Tugas Besar 1 IF 2123 Aljabar Linier dan Geometri
Sistem Persamaan Linier, Determinan, dan Aplikasinya
Semester I Tahun 2020/2021



Shafira Naya Aprisadianti 13519040
Aisyah Farras Aqila 13519054
Ramadhana Bhanuharya Wishnumurti 13519203

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

BAB I

1.1 Deskripsi masalah

Sistem persamaan linier (SPL) Ax = b dengan n peubah (variabel) dan m persamaan adalah berbentuk a11 x1 + a12 x2 + ... + a1n xn = b1 a21 x1 + a22 x2 + ... + a2n xn = b2 : : : : am1 x1 + am2 x2 + ... + amn xn = bm yang dalam hal ini xi adalah peubah, aij dan bi adalah koefisien ∈ R. Sembarang SPL dapat diselesaikan dengan beberapa metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode matriks balikan (x = A -1 b), dan kaidah Cramer (khusus untuk SPL dengan n peubah dan n persamaan). Solusi sebuah SPL mungkin tidak ada, banyak, atau hanya satu (unik/tunggal).

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix}$$

Sebuah matriks M berukuran n × n determinannya adalah

$$\det(M) = \begin{vmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{vmatrix}$$

Determinan matriks M berukuran n × n dapat dihitung dengan beberapa cara: reduksi baris dan ekspansi kofaktor.
SPL memiliki banyak aplikasi dalam bidang sains dan rekayasa, dua diantaranya diterapkan pada tugas besar ini, yaitu interpolasi polinom dan regresi linier.

II. Interpolasi Polinom

Persoalan interpolasi polinom adalah sebagai berikut: Diberikan n+1 buah titik berbeda, (x0, y0), (x1, y1), ..., (xn, yn). Tentukan polinom pn(x) yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga yi = pn(xi) untuk i = 0, 1, 2, …, n. Setelah polinom interpolasi pn(x) ditemukan, pn(x) dapat digunakan untuk menghitung perkiraan nilai y di sembarang titik di dalam selang [x0, xn]. Polinom interpolasi derajat n yang menginterpolasi titik-titik (x0, y0),(x1, y1), ..., (xn, yn). adalah berbentuk pn(x) = a0 + a1x + a2x 2 + … + anx n . Jika hanya ada dua titik, (x0, y0) dan (x1, y1), maka polinom yang menginterpolasi kedua titik tersebut adalah p1(x) = a0 + a1x yaitu berupa persamaan garis lurus. Jika tersedia tiga titik, (x0, y0), (x1, y1), dan (x2, y2), maka polinom yang menginterpolasi ketiga titik tersebut adalah p2(x) = a0 + a1x + a2x 2 atau persaman kuadrat dan kurvanya berupa parabola. Jika tersedia empat titik, (x0, y0), (x1, y1), (x2, y2), dan (x3, y3), polinom yang menginterpolasi keempat titik tersebut adalah p3(x) = a0 + a1x + a2x 2 + a3x 3 , demikian seterusnya. Dengan cara yang sama kita dapat membuat polinom interpolasi berderajat n untuk n yang lebih tinggi asalkan tersedia (n+1) buah titik data. Dengan menyulihkan (xi, yi) ke dalam persamaan polinom pn(x) = a0 + a1x + a2x 2 + … + anx n untuk i = 0, 1, 2, …, n, akan diperoleh n buah sistem persamaan lanjar dalam a0, a1, a2, …, an, a0 + a1x0 + a2x0 2 + ... + an x0 n = y0 a0 + a1x1 + a2x1 2 + ... + an x1 n = y1 ... ... a0 + a1xn + a2xn 2 + ... + an xn n = yn Solusi sistem persamaan lanjar ini, yaitu nilai a0, a1, …, an, diperoleh dengan menggunakan metode eliminasi Gauss yang sudah anda pelajari. Sebagai contoh, misalkan diberikan tiga buah titik yaitu (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513). Tentukan polinom interpolasi kuadratik lalu estimasi nilai fungsi pada x = 9.2. Polinom kuadratik berbentuk p2(x) = a0 + a1x + a2x 2 . Dengan menyulihkan ketiga buah titik data ke dalam polinom tersebut, diperoleh sisten persamaan lanjar yang terbentuk adalah

a0 + 8.0a1 + 64.00a2 = 2.0794

a0 + 9.0a1 + 81.00a2 = 2.1972

a0 + 9.5a1 + 90.25a2 = 2.2513

Penyelesaian sistem persamaan dengan metode eliminasi Gauss menghasilkan a0 = 0.6762, a1 = 0.2266, dan a2 = -0.0064. Polinom interpolasi yang melalui ketiga buah titik tersebut adalah p2(x) = 0.6762 + 0.2266x - 0.0064x 2 . Dengan menggunakan polinom ini, maka nilai fungsi pada x = 9.2 dapat ditaksir sebagai berikut: p2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)2 = 2.2192.

III. Regresi Linier Berganda

Regresi Linear (akan dipelajari lebih lanjut di Probabilitas dan Statistika) merupakan salah satu metode untuk memprediksi nilai selain menggunakan Interpolasi Polinom. Meskipun sudah ada rumus jadi untuk menghitung regresi linear sederhana, terdapat rumus umum dari regresi linear yang bisa digunakan untuk regresi linear berganda, yaitu.

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

Untuk mendapatkan nilai dari setiap βi dapat digunakan Normal Estimation Equation for Multiple Linear Regression sebagai berikut:

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

BAB II

2.1 Teori singkat

1. Metode eliminasi Gauss

   Metode eliminasi Gauss dapat digunakan untuk menemukan solusi pada sistem persamaan linier. Berikut ini langkah-langkah untuk melakukan metode eliminasi Gauss.

   1) Nyatakan SPL dalam bentuk matriks augmented.
   2) Terapkan OBE pada matriks augmented sampai terbentuk matriks eselon baris.
   3) Lakukan teknik substitusi mundur untuk memecahkan persamaan. Teknik substitusi mundur artinya memecahkan persamaan dimulai dari mengambil nilai variabel dari baris terakhir lalu mensubstitusikannya persamaan di baris atasnya, dan seterusnya sampai baris pertama.

2. Metode eliminasi Gauss-Jordan

   Metode eliminasi Gauss-Jordan merupakan pengembangan dari metode eliminasi Gauss. Perbedaannya adalah metode eliminasi Gauss-Jordan berakhir dengan matriks eselon baris tereduksi, sedangkan metode eliminasi Gauss berakhir pada matriks eselon baris. Nilai variabel langsung dapat diperoleh dari matriks augmented akhir sehingga tidak diperlukan teknik subsitusi mundur.

   Metode eliminasi Gauss-Jordan terdiri dari dua fase:

   - Fase pertama yaitu fase maju atau fase eliminasi Gauss, merupakan fase untuk menghasilkan nilai-nilai 0 di bawah 1 utama. Fase ini menghasilkan matriks eselon baris.
   - Fase kedua yaitu fase mundur merupakan fase untuk menghasilkan nilai-nilai 0 di atas 1 utama. Fase ini menghasilkan matriks eselon baris tereduksi.

3. Determinan

   Determinan merupakan suatu unsur nilai yang dapat dihitung dari matriks persegi. Ada dua cara mencari determinan dari sebuah matriks yaitu dengan ekspansi kofaktor dan reduksi baris.

   a. Determinan dengan Ekspansi Kofaktor

      Jika A adalah sebuah matriks persegi, maka minor dari elemen a_{ij} dinotasikan dengan M_{ij} yang merupakan determinan dari submatriks setelah baris ke-i dan kolom ke-j dihilangkan dari A. Besaran $(-1)^{i+j}M_{ij}$ dinotasikan dengan C_{ij} dan dinamakan kofaktor dari elemen a_{ij}.

      Jika A merupakan matriks n x n, maka determinan dari A merupakan penjumlahan dari elemen-elemen suatu baris atau suatu kolom yang dikalikan dengan masing-masing kofaktornya.

   b. Determinan dengan Reduksi Baris

      Mendapatkan determinan dari suatu matriks persegi dengan teknik reduksi baris dilakukan dengan melakukan Operasi Baris Elementer hingga matriks berbentuk atau segitiga atas atau segitiga bawah. Perkalian diagonal utama matriks segitiga atas/bawah tersebut merupakan determinan dari matriks.

      Ada beberapa teorema dalam menentukan matriks dengan metode ini. Jika suatu baris dikalikan oleh k, maka determinan matriks dikalikan oleh k. Jika ada dua baris yang dipertukarkan posisinya, maka determinan matriks dikali -1. Lalu jika suatu baris ditambah k kali baris lainnya, determinan matriks tetap.

4. Matriks balikan

   JIka A dan B matriks bujur sangkar sedemikian rupa sehingga A B = B A = I , maka B disebut balikan atau *invers* dari A dan dapat dituliskan $B = A^{-1}$ (B sama dengan *invers* A). Matriks B juga mempunyai *invers* yaitu A maka dapat dituliskan $A = B^{-1}$. Jika tidak ditemukan matriks B, maka A dikatakan **matriks tunggal** (singular). Jika matriks B dan C adalah *invers* dari A maka B = C.

   Matriks balikan dapat dicari dengan menggunakan ekspansi kofaktor atau dengan eliminasi gauss. Untuk metode ekspansi kofaktor kita mencari determinan lalu mengali dengan adjoin dari matriks yang berupa transpose dari matriks kofaktor.

5. Matriks kofaktor

   Digunakan untuk mencari invers dari matriks yang berdampingan. Kofaktor adalah angka yang didapatkan ketika kita menghapus kolom dan baris dari elemen yang ditentukan dalam matriks, yang hanya berupa grid numerik dalam bentuk persegi panjang atau persegi.

6. Matriks adjoin

   Untuk mencari matriks balikan diperlukan matriks matriks adjoin yang terdiri transpose dari matriks kofaktor.

   If $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$,

   then $adj\,A = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}^T = \begin{bmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{bmatrix}$;

   where C_{ij} denotes the cofactor of a_{ij} in A.

7. Kaidah Cramer

   Kaidah Cramer merupakan salah satu cara untuk menyelesaikan SPL Ax=B dengan syarat SPL tersebut terdiri dari n persamaan linier dan n variabel (A merupakan matriks persegi). Kaidah ini memanfaatkan determinan dalam perhitungannya.

   Jika matriks koefisien A dapat di-invers (det(A) $\neq$ 0), maka SPL tersebut memiliki solusi unik yaitu:

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

   Untuk matriks persegi nxn dengan n>3, biasanya lebih efisien jika menyelesaikan SPL dengan metode eliminasi Gauss-Jordan daripada Kaidah Cramer. Kaidah Cramer ini biasanya digunakan untuk mendapatkan bagian dari solusi dari SPL tanpa menyelesaikan sistem secara keseluruhan.

Misalnya jika hanya ingin mendapatkan variabel x saja dari persamaan dengan 3 variabel yang terdiri dari variabel x, y, dan z.

8. Interpolasi polinom
   Jika sebuah data memiliki ketelitian yang sangat tinggi, kita dapat mendapatkan menemukan sebuah fungsi dari titik-titik yang dipetakan. Dalam kata lain kita menginterpolasi titik-titik data dengan sebuah fungsi. Persamaan dari polinom interpolasi adalah $p_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$. Persamaan dibuat untuk semua titik yang diketahui, lalu diubah dalam bentuk *augmented matrix* dan dicari solusi dari a_0 hingga a_n menggunakan metode eliminasi Gauss.
9. Regresi linier berganda
   Regresi linear berganda adalah regresi linear yang jika jumlah variabel bebas lebih dari satu. Model regresi linear berganda adalah sebagai berikut:

   $$Y = \alpha + \beta 1\ X2 + \beta 2\ X2 + \beta n\ Xn + e$$

   Keterangan:
   Y = Variabel terikat atau response, X = Variabel bebas atau predictor.
   α = Konstanta. β = Slope atau Koefisien estimate.

   Salah satu cara membuat model regresi linear berganda adalah dengan menggunakan normal estimation equation dari

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \qquad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}$$

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}, \qquad \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

Dan mengubahnya menjadi bentuk, dengan bantuan rekayasa vektor dan matriks

$$\begin{array}{cccccc} n\hat{\beta}_0 & +\hat{\beta}_1 \sum_{i=1}^{n} x_{i1} & +\hat{\beta}_2 \sum_{i=1}^{n} x_{i2} & +\cdots & +\hat{\beta}_k \sum_{i=1}^{n} x_{ik} & = \sum_{i=1}^{n} y_i \\ \hat{\beta}_0 \sum_{i=1}^{n} x_{i1} & +\hat{\beta}_1 \sum_{i=1}^{n} x_{i1}^2 & +\hat{\beta}_2 \sum_{i=1}^{n} x_{i1}x_{i2} & +\cdots & +\hat{\beta}_k \sum_{i=1}^{n} x_{i1}x_{ik} & = \sum_{i=1}^{n} x_{i1}y_i \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \hat{\beta}_0 \sum_{i=1}^{n} x_{ik} & +\hat{\beta}_1 \sum_{i=1}^{n} x_{ik}x_{i1} & +\hat{\beta}_2 \sum_{i=1}^{n} x_{ik}x_{i2} & +\cdots & +\hat{\beta}_k \sum_{i=1}^{n} x_{ik}^2 & = \sum_{i=1}^{n} x_{ik}y_i \end{array}$$

BAB III

3.1 Implementasi program dalam Java

1. Sistem Persamaan Linier
    a. Gauss
        i. void Gauss(double[][] a, double[] b) : prosedur eliminasi Gauss
        ii. void MetodeGauss(): prosedur penyelesaian SPL dengan metode eliminasi Gauss, mulai dari input hingga output.
    b. Gauss Jordan
        i. void GaussJordan(double[][] a, double[] b): prosedur eliminasi Gauss Jordan
        ii. void MetodeGaussJordan(): prosedur penyelesaian SPL dengan metode eliminasi Gauss Jordan, mulai dari input hingga output.
    c. Matriks Balikan
        i. void MetodeMatriksBalikan(): prosedur penyelesaian SPL dengan metode matriks balikan (perkalian antara matriks balikan A dan b), mulai dari input hingga output.
    d. Kaidah Cramer
        i. double Cramer(double[][] A, double[] b, int var): mengembalikan solusi SPL dari salah satu variabel dengan kaidah Cramer.
    e. Lainnya
        i. void matriksHilbert(): membentuk matriks augmented matriks Hilbert beserta b yang sesuai di soal [1,0,..,0]. Prosedur ini hanya digunakan untuk keperluan testing tapi tidak dijalankan di program utama.
        ii. void backSubstitution(double[][] a, double[] b): prosedur untuk melakukan substitusi mundur
        iii. void cekBarisTerakhir (double[][] a, double[] b): prosedur untuk mengecek baris terakhir matriks augmented untuk menentukan tipe solusi

2. Determinan
    a. double detCofactor(double matrix[][], int n):
       Mengembalikan determinan matriks dengan metode ekspansi kofaktor.
    b. double detRowReduction(double matrix[][], int n):
       Mengembalikan determinan matriks dengan metode reduksi baris.
3. Matriks Balikan
    a. Double Determinan : mencari determinan dari metode kofaktor.
    b. Void Kofaktor : metode ekspansi kofaktor.
    c. Double[][] adjoint : membentuk matriks adjoint.
    d. Double[][] gaussian : mencari balikan matriks menggunakan eliminasi gauss.

4. Interpolasi Polinom
    a. void intPol(double matrix[][], int n, double x):

Mengeluarkan fungsi polinom interpolasi dan hasil dari fungsi sesuai dengan masukan x.

5. Regresi Linier Berganda
   a. Void normalEq(double aug[], double X[], double Y[], int brs, int kol, int nk)
      Mengubah input menjadi dalam bentuk SPL normal estimation equation.
   b. Solvespl : menyelasaikan persamaan normal equation estimation untuk mencari nilainya dengan elminasi gauss
   c. sumupX, sumupY, sumupXX, sumupXY : fungsi membantu dalam membuat noemal estimation equation.

BAB IV

4.1 Eksperimen

1. SPL Ax=B
    a. A berukuran 4x4

Gauss

```
Masukkan jumlah persamaan = 4
Masukkan jumlah peubah = 4
Input matriks 4x4:
1 1 -1 -1
2 5 -7 -5
2 -1 1 3
5 2 -4 2
Input b:
b[1]: 1
b[2]: -2
b[3]: 4
b[4]: 6
Solusi Gauss adalah:
Tidak ada solusi
```

Gauss Jordan

```
Masukkan jumlah persamaan = 4
Masukkan jumlah peubah = 4
Input matriks 4x4:
1 1 -1 -1
2 5 -7 -5
2 -1 1 3
5 2 -4 2
Input b:
b[1]: 1
b[2]: -2
b[3]: 4
b[4]: 6
Solusi Gauss Jordan adalah:
Tidak ada solusi
```

Matriks balikan

```
Masukkan nilai n = 4
Input matriks 4x4:
1 1 -1 -1
2 5 -7 -5
2 -1 1 3
5 2 -4 2
Input b:
b[1]: 1
b[2]: -2
b[3]: 4
b[4]: 6
Tidak dapat dihitung dengan metode Invers karena matriks tidak mempunyai balikan
```

Kaidah Cramer

```
Masukkan nilai n = 4
Input matriks 4x4:
1 1 -1 -1
2 5 -7 -5
2 -1 1 3
5 2 -4 2
Input b:
b[1]: 1
b[2]: -2
b[3]: 4
b[4]: 6
Matriks augmented yang dimasukkan:
1.0 1.0 -1.0 -1.0 1.0
2.0 5.0 -7.0 -5.0 -2.0
2.0 -1.0 1.0 3.0 4.0
5.0 2.0 -4.0 2.0 6.0
Tidak dapat dihitung dengan Kaidah Cramer
```

b. A berukuran 4x5
   Gauss
   Gauss Jordan
   Solusi banyak (parametrik)
c. A berukuran 3x6
   Gauss
   Gauss Jordan
   Solusi banyak (parametrik)
d. Matriks Hilbert
   Matriks hilbert dengan n=6
   Gauss

```
Masukkan nilai n = 6
Matriks Hilbert dengan n = 6 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
Solusi Gauss adalah:
x[1]: 36.00000000071492
x[2]: -630.0000000212816
x[3]: 3360.0000001474646
x[4]: -7560.000000389726
x[5]: 7560.000000435156
x[6]: -2772.0000001729727
```

   Gauss Jordan

```
Masukkan nilai n = 6
Matriks Hilbert dengan n = 6 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
Solusi Gauss Jordan adalah:
x[1]: 36.00000000071624
x[2]: -630.0000000212908
x[3]: 3360.0000001475005
x[4]: -7560.00000038978
x[5]: 7560.000000435184
x[6]: -2772.0000001729727
```

Matriks balikan

```
Masukkan nilai n = 6
Matriks Hilbert dengan n = 6 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
Solusi:
x[1] = 35.99999849275147
x[2] = -629.9999497041402
x[3] = 3359.9997099521784
x[4] = -7559.999411026779
x[5] = 7559.999515511824
x[6] = -2771.9998638224647
```

Cramer

```
Masukkan nilai n = 6
Matriks Hilbert dengan n = 6 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
Solusi:
x[1] = 35.99999849275147
x[2] = -629.9999497041402
x[3] = 3359.9997099521784
x[4] = -7559.999411026779
x[5] = 7559.999515511824
x[6] = -2771.9998638224647
```

Matriks hilbert dengan n=10

Gauss

```
Masukkan nilai n = 10
Matriks Hilbert dengan n = 10 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.066666666666666
0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.0
0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0
0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0555555555555
0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.05555555555555555 0.05263157894
Solusi Gauss adalah:
x[1]: 99.99771492884611
x[2]: -4949.802359000344
x[3]: 79195.78687403111
x[4]: -600561.6733510367
x[5]: 2522337.094138327
x[6]: -6305796.985462033
x[7]: 9608774.43664348
x[8]: -8750801.980572635
x[9]: 4375380.959054107
x[10]: -923687.8310730107
```

Gauss Jordan

```
Masukkan nilai n = 10
Matriks Hilbert dengan n = 10 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666
0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.0
0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0
0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0555555555555
0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.05555555555555555 0.052631578947
Solusi Gauss Jordan adalah:
x[1]: 99.99782625811696
x[2]: -4949.812378549992
x[3]: 79196.00729054434
x[4]: -600563.7304030529
x[5]: 2522347.1213652194
x[6]: -6305825.059131507
x[7]: 9608821.221904255
x[8]: -8750847.807093533
x[9]: 4375405.302407374
x[10]: -923693.2402909148
```

Matriks balikan

```
Masukkan nilai n = 10
Matriks Hilbert dengan n = 10 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666
0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.0
0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0
0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0555555555555
0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.05555555555555555 0.052631578947
```

Cramer

```
Masukkan nilai n = 10
Matriks Hilbert dengan n = 10 beserta b:
1.0 0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 1.0
0.5 0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.0
0.3333333333333333 0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.0
0.25 0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.0
0.2 0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.0
0.16666666666666666 0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666
0.14285714285714285 0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.0
0.125 0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0
0.1111111111111111 0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.0555555555555
0.1 0.09090909090909091 0.08333333333333333 0.07692307692307693 0.07142857142857142 0.06666666666666667 0.0625 0.058823529411764705 0.05555555555555555 0.052631578947
Solusi:
x[1] = 34.59017767772993
x[2] = -381.54024790908835
x[3] = 0.37562228166023426
x[4] = 5698.242961320487
x[5] = -9412.043303205875
x[6] = -3176.6755953264324
x[7] = 7577.607165479928
x[8] = 5393.857417158976
x[9] = -2806.7100611943056
x[10] = -3005.422213180605
```

Pada matriks Hilbert dengan n=6, keempat metode dapat menghitung solusi dengan tepat. Namun, pada matriks Hilbert dengan n=10 hanya tepat bila dihitung dengan metode Gauss dan Gauss Jordan. Saat dihitung dengan metode kaidah Cramer hasilnya tidak tepat, sedangkan dengan metode matriks balikan tidak mengeluarkan solusi sama sekali.

2. SPL matrix augmented
    a. 4x5

        Gauss

        -

        Gauss Jordan

        -

        Matriks balikan

```
Masukkan nilai n = 4
Input matriks 4x4:
1 -1 2 -1
2 1 -2 -2
-1 2 -4 1
3 0 0 -3
Input b:
b[1]: -1
b[2]: -2
b[3]: 1
b[4]: -3
Tidak dapat dihitung dengan metode Invers karena matriks tidak mempunyai balikan
```

Cramer

```
Masukkan nilai n = 4
Input matriks 4x4:
1 -1 2 -1
2 1 -2 -2
-1 2 -4 1
3 0 0 -3
Input b:
b[1]: -1
b[2]: -2
b[3]: 1
b[4]: -3
Tidak dapat dihitung dengan Kaidah Cramer
```

Solusi parametrik yang tepat adalah:

Solution set:

$x_1 = -1$
$x_2 = 2x_3$
$x_4 = 0$

x_3 - free

b. 6x5

Gauss

```
Masukkan jumlah persamaan = 6
Masukkan jumlah peubah = 4
Input matriks 6x4:
2 0 8 0
0 1 0 4
-4 0 6 0
0 -2 0 3
2 0 -4 0
0 1 0 -2
Input b:
b[1]: 8
b[2]: 6
b[3]: 6
b[4]: -1
b[5]: -4
b[6]: 0
Solusi Gauss adalah:
x[1]: 0.0
x[2]: 2.0
x[3]: 1.0
x[4]: 1.0
x[5]: 0.0
x[6]: 0.0
```

Gauss Jordan

```
Masukkan jumlah persamaan = 6
Masukkan jumlah peubah = 4
Input matriks 6x4:
2 0 8 0
0 1 0 4
-4 0 6 0
0 -2 0 3
2 0 -4 0
0 1 0 -2
Input b:
b[1]: 8
b[2]: 6
b[3]: 6
b[4]: -1
b[5]: -4
b[6]: 0
Solusi Gauss Jordan adalah:
x[1]: 0.0
x[2]: 2.0
x[3]: 1.0
x[4]: 1.0
x[5]: 0.0
x[6]: 0.0
```

3. SPL
    a. SPL dengan 4 variabel dan 4 persamaan
       Gauss

```
Masukkan jumlah persamaan = 4
Masukkan jumlah peubah = 4
Input matriks 4x4:
8 1 3 2
2 9 -1 -2
1 3 2 -1
1 0 6 4
Input b:
b[1]: 0
b[2]: 1
b[3]: 2
b[4]: 3
Solusi Gauss adalah:
x[1]: -0.22432432432432428
x[2]: 0.1824324324324324
x[3]: 0.7094594594594594
x[4]: -0.25810810810810814
```

Gauss Jordan

```
Masukkan jumlah persamaan = 4
Masukkan jumlah peubah = 4
Input matriks 4x4:
8 1 3 2
2 9 -1 -2
1 3 2 -1
1 8 6 4
Input b:
b[1]: 0
b[2]: 1
b[3]: 2
b[4]: 3
Solusi Gauss Jordan adalah:
x[1]: -0.15263157894736842
x[2]: 0.11842105263157893
x[3]: 0.6710526315789473
x[4]: -0.4552631578947368
```

Matriks balikan

```
Masukkan nilai n = 4
Input matriks 4x4:
8 1 3 2
2 9 -1 -2
1 3 2 -1
1 0 6 4
Input b:
b[1]: 0
b[2]: 1
b[3]: 2
b[4]: 3
Solusi:
x[1] = -0.22432432432432434
x[2] = 0.18243243243243243
x[3] = 0.7094594594594594
x[4] = -0.25810810810810814
```

Cramer

```
Masukkan nilai n = 4
Input matriks 4x4:
8 1 3 2
2 9 -1 -2
1 3 2 -1
1 0 6 4
Input b:
b[1]: 0
b[2]: 1
b[3]: 2
b[4]: 3
Solusi:
x[1] = -0.22432432432432434
x[2] = 0.18243243243243243
x[3] = 0.7094594594594594
x[4] = -0.2581081081081081
```

b. SPL dengan 9 variabel dan 12 persamaan

Gauss

```
Masukkan jumlah persamaan = 12
Masukkan jumlah peubah = 9
Input matriks 12x9:
0 0 0 0 0 0 1 1 1
0 0 0 1 1 1 0 0 0
1 1 1 0 0 0 0 0 0
0 0 0.04289 0 0.04289 0.75 0.04289 0.75 0.61396
0 0.25 0.91421 0.25 0.91421 0.25 0.91421 0.25 0
0.61396 0.75 0.04289 0.75 0.04289 0 0.04289 0 0
0 0 1 0 0 1 0 0 1
0 1 0 0 1 0 0 1 0
1 0 0 1 0 0 1 0 0
0.04289 0.75 0.61396 0 0.04289 0.75 0 0 0.04289
0.91421 0.25 0 0.25 0.91421 0.25 0 0.25 0.91421
0.04289 0 0 0.75 0.04289 0 0.61396 0.75 0.04289
Input b:
b[1]: 13.00
b[2]: 15.00
b[3]: 8.00
b[4]: 14.79
b[5]: 14.31
b[6]: 3.81
b[7]: 18.00
b[8]: 12.00
b[9]: 6.00
b[10]: 10.51
b[11]: 16.13
b[12]: 7.04
Solusi Gauss adalah:
Solusi banyak (in progress)
```

Gauss Jordan

-

4. Analisis Rangkaian
5. Interpolasi Polinom

   Pada kasus ini didapatkan polinom interpolasi berderajat 6 dari 7 data yaitu:
   $-0.022977+0.240000x+0.197396x^2+0.000000x^3+0.026042x^4+0.000000x^5-0.000000x^6$.

   a. f(0.2) = 0.03296093750000002

```
n:
7
Matriks:
0.1 0.003
0.3 0.067
0.5 0.148
0.7 0.248
0.9 0.370
1.1 0.518
1.3 0.697
x:
0.2
Persamaan polinom:
-0.022977+0.240000x+0.197396x^2+0.000000x^3+0.026042x^4+0.000000x^5-0.000000x^6

Estimasi nilai fungsi:
0.03296093750000002
```

   b. f(0.55) = 0.17111865234374998

```
n:
7
Matriks:
0.1 0.003
0.3 0.067
0.5 0.148
0.7 0.248
0.9 0.370
1.1 0.518
1.3 0.697
x:
0.55
Persamaan polinom:
-0.022977+0.240000x+0.197396x^2+0.000000x^3+0.026042x^4+0.000000x^5-0.000000x^6

Estimasi nilai fungsi:
0.17111865234374998
```

c. f(0.85) = 0.33723583984375005

```
n:
7
Matriks:
0.1 0.003
0.3 0.067
0.5 0.148
0.7 0.248
0.9 0.370
1.1 0.518
1.3 0.697
x:
0.85
Persamaan polinom:
-0.022977+0.240000x+0.197396x^2+0.000000x^3+0.026042x^4+0.000000x^5-0.000000x^6

Estimasi nilai fungsi:
0.33723583984375005
```

d. f(1.28) = 0.6775418375000001

```
n:
7
Matriks:
0.1 0.003
0.3 0.067
0.5 0.148
0.7 0.248
0.9 0.370
1.1 0.518
1.3 0.697
x:
1.28
Persamaan polinom:
-0.022977+0.240000x+0.197396x^2+0.000000x^3+0.026042x^4+0.000000x^5-0.000000x^6

Estimasi nilai fungsi:
0.6775418375000001
```

6. Interpolasi Polinom untuk Memprediksi Jumlah Kasus Covid-19 di Indonesia
   Pada kasus ini didapat polinom interpolasi berderajat 9 dari 10 data yaitu: $227096397.610844 - 415842670.299822x + 318150082.987350x^2 - 136003290.729656x^3 + 36176040.522150x^4 - 6249554.706972x^5 + 704212.301690x^6 - 50061.956074x^7 + 2041.919782x^8 - 36.470958x^9$.
   a. 25/05/20 = 5,806 (tanggal desimal)
      Pada tanggal ini perkiraan jumlah kasus adalah 22.795 (setelah dibulatkan).

```
n:
10
Matriks:
4.800 8211
5.000 10118
5.516 17025
5.710 20796
6.500 39294
7.194 64958
8.097 113134
8.258 123503
9.033 177571
9.333 145510
x:
5.806
Persamaan polinom:
227096397.610844-415842670.299822x+318150082.987350x^2-136003290.729656x^3+36176040.522150x^4
-6249554.706972x^5+704212.301690x^6-50061.956074x^7+2041.919782x^8-36.470958x^9

Estimasi nilai fungsi:
22794.691296577454
```

b. 30/03/20 = 8,968 (tanggal desimal)
   Pada tanggal ini perkiraan jumlah kasus adalah 183.022 (setelah dibulatkan).

```
n:
10
Matriks:
4.800 8211
5.000 10118
5.516 17025
5.710 20796
6.500 39294
7.194 64958
8.097 113134
8.258 123503
9.033 177571
9.333 145.510
x:
8.968
Persamaan polinom:
4957239869.859171-7099103871.949084x+4490627816.681318x^2-1646664026.619259x^3+385707648.0476
79x^4-59845739.577599x^5+6150517.556139x^6-403729.443874x^7+15359.341857x^8-258.029187x^9

Estimasi nilai fungsi:
183022.4394378662
```

c. 15/09/20 = 9,500 (tanggal desimal)
   Pada tanggal ini perkiraan jumlah kasus adalah 68.216 (setelah dibulatkan).

```
n:
10
Matriks:
4.800 8211
5.000 10118
5.516 17025
5.710 20796
6.500 39294
7.194 64958
8.097 113134
8.258 123503
9.033 177571
9.333 145510
x:
9.5
Persamaan polinom:
227096397.610844-415842670.299822x+318150082.987350x^2-136003290.729656x^3+36176040.522150x^4
-6249554.706972x^5+704212.301690x^6-50061.956074x^7+2041.919782x^8-36.470958x^9

Estimasi nilai fungsi:
68216.42846679688
```

d. 21/05/20 = 5,677 (tanggal desimal)
   Pada tanggal ini perkiraan jumlah kasus adalah 20.126 (setelah dibulatkan).

```
n:
10
Matriks:
4.800 8211
5.000 10118
5.516 17025
5.710 20796
6.500 39294
7.194 64958
8.097 113134
8.258 123503
9.033 177571
9.333 145510
x:
5.677
Persamaan polinom:
227096397.610844-415842670.299822x+318150082.987350x^2-136003290.729656x^3+36176040.522150x^4
-6249554.706972x^5+704212.301690x^6-50061.956074x^7+2041.919782x^8-36.470958x^9

Estimasi nilai fungsi:
20125.8361736238
```

7. Interpolasi Polinom untuk Menyederhanakan Fungsi
   Akan dicari fungsi polinom berderajat 5 sehingga dibutuhkan 6 data, yaitu: (0, 0), (0.4, 0.4189), (0.8, 0.5071), (1.2, 0.5609), (1.6, 0.5837), dan (2, 0.5766). Lalu didapat persamaan polinom yaitu $0.000000 + 2.035904x - 3.555182x^2 + 3.240104x^3 - 1.422689x^4 + 0.236491x^5$. Setelah itu didapat $f(1.5) = 0.5835382568359349$.

```
n:
6
Matriks:
0 0
0.4 0.4189
0.8 0.5071
1.2 0.5609
1.6 0.5837
2 0.5766
x:
1.5
Persamaan polinom:
0.000000+2.035904x-3.555182x^2+3.240104x^3-1.422689x^4+0.236491x^5

Estimasi nilai fungsi:
0.5835382568359349
```

8. Regresi Linear Berganda
   Dicari prediksi dari Nitrous Oxide apabila Humidity bernilai 50%, temperatur 76°F, dan tekanan udara sebesar 29.30. Pertama mengubah menjadi bentuk normal estimation equation supaya bisa didapat nilai dari B0,B1, .. Bn. Didapatkan nilai nitrous oxide berupa 0.9384342262215828

```
Bentuk normal equation for multiple linear regression :

20.0  863.0999999999999  1530.4000000000003  587.8399999999999  19.42
863.0999999999999  54876.89  67000.09  25283.395  779.4769999999999
1530.4000000000003  67000.09  117912.32000000002  44976.866999999984  1483.4369999999997
587.8399999999999  25283.395  44976.866999999984  17278.508600000005  571.1219000000001
```

```
hasilnya adalah:
B0 = -3.507778140882919
B1 = -0.0026249907458783684
B2 = 7.989410472204216E-4
B3 = 0.1541550301982822
```

```
Prediksi dengan data :
y = B0 + B1x11 + B2x21 + B3x31

Masukan nilai humidity :
50
Masukan nilai temperature :
76
Masukan nilai pressure :
29.3
Nilai Prediksi adalah:
0.9384342262215828
```

Bab V

5.1 Kesimpulan

Program menjalankan beberapa operasi Aljabar Geometri yaitu menyelesaikan Sistem Persamaan Linier, mencari determinan matriks, mencari matriks balikan, mencari polinom interpolasi, dan regresi linier.

5.2 Saran

Sebaiknya dipelajari lebih dalam lagi mengenai pemrograman dengan Java dan koordinasi kelompok dilaksanakan lebih baik.

5.3 Refleksi

Program tidak dapat menerima masukan dari file karena terdapat kesalahan dalam prosedur bacaFile(). Prosedur tersebut tidak dapat menemukan file yang dituju sehingga perlu dipelajari lagi mengenai pemrograman dengan Java. Program tidak dapat memenuhi semua kasus pada testcase dengan metode Gauss dan Gauss Jordan. Perlu dibuat algoritma yang dapat memenuhi semua kemungkinan kasus. Untuk fungsi regresi linear berganda hanya berdasarkan data yang diberikan pada soal, kedepannya bisa diubah sedikit dari program yang ada supaya bisa dipakai pada data dan model regresi linear berganda yang lain.

Daftar Referensi

i) Anton, Howard dan Chris Rorres. 2013. Elementary Linear Algebra: Applications Version, 11th Edition. Kanada: Wiley.
ii) https://informatika.stei.itb.ac.id/~rinaldi.munir/MetNum/2010-2011/Interpolasi%20Polinom.pdf
iii) http://mezeylab.cb.bscb.cornell.edu/labmembers/documents/supplement%205%20-%20multiple%20regression.pdf