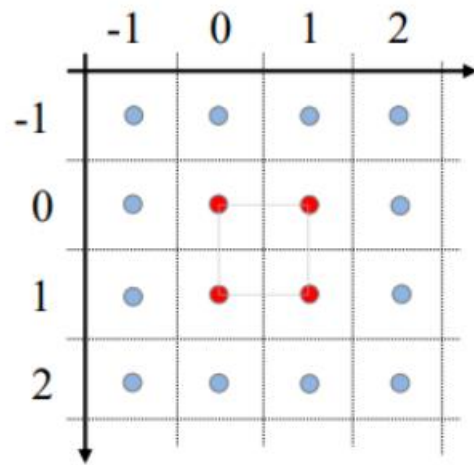
$$F = XY \cdot a$$

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

Sehingga akan bisa didapatkan matriks **a** dengan cara mengalikan matriks **F** yang dimiliki dengan invers matriks **XY** (karena matriks **XY** konstan):

$$XY^{-1} \cdot F = a$$

Lalu matriks **a** yang didapat bisa digunakan untuk nilai variable dalam $f(x, y)$ lalu melakukan interpolasi berdasarkan $f(a, b)$ dari matriks 4x4 di dalam rentang indeks 0 sampai 1



BAB III IMPLEMENTASI

I. Class Matrix

- Attribute

Nama	Tipe	Deskripsi
input	Static Scanner	Scanner untuk input yang digunakan dalam class Matrix
M	Double[][]	Buffer yang menyimpan nilai matriks
row	int	Jumlah Baris
col	int	Jumlah Kolom

- Method

Nama	Tipe	Parameter	Deskripsi
Matrix	Public Constructor	int r, int c	Membuat matriks kosong dengan dimensi r x c
getRow	Public int		Mengambil jumlah baris dari Matrix
getCol	Public int		Mengambil jumlah kolom dari Matrix
getELMT	Public double	int i, int j)]	Mengambil elemen di baris ke i dan kolom ke j
setELMT	Public void	int i, int j, double val	Menaruh val sebagai elemen di baris ke i dan kolom ke j
setIdentitas	Public void		Mengubah matriks menjadi matriks identitas
setRow	Public void	int row, double[] data	Menaruh suatu data sebagai data dari Sizbaris ke row
readMatrix	Public void		Menerima input untuk mengisi suatu matriks dari keyboard
readMatrixRegresi	Public void		Menerima input untuk mengisi suatu matriks untuk regresi dari keyboard
displayMatrix	Public void		Meng- <i>output</i> matriks ke layar

inputDataFromDouble	Public final void	Double[][] data	Memasukkan data ke dalam matriks
Size	Public int		Mengambil ukuran matriks (jumlah elemen)
operationRow	Public void	Int i1, int i2, double k	Melakukan operasi di baris ke i1 dan baris ke i2 dengan baris ke i1 dikurangi k kali baris ke i2
divRow	Public void	Int I, double k	Melakukan pembagian di baris ke I dengan konstanta k
switchRow	Public void	Int i1, int i2	Melakukan penukaran antara baris ke i1 dan baris ke i2
switchRowEmpty	Public static void	Matrix m	
changeZeroVal	Public static void	Matrix m	
transpose	Public static Matrix	Matrix m	Mengembalikan transpose dari matriks m
getMKoef	Public static Matrix	Matrix m	
createMAug	Public static Matrix	Matrix koef, Matrix cons	
getMConst	Public static Matrix	Matrix m	
Multiply	Public static Matrix	Matrix A, Matrix B	Mengalikan matriks A dan B dengan aturan perkalian matriks
Multiply_altern	Public Matrix	Matrix m1, Matrix m2	Alternatif dari fungsi Multiply
getLeadingOne	Public static int	Matrix M, int row	
stripNonDigits	Public static String[]	String x	
rowOperations	Public void	Int i1, int i2, double k	
isRowEmpty	Public static boolean	Matrix m, int i	Mengembalikan hasil pengecekan apakah baris ke i kosong
isRowZero	Public static boolean	Matrix M, int row	Mengembalikan hasil pengecekan apakah baris ke i berisi 0 semua
isUnderEmpty	Public static boolean	Matrix m, int i, int j	
isDiagonalSatu	Public static boolean	Matrix M	

determinanReduksiBaris	Public static double	Matrix mat	Mengembalikan determinan matriks dengan cara reduksi baris
detKofaktor	Public static double	Matrix M	Mengembalikan determinan matriks dengan cara ekspansi kofaktor
InverseDgnGauss	Public static Matrix	Matrix M	Mengembalikan matriks invers dengan cara Gauss-Jordan
findCofactorMatrix	Public static Matrix	Matrix M	Mengembalikan matriks ekspansi kofaktor dari matriks M
findInverseUsingAdjugate	Public static Matrix	Matrix M	Mengembalikan matriks invers dengan cara Adjoin

II. Class InputOutput

- Attribute

Nama	Tipe	Deskripsi
file	Private scanner	File yang akan di read/write
namaFile	Private string	Nama file dalam string
in	Static Scanner	Input string
sc	Static Scanner	Input integer atau double

- Method

Nama	Tipe	Parameter	Deskripsi
InputOutput	Public Constructor	String namaFile	Mengkonstruksi object InputOutput
stringToMatrix	Public static Matrix	String inp	Mengkonversi string ke matriks
fileToMatrix	Public static Matrix	File inputFile	Mengkonversi masukan dari file ke matrix
stringToFile	Public static void	String inp, String dir	Mengkonversi string ke file di sebuah directory
stringToDoubleArray	Public static double[]	String inp	Mengkonversi masukan dari sebuah string menjadi sebuah array double
openFile	Public void		Membuka suatu file

closeFile	Public void		Menutup suatu file
readRow	Public int		Membaca satu baris dari file
readCol	Public int		Membaca satu kolom dari file
readFile	Public Matrix		Membaca sebuah matriks dari sebuah file
displaySave	Public static void		Memunculkan opsi untuk pilihan
writeMatrix	Public static void	Matrix M, string namaFile	Menulis matriks kedalam file
saveFileInverse	Public static void	Matrix hasil	Menulis matriks inverse ke file
saveFileSPL	Public static void	Matrix hasil	Menulis Solusi SPL ke file
saveFilePolinom	Public static void	String m1, String m2	Menulis Hasil Interpolasi Polinom ke file
saveFileRegresi	Public static void	String m1, String m2	Menulis hasil regresi ke file
saveFileParametric	Public static void	Int N, String[] Eq	Menulis solusi parametrik ke file
saveFile	Public static void	String m	Save suatu string ke file secara general

III. Class InterpolasiBikubik

- Attribute

Class ini tidak memiliki attribute

- Methods

Nama	Tipe	Parameter	Deskripsi
getMatrixX	Public static void	Matrix XVar	Menghasilkan matriks X
getRowX	Public static double[]	Double x, double y	Menghasilkan baris dari matriks X
getMatrixA	Public static Matrix	Matrix Fungsi, Matrix invXVar	Menghasilkan matriks A
getMatrixF	Public static Matrix	Matrix M	Mengubah matriks M menjadi matriks F
getValue	Public static double	Matrix AVar, double x, double y	Menghasilkan fungsi di x, y untuk matriks A tertentu
readFileName	Public static String		Membaca nama file input

readBicubicFromFile	Public static Matrix	String fileName	Membaca matriks fungsi interpolasi bikubik dari file
readFFFromFile	Public static Matrix	String fileName	Membaca matriks F dari file
getX	Public static double	String fileName	Mendapatkan nilai X dari file
getY	Public static double	String fileName	Mendapatkan nilai Y dari file

IV. Class PolynomialInterpolation

- Attribute

Class ini tidak memiliki Attribute

- Methods

Nama	Type	Parameter	Deskripsi
checkChoiceBetweenOneOrTwo	Public static void	Int choice	Mengecek apakah pilihan 1 atau 2
readN	Public static int		Menerima masukan n dari keyboard dan mengembalikan integer n
readFileName	Public static String		Menerima masukan nama file dan mengembalikan string nama file
readPointsFromKeyboard	Public static Matrix		Menerima masukan points dari keyboard
readPointsFromFile	Public static Matrix		Menerima masukan points dari file
createSystemOfLinearEquations	Public static Matrix	Matrix inputMatrix	Membuat matriks yang berisikan SPL yang diperoleh dari points
findSolutionMatrix	Public static Matrix	Matrix systemOfLinearEquations	Mengembalikan matriks berisikan solusi dari SPL

printPolynomialInterpolationEquation	Public static void	Matrix solutionMatrix	Men- <i>output</i> persamaan interpolasi polinom ke layar
interpolate	Public static double	Matrix solutionMatrix, double x	Mengembalikan estimasi dari nilai x yang diinterpolasikan ke interpolasi polinom

V. Class SPL

- Attribute

Class ini tidak memiliki Attribute

- Methods

Nama	Tipe	Parameter	Deskripsi
getRowEchelon	Public static Matrix	Matrix M	Menghasilkan matriks eselon baris
GaussElimination	Public static void	Matrix M	Menghasilkan solusi SPL dengan metode Gauss
getGaussSolutions	Public static void	Matrix M	Menghasilkan solusi SPL
elimGaussJordan	Public static Matrix	Matrix m	Menghasilkan solusi dengan metode eliminasi Gauss-Jordan
inverseSPL	Public static Matrix	Matrix M	Menghasilkan solusi SPL dengan metode invers
Cramer	Public static Matrix	Matrix M	Menghasilkan solusi SPL dengan metode Cramer

VI. Class UserInterface

- Attribute

Class ini tidak memiliki Attribute

- Methods

Nama	Tipe	Parameter	Deskripsi
------	------	-----------	-----------

displayTipeInput	Public static void		Mengeluarkan pilihan tipe input
------------------	--------------------	--	---------------------------------

VII. Class RegresiLinear

- Attribute

Class ini tidak memiliki Attribute

- Methods

Nama	Tipe	Parameter	Deskripsi
regresiGandaSPL	Public static void	Matrix M, Matrix X	Menghasilkan taksiran dari regresi data sampel

BAB IV EKSPERIMEN

I. Studi Kasus Sistem Persamaan Linear

1. a.

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

SPL tidak memiliki solusi

b.

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

Solusi SPL:

$$x_1 = 3.0 + (-1.0 - (-1.0 + p) + 3.0p) - p$$

$$x_2 = -1.0 - (-1.0 + p) + 3.0p$$

$$x_3 = q$$

$$x_4 = -1.0 + p$$

$$x_5 = p$$

c.

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

Solusi SPL:

$$x_1 = q$$

$$x_2 = 2.0 - (1.0 + p)$$

$$x_3 = r$$

$$x_4 = -1.0 - (1.0 + p)$$

$$x_5 = 1.0 + p$$

$$x_6 = p$$

d. (n = 6)

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

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

Solusi SPL:

```
X1 = 35.49563896353709
X2 = -615.8281912431862
X3 = 3265.109507180493
X4 = -7315.032055280091
X5 = 7291.156565240483
X6 = -2666.5483153034925
```

d. (n = 10)

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

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

Solusi SPL:

```
X1 = 35.027450833134026
X2 = -623.8571512143317
X3 = 3852.1744040605336
X4 = -14392.89882482888
X5 = 42154.722722905455
X6 = -83250.61287890965
X7 = 79856.34633187548
X8 = -3670.9170362678706
X9 = -46643.2922405285
X10 = 22696.28222900051
```

2. a.

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

Solusi SPL:

$$X1 = -1.0 + (+2.0p) - 2.0p + q$$

$$X2 = +2.0p$$

$$X3 = p$$

$$X4 = q$$

b.

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

Solusi SPL:

$$X1 = 0.0$$

$$X2 = 2.0$$

$$X3 = 1.0$$

$$X4 = 1.0$$

3. a.

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

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

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

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

Solusi SPL:

$$X1 = -0.2243243243243263$$

$$X2 = 0.1824324324324329$$

$$X3 = 0.7094594594594597$$

$$X4 = -0.2581081081081081$$

b.

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

SPL tidak memiliki solusi

II. Studi Kasus Interpolasi Polinom

a.

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:

$x = 0.2$ $f(x) = ?$
 $x = 0.55$ $f(x) = ?$
 $x = 0.85$ $f(x) = ?$
 $x = 1.28$ $f(x) = ?$

```

Persamaan interpolasi polinomial adalah:
f(x) = - 4212.434532x^6 + 7102.399162x^5 - 4346.313951x^4 + 1220.854891x^3 - 163.915663x^2 + 10.276384x - 0.184559x^0
Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
0.2
Estimasi dari f(0.200000) adalah 0.130000

Persamaan interpolasi polinomial adalah:
f(x) = - 4212.434532x^6 + 7102.399162x^5 - 4346.313951x^4 + 1220.854891x^3 - 163.915663x^2 + 10.276384x - 0.184559x^0
Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
0.55
Estimasi dari f(0.550000) adalah 2.137572

Persamaan interpolasi polinomial adalah:
f(x) = - 4212.434532x^6 + 7102.399162x^5 - 4346.313951x^4 + 1220.854891x^3 - 163.915663x^2 + 10.276384x - 0.184559x^0
Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
0.85
Estimasi dari f(0.850000) adalah -66.269639

```

Persamaan interpolasi polinomial adalah:
 $f(x) = -4212.434532x^6 + 7102.399162x^5 - 4346.313951x^4 + 1220.854891x^3 - 163.915663x^2 + 10.276384x - 0.184559x^0$
 Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
 1.28
 Estimasi dari $f(1.280000)$ adalah -3485.144902

b.

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 16/07/2022 (Tanggal = 7,516)

Persamaan interpolasi polinomial adalah:
 $f(x) = -140993.712249x^9 + 9372849.239101x^8 - 275474539.420669x^7 + 4695806315.428793x^6 - 51131876760.132810x^5 + 368550807175.535000x^4 - 1756810186361.356400x^3 + 5334203055$
 Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
 7.516
 Estimasi dari $f(7.516000)$ adalah 53566.808594

Tanggal 10/08/2022 (Tanggal = 8,323)

Persamaan interpolasi polinomial adalah:
 $f(x) = -140993.712249x^9 + 9372849.239101x^8 - 275474539.420669x^7 + 4695806315.428793x^6 - 51131876760.132810x^5 + 368550807175.535000x^4 - 1756810186361.356400x^3 + 5334203055$
 Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
 8.323
 Estimasi dari $f(8.323000)$ adalah 36331.722656

Tanggal 05/09/2022 (Tanggal = 9,167)

Persamaan interpolasi polinomial adalah:
 $f(x) = -140993.712249x^9 + 9372849.239101x^8 - 275474539.420669x^7 + 4695806315.428793x^6 - 51131876760.132810x^5 + 368550807175.535000x^4 - 1756810186361.356400x^3 + 5334203055$
 Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
 9.167
 Estimasi dari $f(9.167000)$ adalah -667646.218750

Tanggal 01/08/2022 (Tanggal = 8,032)

Persamaan interpolasi polinomial adalah:
 $f(x) = -140993.712249x^9 + 9372849.239101x^8 - 275474539.420669x^7 + 4695806315.428793x^6 - 51131876760.132810x^5 + 368550807175.535000x^4 - 1756810186361.356400x^3 + 5334203055$
 Masukkan nilai x yang ingin diinterpolasi ke dalam persamaan polinomial:
 8.032
 Estimasi dari $f(8.032000)$ adalah 29122.652344

c.

c. 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 $h = (2 - 0)/5 = 0.4$.

Untuk $n = 5$

Persamaan interpolasi polinomial adalah:

$$f(x) = 0.236257x^5 - 1.421266x^4 + 3.237115x^3 - 3.552682x^2 + 2.035257x + 0.000000$$

III. Studi Kasus Interpolasi Bikubik

Diberikan matriks input:

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

Tentukan nilai:

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

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

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

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

$$f(0, 0)$$

Hasil : 161.0

$$f(0.5, 0.5)$$

Hasil : 97.72656249999997

$$f(0.25, 0.25)$$

Hasil : 105.51477050781247

$$f(0.1, 0.9)$$

Hasil : 104.22911850000001

IV. Studi Kasus Regresi Linear 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.

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

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

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

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

```

86.6
29.35
0.94
31.5
76.9
29.63
1.10
10.6
75.4
77.9
29.28
0.87
96.6
78.7
29.29
0.78
54.9
70.9
29.37
0.95
50.0
76.0
29.30
0
>>> Masukkan nilai x yang akan ditaksir:
50
76
29.3

Persamaan Regresi yang terbentuk:
f(x) = -9.424118013412226 + -0.0026919006446863847 X1 + -5.483332505606953E-6 X2 + 0.355817
03608136195 X3

Hasil taksiran regresi adalah: 0.8663093782669335

```


BAB V

PENUTUP

I. Simpulan

1. *Library* telah diimplementasikan dalam program sebagaimana dijelaskan lebih lanjut pada bab III.
2. Program telah digunakan dalam menyelesaikan permasalahan mengenai sistem persamaan linear, interpolasi polinom, interpolasi bikubik, dan regresi linear sebagaimana dijelaskan lebih lanjut pada bab IV.

II. Saran

1. Program dapat di-*improve* dengan meng-*handle* berbagai *input error* untuk mencapai *user experience* yang lebih *seamless*.
2. Dapat diimplementasikan *graphical user interface* (GUI) kepada program untuk meningkatkan lagi *user experience*.

III. Refleksi

Kami belajar banyak dalam tugas ini. Pengalaman kami selama mengerjakan tugas ini cukup memuaskan, tetapi karena ditambahkan tekanan oleh matkul lain, terasa lebih tertekan. Dari segi kekurangan, kerja sama kami dapat di-*improve* sebagai suatu tim. Selain itu, komunikasi antaranggota juga masih kurang optimal sehingga menyebabkan ketidakmaksimalan pada penyelesaian dan pengembangan program.

DAFTAR PUSTAKA

Howard Anton, Dkk. *Elementary Linear Algebra with Applications ninth edition*. USA: John Wiley & Sons, 2005.

<https://docs.oracle.com/javase/6/docs/api/allclasses-noframe.html> (Diakses pada 26 September, 2022)

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-02-Matriks-Eselon.pdf> (Diakses pada 25 September, 2022)

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-03-Sistem-Persamaan-Linear-2.pdf> (Diakses pada 25 September, 2022)

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-05-Sistem-Persamaan-Linear-2.pdf> (Diakses pada 25 September, 2022)

https://www.w3schools.com/java/java_files_create.asp (Diakses pada 29 September, 2022)

<https://stackoverflow.com/questions/4769976/reading-2-d-array-from-a-file> (Diakses pada 29 September, 2022)

<https://codegym.cc/groups/posts/matrix-in-java-2d-arrays> (Diakses pada 30 September, 2022)

https://www.mssc.mu.edu/~daniel/pubs/RoweTalkMSCS_BiCubic.pdf (Diakses pada 3 Oktober, 2022)