IMPLEMENTASI OPERASI DASAR MATRIKS DALAM BAHASA PEMROGRAMAN *JAVA* UNTUK MENYELESAIKAN BERAGAM PERSOALAN

Disusun untuk memenuhi laporan tugas besar mata kuliah Aljabar Linier dan Geometri semester 3 di Institut Teknologi Bandung.

Disusun oleh kelompok 13:

Nigel Sahl (13521043)

Ghazi Akmal Fauzan (13521058)

Ryan Samuel Chandra (13521140)

TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

Jl. Ganesa No. 10, Lb. Siliwangi, Kecamatan Coblong,
Kota Bandung, Jawa Barat, 40132

2022

PRAKATA

Puji dan syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena berkat-Nya begitu melimpah dalam kehidupan kami. Akhirnya, kami dapat menyelesaikan laporan ini tepat waktu setelah hampir 3 minggu penuh melakukan kerja sama pemrograman secara rutin. Kami juga mengucapkan terima kasih kepada pihak ITB, khususnya Bapak Dr. Ir. Rinaldi Munir, M.T., serta seluruh Kakak-Kakak asisten mata kuliah Aljabar Linier dan Geometri, yang telah memberikan kami kesempatan untuk mengerjakan tugas ini.

Berikut disajikan laporan lengkap tentang “Implementasi Operasi Dasar Matriks dalam Bahasa Pemrograman *Java* untuk Menyelesaikan Beragam Persoalan”. Laporan ini selain dibuat agar dapat bermanfaat bagi masyarakat, juga secara khusus disusun untuk memenuhi salah satu tugas besar dalam mata kuliah Aljabar Linier dan Geometri di semester 3 Teknik Informatika Institut Teknologi Bandung (IF ITB).

Dengan semua informasi yang terkumpul dari berbagai sumber serta percobaan, kami berusaha sebaik mungkin untuk memenuhi tujuan pembuatan yang telah ditetapkan di tengah hambatan lingkungan dan tekanan waktu.

Kami berharap laporan ini dapat menjadi sesuatu yang memuaskan bagi semua pembaca. Tetapi kami menyadari bahwa laporan ini masih jauh dari kesempurnaan. Untuk itu, kami mohon kritik dan saran yang bersifat konstruktif untuk perkembangan percobaan kami.

Bandung, 3 Oktober 2022

Penyusun Laporan

Daftar Isi

BAB 1

PENDAHULUAN

1.1 Latar Belakang

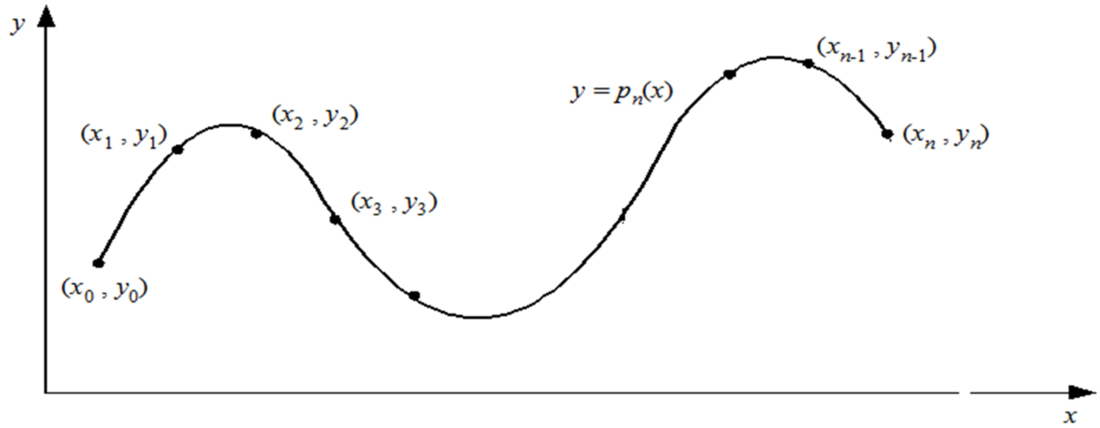
Sistem persamaan linier (SPL) banyak ditemukan di dalam bidang sains dan rekayasa. Berbagai metode untuk menyelesaikan SPL sudah diteliti dan dipelajari, termasuk menghitung determinan matriks. Sembarang SPL dapat diselesaikan dengan beberapa metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan, 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 (tidak berhingga), atau hanya satu (unik/tunggal).

Dalam tugas besar kali ini, akan dibuat beberapa *library* aljabar linier dalam bahasa pemrograman Java. *Library* tersebut berisi fungsi-fungsi seperti eliminasi Gauss, eliminasi Gauss-Jordan, menentukan balikan matriks, menghitung determinan, hingga kaidah Cramer (khusus untuk SPL dengan n peubah dan n persamaan). Selanjutnya, *library* tersebut digunakan dalam program Java untuk menyelesaikan berbagai persoalan yang dimodelkan dalam bentuk SPL, menyelesaikan persoalan interpolasi, dan persoalan regresi. Penjelasan tentang interpolasi dan regresi akan dijelaskan pada bagian lain laporan ini.

1.2 Deskripsi Permasalahan

2.3.1 Interpolasi Polinom

Persoalan interpolasi polinom adalah sebagai berikut: Diberikan n+1 buah titik berbeda, (x_0, y_0), (x_1, y_1), ..., (x_n, y_n). Tentukan polinom $p_n(x)$ yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga $y_i = p_n(x_i)$ untuk $i = 0, 1, 2, \ldots, n$.



Setelah polinom interpolasi $p_n(x)$ ditemukan, $p_n(x)$ dapat digunakan untuk menghitung perkiraan nilai y di sembarang titik di dalam selang $[x_0, x_n]$.

Polinom interpolasi derajat n yang menginterpolasi titik-titik (x_0, y_0), (x_1, y_1), ..., (x_n, y_n). adalah berbentuk $p_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$. Jika hanya ada dua titik, (x_0, y_0) dan (x_1, y_1), maka polinom yang menginterpolasi kedua titik tersebut adalah $p_1(x) = a_0 + a_1x$ yaitu berupa persamaan garis lurus. Jika tersedia tiga titik, (x_0, y_0), (x_1, y_1), dan (x_2, y_2), maka polinom yang menginterpolasi ketiga titik tersebut adalah $p_2(x) = a_0 + a_1x + a_2x^2$ atau persaman kuadrat dan kurvanya berupa parabola. Jika tersedia empat titik, (x_0, y_0), (x_1, y_1), (x_2, y_2), dan (x_3, y_3), polinom yang menginterpolasi keempat titik tersebut adalah $p_3(x) = a_0 + a_1x + a_2x^2 + a_3x^3$, 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 (x_i, y_i) ke dalam persamaan polinom $p_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$ untuk i = 0, 1, 2, …, n, akan diperoleh n buah sistem persamaan lanjar dalam a_0, a_1, a_2, …, a_n :

$$
\begin{gathered}
a_0 + a_1x_0 + a_2x_0^2 + \ldots + a_n x_0^n = y_0 \\
a_0 + a_1x_1 + a_2x_1^2 + \ldots + a_n x_1^n = y_1 \\
\ldots \ldots \\
a_0 + a_1x_n + a_2x_n^2 + \ldots + a_n x_n^n = y_n
\end{gathered}
$$

Solusi sistem persamaan lanjar ini, yaitu nilai a_0, a_1, …, a_n, dapat diperoleh dengan menggunakan metode eliminasi Gauss. 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 $p_2(x) = a_0 + a_1x + a_2x^2$. Dengan menyulihkan ketiga buah titik data ke dalam polinom tersebut, sistem persamaan lanjar yang terbentuk adalah :

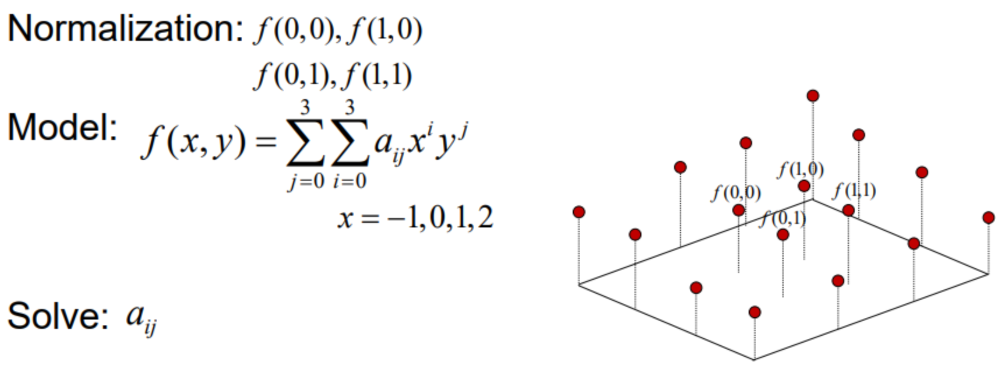
$$
\begin{gathered}
a_0 + 8.0a_1 + 64.00a_2 = 2.0794 \\
a_0 + 9.0a_1 + 81.00a_2 = 2.1972 \\
a_0 + 9.5a_1 + 90.25a_2 = 2.2513
\end{gathered}
$$

Penyelesaian sistem persamaan dengan metode eliminasi Gauss menghasilkan a_0 = 0.6762, a_1 = 0.2266, dan a_2 = -0.0064. Polinom interpolasi yang melalui ketiga buah titik tersebut adalah $p_2(x) = 0.6762 + 0.2266x - 0.0064x^2$. Dengan menggunakan 2 polinom ini, maka nilai fungsi pada x = 9.2 dapat ditaksir sebagai berikut: $p_2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)^2 = 2.2192$.

2.3.2 Interpolasi Bicubic

Bicubic interpolation merupakan teknik interpolasi pada data 2D, umumnya digunakan dalam pembesaran citra yang merupakan pengembangan dari interpolasi linear dan cubic yang telah dipelajari pada kuliah metode numerik di aljabar geometri.

Diberikan sebuah matrix awal, misal M, kita akan mencari persamaan interpolasi *f(x,y)* dengan pemodelan sebagai berikut:



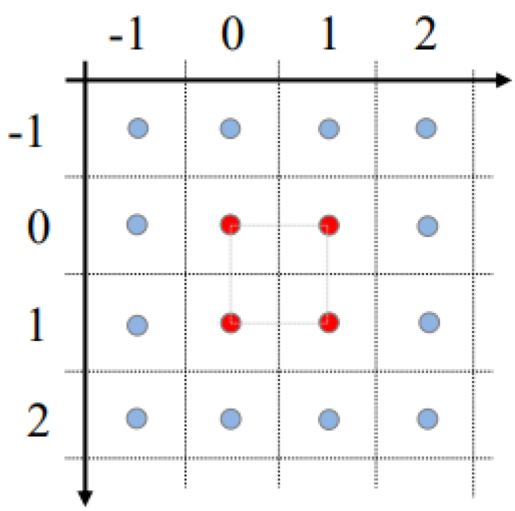
Melakukan substitusi nilai-nilai diketahui pada matriks 4×4 tersebut ke persamaan f(x, y) akan menghasilkan sebuah matriks persamaan:

$$y = Xa$$

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

Elemen pada matrix X adalah koefisien a_{ij} yang diperoleh dari persamaan $f(x, y)$ di atas. Sebagai contoh, elemen pada baris 4 kolom ke 10 adalah koefisien dari a_{12} dan diperoleh dari $2^1 * (-1)^2 = 2$, sesuai persamaan $x^i * y^j$.

Vektor a dapat dicari dari persamaan tersebut (menggunakan inverse), lalu vektor a digunakan sebagai nilai variabel dalam $f(x,y)$. Sehingga terbentuk fungsi interpolasi bicubic sesuai model. Tugasnya adalah menentukan persamaan $f(x,y)$ lalu melakukan interpolasi berdasarkan $f(A,b)$ dari masukan matriks 4×4. Nilai masukan A dan b dalam rentang [0,1] (Referensi gambar di bawah, nilai untuk diinterpolasi dalam kotak merah).



2.3.3 Regresi Linier Berganda

Regresi linear (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:

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan eliminasi Gauss.

BAB 2

TINJAUAN PUSTAKA

2.1 Metode Eliminasi Gauss

Metode eliminasi Gauss adalah suatu metode untuk mencari himpunan penyelesaian dari Sistem Persamaan Linier (SPL) dengan menggunakan OBE (Operasi Baris Elementer), sedemikian hingga matriksnya memiliki bentuk eselon baris. Selanjutnya, matriks tersebut diubah kembali ke dalam bentuk sistem persamaan linier dan kemudian dilakukan substitusi balik mulai dari persamaan paling bawah.

Pertama-tama, harus diketahui dulu definisi dari matriks eselon baris, serta OBE. Matriks eselon baris adalah matriks yang setiap barisnya memiliki satu utama (*leading one*), kecuali baris yang seluruhnya nol. Berikut beberapa aturan terkait matriks eselon baris:

1. Jika baris dalam matriks tidak terdiri seluruhnya dari nol, maka bilangan tak nol pertama dalam baris tersebut adalah 1. Kita menamakan ini sebagai satu utama.
2. Jika terdapat baris yang seluruh entri atau elemennya terdiri dari nol, maka semua baris dengan elemen-elemennya yang berupa nol tersebut dikelompokkan bersama-sama di bawah matriks.
3. Untuk dua baris berurutan yang seluruh elemennya tidak terdiri dari nol, maka satu utama dalam baris yang lebih rendah diletakkan lebih jauh ke kanan dari satu utama dalam baris yang lebih tinggi.

$$A = \begin{bmatrix} 0 & 1 & 3 & 7 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Gambar 2.1 Contoh matriks eselon baris

(https://www.hudzaifah.net/2020/04/perbedaan-matriks-eselon-dan-matriks.html)

Untuk memecahkan sebuah sistem persamaan linier, kita dapat mengubah matriksnya menjadi sistem persamaan linier baru yang mempunyai himpunan penyelesaian yang sama (tidak mengubah fundamental sistem awal), namun pastinya lebih mudah untuk dipecahkan. Proses pembentukan sistem persamaan linier baru dilakukan dengan menggunakan 3 operasi dasar yang disebut Operasi Baris Elementer (OBE) yaitu:

1. Mempertukarkan dua buah baris persamaan linier ($Li \leftrightarrow Lj$).
2. Mengalikan sebuah baris persamaan linier dengan konstanta/skalar, selama skalar bukan nol ($kLi \rightarrow Li$).
3. Menambahkan kelipatan dari suatu baris persamaan ke baris lain ($kLi + Lj \rightarrow Lj$).

Sebagai contoh, kita akan menyelesaikan SPL di samping dengan metode eliminasi Gauss. Maka, sebagai langkah awal, persamaan-persamaan tersebut akan diubah menjadi matriks yang diperbesar (*augmented*), kemudian diterapkan OBE. Langkah demi langkah kira-kira sebagai berikut:

$$x + 2y + z = 6$$
$$x + 3y + 2z = 9$$
$$2x + y + 2z = 12$$

$$\begin{bmatrix}1 & 2 & 1 & 6\\1 & 3 & 2 & 9\\2 & 1 & 2 & 12\end{bmatrix}\overset{B_2 - B_1}{\sim}\begin{bmatrix}1 & 2 & 1 & 6\\0 & 1 & 1 & 3\\2 & 1 & 2 & 12\end{bmatrix}\overset{B_3 - 2B_1}{\sim}$$

$$\begin{bmatrix}1 & 2 & 1 & 6\\0 & 1 & 1 & 3\\0 & -3 & 0 & 0\end{bmatrix}\overset{B_3 + 3B_2}{\sim}\begin{bmatrix}1 & 2 & 1 & 6\\0 & 1 & 1 & 3\\0 & 0 & 3 & 9\end{bmatrix}\overset{B_3\left(\frac{1}{2}\right)}{\sim}\begin{bmatrix}1 & 2 & 1 & 6\\0 & 1 & 1 & 3\\0 & 0 & 1 & 3\end{bmatrix}$$

Matriks eselon baris yang didapatkan setelah OBE dikembalikan lagi ke dalam bentuk persamaan, kemudian substitusi mundur dilakukan sebagai berikut:

$$\begin{bmatrix}1 & 2 & 1 & 6\\0 & 1 & 1 & 3\\0 & 0 & 1 & 3\end{bmatrix} \Rightarrow \begin{aligned} x + 2y + z &= 6\\ y + z &= 3\\ z &= 3 \end{aligned} \Rightarrow \begin{gathered} z = 3\\ y + 3 = 3,\ y = 0\\ x + 2.0 + 3 = 6,\ x = 3 \end{gathered}$$

2.2 Metode Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan adalah pengembangan dari metode eliminasi Gauss. Metode ini juga digunakan mencari himpunan penyelesaian dari SPL dengan menggunakan OBE, sedemikian hingga matriksnya memiliki bentuk eselon baris tereduksi. Selanjutnya, matriks tersebut diubah kembali ke dalam bentuk sistem persamaan, untuk kemudian dilakukan substitusi balik mulai dari persamaan paling bawah.

Dengan demikian, kita perlu mengenal matriks eselon baris tereduksi beserta perbedaannya dengan matriks eselon baris. Berikut adalah beberapa aturan dari matriks eselon baris tereduksi, yang sebenarnya sama dengan matriks eselon baris, hanya saja ditambahkan aturan 4:

1. Jika baris dalam matriks tidak terdiri seluruhnya dari nol, maka bilangan tak nol pertama dalam baris tersebut adalah 1. Kita menamakan ini sebagai satu utama.
2. Jika terdapat baris yang seluruh entri atau elemennya terdiri dari nol, maka semua baris dengan elemen-elemennya yang berupa nol tersebut dikelompokkan bersama-sama di bawah matriks.
3. Untuk dua baris berurutan yang seluruh elemennya tidak terdiri dari nol, maka satu utama dalam baris yang lebih rendah diletakkan lebih jauh ke kanan dari satu utama dalam baris yang lebih tinggi.
4. Masing-masing kolom yang mengandung satu utama mempunyai elemen nol pada tempat lain di kolom yang sama (semua angka di atas dan bawah satu utama adalah 0).

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

Gambar 2.2 Contoh matriks eselon baris tereduksi
(https://jagostat.com/aljabar-linear/bentuk-eselon-baris-dan-eselon-baris-tereduksi)

2.3 Determinan Matriks

Setiap matriks bujur sangkar (persegi; ukuran baris dan kolomnya sama) selalu dihubungkan dengan suatu skalar yang disebut determinan dari matriks tersebut. Determinan dari matriks A ditulis det (A) atau $|A|$.

Untuk mencari determinan matriks 2×2 dan 3×3, dapat digunakan metode Sarrus, yaitu dengan menggambar anak panah berarah sebagai berikut:

$$A_1 = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \Rightarrow \begin{bmatrix} a & b \\ c & d \end{bmatrix} \Rightarrow \det(A_1) = ad - bc$$

$$A_2 = \begin{bmatrix} a & b & c \\ d & e & e \\ f & g & h \end{bmatrix} \Rightarrow \left[\begin{matrix} a & b & c \\ d & e & f \\ g & h & i \end{matrix}\right] \begin{matrix} a & b \\ d & e \\ g & h \end{matrix}$$

$$\det(A_2) = aei + bfg + cdh - ceg - afh - bdi$$

2.3.1 Metode Reduksi Baris

Metode ini sangat membantu untuk menghitung determinan matriks yang ukurannya yang cukup besar. Berikut beberapa teorema dan aturannya:

1). Misal diberikan suatu matriks bujur sangkar A. Jika A memiliki satu atau lebih baris (atau kolom) yang semua elemennya nol, maka det (A) = 0. Jika mentranspos matriks A maka det (A) = det (A^T).

2). Jika A adalah matriks segitiga atas, segitiga bawah, atau matriks diagonal dengan ukuran n×n, maka det (A) adalah hasil kali elemen-elemen pada diagonal utamanya.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 4 & 0 & 0 \\ 5 & -2 & 3 & 0 \\ 3 & 6 & 7 & 3 \end{bmatrix}$$

Sebagai contoh, diberikan matriks A seperti di samping. A adalah matriks segitiga bawah (karena elemen di atas diagonal utama semuanya bernilai 0), sehingga diperoleh det (A) = 1×4×3×3 = 36.

Perhatikan bahwa pada matriks awal yang diberikan, bisa diterapkan OBE sampai membentuk matriks segitiga atau matriks diagonal (matriks yang elemen di atas dan bawah diagonal utama semuanya bernilai nol), dan determinannya dengan mudah ditentukan. Namun, jangan lupa syarat berikut:

1). Misal diberikan suatu matriks A berukuran $n \times n$. Matriks B adalah hasil OBE dari matriks A. Jika matriks B diperoleh dengan mengalikan salah satu baris atau kolom pada matriks A dengan suatu skalar k, maka det (B) = k det (A).

2). Jika matriks B diperoleh dengan menukarkan dua baris atau dua kolom pada matriks A, maka det (B) = –det (A).

3). Jika matriks B diperoleh dengan menambahkan atau mengurangkan kelipatan suatu baris (kolom) pada matriks A ke baris (kolom) yang lain, maka det (B) = det (A).

Ketiga aturan OBE untuk determinan di atas dapat dirangkum menjadi sebuah rumus:

$$|A| = \frac{-1^p}{k_1 \times k_2 \times k_3 \times \ldots \times k_n} \times |B|$$

dengan p adalah jumlah total pertukaran baris, dan k_x adalah seluruh konstanta yang dikalikan pada suatu baris di matriks (jika ada).

2.3.2 Metode Ekspansi Kofaktor

Untuk menghitung determinan matriks dengan metode ini, pertama-tama kita perlu mengetahui definisi dari minor entri dan kofaktor.

Jika $A = (a_{ij})$ adalah suatu matriks bujur sangkar, maka minor dari elemen a_{ij} dinyatakan dengan M_{ij}, adalah determinan dari submatriks A yang dibentuk dengan menghilangkan baris ke-i dan kolom ke-j. Sedangkan kofaktor dari elemen a_{ij} dinyatakan dengan $C_{ij} = (-1)^{i+j} \times M_{ij}$. Contohnya sebagai berikut:

$$A = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \qquad M_{11} = \begin{vmatrix} \cancel{1} & \cancel{2} & \cancel{3} \\ \cancel{4} & 5 & 6 \\ \cancel{7} & 8 & 9 \end{vmatrix} = \begin{vmatrix} 5 & 6 \\ 7 & 8 \end{vmatrix} = -3$$

$$C_{11} = (-1)^{1+1} M_{11} = -3 \qquad M_{12} = \begin{vmatrix} \cancel{1} & \cancel{2} & \cancel{3} \\ 4 & \cancel{5} & 6 \\ 7 & \cancel{8} & 9 \end{vmatrix} = \begin{vmatrix} 4 & 6 \\ 7 & 9 \end{vmatrix} = -6$$

$$C_{12} = (-1)^{1+2} M_{11} = -M_{11} = 6$$

Jika A adalah matriks bujursangkar $n \times n$, maka determinan matriks A dapat dicari dengan mengalikan elemen-elemen pada sembarang baris (atau kolom) dengan kofaktornya masing-masing, kemudian menjumlahkannya. Contohnya sebagai berikut:

$$A = \begin{bmatrix} 2 & 3 & 2 \\ 4 & 1 & 1 \\ 0 & 3 & 2 \end{bmatrix} \qquad \begin{aligned} Det\ (A) &= a_{11}C_{11} + a_{12}C_{12} + a_{13}C_{13} \\ &= 2(1)(-1) + 3(-1)(8) + 2(1)(12) \\ &= -2 - 24 + 24 \ = -2 \end{aligned}$$

2.4 Matriks Balikan

Matriks balikan adalah kebalikan (inversi / *invers*) dari sebuah matriks yang berukuran $n \times n$. Apabila sebuah matriks dikalikan dengan inversinya, akan menjadi matriks identitas. *Invers* matriks dilambangkan dengan A^{-1}. Suatu matriks dikatakan memiliki balikan jika determinan dari matriks tersebut tidak sama dengan nol.

2.4.1 Metode Reduksi Baris

Untuk matriks bujur sangkar secara umum, balikan dari matriks tersebut dapat dicari dengan menggunakan Operasi Baris Elementer (OBE). Misalkan A adalah matriks bujur sangkar, pada matriks tersebut dilakukan serangkaian OBE sedemikian sehingga menjadi matriks identitas I. Selanjutnya, pada matriks I juga dilakukan serangkaian

OBE yang sama percis dengan yang dilakukan pada matriks *A* dari awal sampai akhir. Secara otomatis akan diperoleh matriks A^{-1}.

[*A* | *I*] diterapkan OBE pada *A* sekaligus *I* sampai akhirnya menjadi $[\ I \mid A^{-1}\ ]$

2.4.2 Metode Matriks Adjoin

Balikan suatu matriks bujur sangkar juga dapat dicari dengan menggunakan adjoin matriks tersebut. Untuk itu, pertama-tama kita harus memahami definisi dari matriks kofaktor dan matriks adjoin.

Jika *A* adalah matriks bujur sangkar $n \times n$ dan C_{ij} adalah kofaktor dari a_{ij}, maka matriks kofaktor dari *A* adalah matriks dalam bentuk:

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

Transpos (*transpose*) dari matriks kofaktor dari *A* tersebut dinamakan matriks adjoin A, atau biasa dilambangkan dengan adj (*A*).

Selanjutnya, bisa ditentukan balikan dari matriks *A* dengan rumus:

$$A^{-1} = \frac{1}{|A|} \times adj(A)$$

2.5 Kaidah Cramer

Aturan Cramer (*Cramer's Rule*) merupakan formula yang dipakai untuk menyelesaikan sistem persamaan linier dengan menggunakan determinan dari matriks yang terbentuk dari koefisien dan konstanta masing-masing persamaan di sistem tersebut. Berdasarkan catatan sejarah, Gabriel Cramer (1704 – 1752), matematikawan dari Swiss, merupakan orang pertama yang menerbitkan aturan ini pada tahun 1750. Contoh menggunakan kaidah Cramer:

$$\begin{cases} a_{11}x + a_{12}y + a_{13}z & = c_1 \\ a_{21}x + a_{22}y + a_{23}z & = c_2 \\ a_{31}x + a_{32}y + a_{33}z & = c_3 \end{cases}$$

Jika diberikan SPL seperti di atas, kemudian diubah ke dalam bentuk matriks *A*, solusinya dapat ditemukan: $x = \frac{|A_1|}{|A|}$, $y = \frac{|A_2|}{|A|}$, $z = \frac{|A_3|}{|A|}$, dengan A_i adalah matriks *A* yang kolom ke-*i*-nya digantikan dengan masing-masing *c* pada kolom solusi.

$$A_1 = \begin{bmatrix} c_1 & a_{12} & a_{13} \\ c_2 & a_{22} & a_{23} \\ c_3 & a_{32} & a_{33} \end{bmatrix} \quad A_2 = \begin{bmatrix} a_{11} & c_1 & a_{13} \\ a_{21} & c_2 & a_{23} \\ a_{31} & c_3 & a_{33} \end{bmatrix} \quad A_3 = \begin{bmatrix} a_{11} & a_{12} & c_1 \\ a_{21} & a_{22} & c_2 \\ a_{31} & a_{32} & c_3 \end{bmatrix}$$

2.6 Interpolasi Polinom

Interpolasi adalah perkiraan suatu nilai tengah dari suatu set nilai yang diketahui. Interpolasi polinomial merupakan teknik interpolasi dengan mengasumsikan pola data yang dimiliki mengikuti pola polinomial, baik berderajat satu (linier) maupun berderajat tinggi. Interpolasi dengan metode ini dilakukan dengan terlebih dahulu membentuk persamaan polinomial dari beberapa titik yang merupakan nilai fungsi pada *range* yang diinginkan. Persamaan yang terbentuk selanjutnya digunakan untuk mendapatkan nilai fungsi dari suatu titik sembarang di tengah *range*.

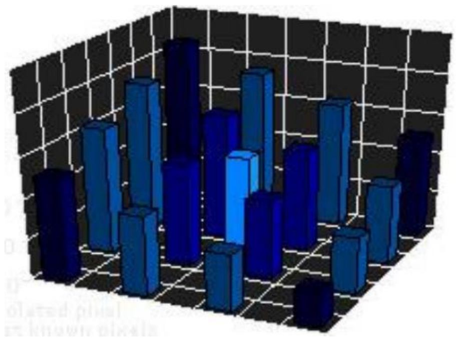
Persamaan polinomial biasanya disajikan dalam bentuk:

$$a_0 + a_1x_0 + a_2x_0^2 + \ldots + a_n x_0^n = y_0$$
$$a_0 + a_1x_1 + a_2x_1^2 + \ldots + a_n x_1^n = y_1$$
$$\ldots \ldots$$
$$a_0 + a_1x_n + a_2x_n^2 + \ldots + a_n x_n^n = y_n$$

di mana a merupakan koefisien polinomial, x merupakan variabel, dan n merupakan indeks serta pangkat polinomial. Untuk melakukan evaluasi polinomial, nilai x dan y yang bersangkutan akan disubstitusikan pada masing-masing elemen persamaan polinomial. Maka terbentuklah sebuah Sistem Persamaan Linier, dan a_0, a_1, a_2, … a_n harus ditentukan.

2.7 Interpolasi Bicubic

Interpolasi bicubic dianggap sebagai salah satu interpolasi yang cukup baik dalam interpolasi piksel. Interpolasi bicubic mengambil nilai piksel dari 4×4 nilai piksel di sekitarnya, atau dengan kata lain 16 piksel yang terdekat. Interpolasi ini dapat memberikan hasil penskalaan yang lebih baik karena mengambil sampel piksel yang lebih banyak dibandingkan metode interpolasi lain.



Gambar 2.3 Penentuan nilai piksel dengan algoritma interpolasi bicubic

Semakin banyak sampel piksel yang diambil, akan dihasilkan kualitas citra yang lebih baik. Oleh karena itu, interpolasi ini banyak digunakan sebagai standar program *image editing* seperti Photoshop, Corel, dan Photopaint. Rumus untuk melakukan interpolasi bicubic.

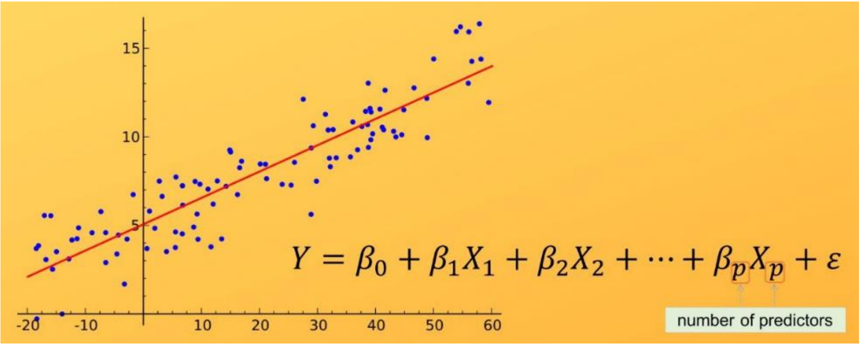
$$p(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j.$$

2.8 Regresi Linier Berganda

Regresi adalah salah satu cara memprediksi nilai selain menggunakan interpolasi. Analisis regresi yang satu ini melibatkan lebih dari satu variabel bebas atau prediktor, berbeda dengan analisis regresi sederhana hanya melibatkan satu variabel dependen dan satu variabel independen. Analisis regresi linier berganda adalah hubungan secara linier antara dua atau lebih variabel independen (x_1, x_2, … x_n) dengan variabel dependen (y). Maka, data yang digunakan biasanya berskala interval atau rasio. Persamaan regresi linear berganda adalah sebagai berikut:

$$y' = a + b_1x_1 + b_2x_2 + \cdots + b_nx_n$$

dengan y' adalah variabel dependen (nilai yang diprediksi), x_1 dan x_2 adalah variabel independen, a adalah konstanta (nilai y' apabila seluruh x bernilai 0), dan b adalah koefisien regresi (nilai peningkatan atau penurunan).



Gambar 2.4 Contoh grafik regresi linier berganda

(https://medium.com/analytics-vidhya/multiple-linear-regression-from-scratch-using-only-numpy-98fc010a1926)

BAB 3
IMPLEMENTASI OPERASI DALAM BAHASA JAVA

3.1 Proyek Java

Program yang kami buat dalam bentuk *Java project* dengan nama Void_Algeo_Matrix yang berisikan *setting* VSCode, bin, FileKeluaran, lib, src, dan test. *Setting* VSCode sebagai sebuah pengaturan dalam *Java project* yang kami buat. Folder bin berisi *file* (xxx.class) pada program Java. Folder FileKeluaran berfungsi untuk menampung *file-file* yang akan disimpan ketika pengguna memilih untuk menyimpan *file* saat program dijalankan.

Folder src berisi program utama yang kami buat. Folder tersebut berisi satu buah *package*, yaitu VoidMatrix. Di dalam *package* itu ada *file-file* program Java yang kami buat, terdiri atas Main.java sebagai kelas utama untuk menjalankan keseluruhan program, serta kelas-kelas lain yaitu BicubicInterpolation.java, Cramer.java, DisplayPengguna.java, EliminasiGauss.java, GaussJordan.java, InversOBE.java, Matrix.java, MultipleLinearRegression.java, dan PolynomialInterpolation.java.

3.2 Penjelasan Main.java

Program Main adalah sebuah program utama yang akan menjalankan keseluruhan program. Di dalam Main menggunakan

3.3 Menu Sistem Persamaan Linier

Pada saat pengguna memilih menu ini, akan ditampilkan beberapa submenu. Setelah memilih submenu, pengguna akan ditanya mengenai matriks yang ingin dioperasikan, apakah ingin dimasukkan dengan mengetik secara langsung atau mengambil dari sebuah fail. Jika pengguna ingin membaca dari sebuah fail, maka pengguna akan diminta untuk menuliskan alamat fail yang diinginkan.

3.4.1 Metode Eliminasi Gauss

Jika pengguna memilih metode ini, program akan membuat sebuah objek untuk kelas Matrix dengan beberapa parameter *boolean*. Pada kelas Matrix sendiri, terdapat sebuah konstruktor yang akan membuat sebuah matriks dari masukan pengguna, baik dari *keyboard* maupun fail.

Setelah itu, program membuat objek eliminasi Gauss, kemudian memanggil prosedur GaussElimination. Pertama-tama, fungsi tersebut akan memisahkan matriks *augmented* ($Ax = B$) pilihan pengguna menjadi matriks A dan matriks B. Kedua matriks ini digunakan untuk menentukan apakah SPL pengguna memiliki solusi unik, banyak, atau bahkan tidak punya solusi. Pemeriksaannya pun mengundang fungsi lain, seperti det (determinan), NotUniqueSolution, NotUniqueSolutionRow, serta baris_nol_semua.

Setelah itu, kolom mulai dicacah satu per satu untuk melakukan operasi terhadap matriks *augmented*. Dimulai dari pertukaran baris berdasarkan nilai maksimum per kolom, dilanjutkan dengan Operasi Baris Elementer (OBE) per baris.

Langkah selanjutnya, program membuat *leading one* pada masing-masing baris dengan membagi setiap elemen pada diagonal utama dengan dirinya sendiri. Pada akhirnya, solusi akan ditampilkan berdasarkan evaluasi yang sudah dilakukan di awal. Apabila solusi banyak, akan ditampilkan dalam bentuk persamaan parametrik sederhana (menggunakan prosedur SubstitusiParametrik). Jika tidak punya solusi, maka ditampilkan pesan khusus. Jika solusinya unik, digunakan prosedur SubstitusiMundur untuk mendapatkan *array* hasil.

3.4.2 Metode Eliminasi Gauss-Jordan

Metode ini sangat mirip dengan metode eliminasi Gauss. Hanya saja, setelah program membuat *leading one* untuk setiap baris, ditambahkan satu operasi pencacahan lagi bagi setiap baris untuk membuat semua angka di kanan dan kiri serta atas dan bawah *leading one* menjadi 0. Solusi ditampilkan dengan cara sama.

3.4.3 Metode Matriks Balikan

Pada submenu ini program akan membuat objek untuk kelas Matrix. Selanjutnya program akan memanggil fungsi inversSPL dari kelas Matrix untuk mendapatkan hasil dari sitem persamaan linear.

3.4.4 Kaidah Cramer

Jika pengguna memilih metode ini, setelah objek untuk kelas Matrix dibuat sama seperti pada metode sebelumnya, program akan memisahkan matriks *augmented* pilihan pengguna ($Ax = B$) menjadi matriks A dan B. Setelah itu, matriks salinan A dibuat secara berulang. Pada setiap pengulangan, sebuah kolom (tertentu, terurut dari kiri ke kanan) pada matriks salinan diganti dengan matriks B, sehingga dengan memanggil fungsi det (determinan), *array* hasil bisa langsung diisi satu per satu dengan pembagian antara determinan salinan A dengan determinan A.

3.4 Menu Determinan

Sama seperti sebelumnya, saat pengguna memilih menu ini, akan ditampilkan dua buah submenu, serta pertanyaan pada pengguna mengenai asal matriks yang ingin dioperasikan.

3.4.1 Metode Reduksi Baris

3.4.2 Metode Ekspansi Kofaktor

3.5 Menu Matriks Balikan

Sama seperti sebelumnya, saat pengguna memilih menu ini, akan ditampilkan dua buah submenu, serta pertanyaan pada pengguna mengenai asal matriks yang ingin dioperasikan.

3.5.1 Metode Reduksi Baris

Jika pengguna memilih metode ini, setelah objek untuk kelas Matrix dibuat, program akan membuat sebuah matriks tambahan terpisah, yaitu matriks identitas berukuran $n \times n$, yaitu ukuran yang sama dengan matriks milik pengguna. Setelah itu, diterapkan operasi yang sama seperti pada prosedur GaussJordanElimination, namun matriks identitas yang telah dibuat ikut dioperasikan dengan cara yang sama percis baris per baris. Matriks identitas yang telah selesai dioperasikan itulah yang ditampilkan sebagai hasil balikan.

3.6 Menu Interpolasi Polinom

Program menerima masukkan matriks, kemudian matriks tersebut diproses untuk mendapatkan interpolasi polinom.

3.7 Menu Interpolasi Bicubic

Pada menu ini program akan membuat objek untuk kelas BicubicInterpolation. Pada kelas tersebut terdapat beberapa deklarasi untuk matriks yang akan digunakan yaitu matriks berukuran 16x16 dengan isi yang telah ditentukan pada spesifikasi tugas. Program akan menginisiasi nilai matriks dengan iterasi. Selanjutnya program akan meminta untuk memilih masukan yang berasal dari keyboard dan file. Jika dengan keyboard meminta 16 titik masukan pada matriks 4x4 yaitu nilai f(x,y). Selanjutnya program akan meminta masukan untuk

3.8 Menu Regresi Linier berganda

Program menerima masukan banyaknya peubah dan banyak sampel. Banyak peubah dan sampel ini menjadi jumlah baris dan kolom matriks res. Kemudian program meminta memasukkan nilai sampel. Nilai sampel ini kemudian dimasukkan ke matriks toEst yang memiliki panjang peubah + 1. Kemudian matriks res dikalkulasi sesuai rumus regresi linear berganda menggunakan salah satu fungsi SPL, dalam program kami menggunakan fungsi cramer. Setelah didapatkan koefisien masing-masing variabel, matriks toEst dimasukkan ke dalam persamaan untuk mendapatkan nilai taksiran.

BAB 4
EKSPERIMEN DAN STUDI KASUS

3.9 Menemukan Solusi Sistem Persamaan Linier

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

```
Matriks tidak memilki balikan


a0: -11.00
a1: -66.00
a2: 26.00
a3: -3.00
Apakah ingin disimpan (y/n) : n

File tidak disimpan
```

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

```
Matriks tidak memilki balikan
```

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

```
Matriks tidak memilki balikan
```

d)

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

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

n = 6

```
1.00 0.50 0.33 0.25 0.20 0.17 1.00
0.00 1.00 1.07 1.00 0.91 0.83 -4.00
0.00 0.00 1.00 1.67 2.06 2.27 12.00
0.00 0.00 0.00 1.00 2.07 2.95 -109.05
-0.00 -0.00 -0.00 -0.00 1.00 2.46 692.85
-0.00 -0.00 -0.00 -0.00 -0.00 1.00 -1200.39

x1 = 31.658301
x2 = -462.958969
x3 = 2101.724643
x4 = -4111.774565
x5 = 3641.329725
x6 = -1200.389964
```

n = 10

```
Solusi Sistem Persamaan Linier Anda:

1.00 0.50 0.33 0.25 0.20 0.17 0.14 0.13 0.11 0.10 1.00
0.00 1.00 1.07 1.00 0.91 0.83 0.76 0.70 0.65 0.60 -4.00
0.00 0.00 1.00 1.70 2.14 2.40 2.55 2.63 2.65 2.65 10.71
0.00 0.00 0.00 1.00 2.10 3.04 3.76 4.30 4.69 4.97 -95.11
0.00 0.00 0.00 0.00 1.00 2.43 3.92 5.53 7.43 8.86 254.43
-0.00 -0.00 -0.00 0.00 -0.00 1.00 2.24 2.97 3.10 3.69 175.97
-0.00 -0.00 -0.00 0.00 -0.00 -0.00 1.00 1.84 1.63 1.44 277.17
-0.00 -0.00 -0.00 -0.00 -0.00 -0.00 -0.00 1.00 0.91 0.12 686.35
-0.00 -0.00 -0.00 -0.00 -0.00 -0.00 -0.00 -0.00 1.00 1.06 -206.03
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 -1087.47

x1 = 35.737666
x2 = -526.570885
x3 = 2069.789456
x4 = -2617.704547
x5 = 287.080708
x6 = 549.509169
x7 = 369.863408
x8 = -45.441677
x9 = 950.743993
x10 = -1087.474827
```

4.2 Menemukan Solusi SPL Berbentuk Matriks *Augmented*

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

```
1.00 0.00 0.00 -1.00 -1.00
0.00 1.00 -2.00 0.00 0.00
0.00 0.00 0.00 0.00 0.00
0.00 0.00 0.00 0.00 0.00

Solusi Sistem Persamaan Linier Anda ada banyak, namun harus memenuhi:
x4 = 0
x1 = 0
-2.000000 x3 + 1.000000 x2 = 0.000000

ATAU

x3 = 0
x2 = 0
-1.000000 x4 + 1.000000 x1 = -1.000000
```

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

```
Matriks tidak memilki balikan
```

4.3 Menemukan Solusi SPL Berbentuk Persamaan

a) $$\begin{aligned} 8x_1 + x_2 + 3x_3 + 2x_4 &= 0 \\ 2x_1 + 9x_2 - x_3 - 2x_4 &= 1 \\ x_1 + 3x_2 + 2x_3 - x_4 &= 2 \\ x_1 + 6x_3 + 4x_4 &= 3 \end{aligned}$$

```
1.00 0.13 0.38 0.25 0.00
0.00 1.00 -0.20 -0.29 0.11
0.00 0.00 1.00 0.66 0.54
-0.00 -0.00 -0.00 1.00 -0.26

x1 = -0.224324
x2 = 0.182432
x3 = 0.709459
x4 = -0.258108
```

b)

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

```
Matriks tidak memilki balikan
```

4.4 Studi Kasus Interpolasi

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) = ?

```
x = 0.2, y = 0.13
```

```
x = 0.55, y = 2.1375716210670817
```

```
x = 0.85, y = -66.26963932037575
```

```
x = 1.28, y = -3485.1449018889107
```

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

tanggal(desimal) = bulan + (tanggal / jumlah hari pada bulan tersebut)

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

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:

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

```
x = 7.516, y = 53.53776015394944
```

```
x = 8.323, y = 36.29469165556574
```

```
x = 9.16, y = -608.620770120513
```

4.5 Studi Kasus Interpolasi Bicubic

Diberikan matriks input:

$$\begin{bmatrix} 153 & 59 & 210 & 96 \\ 125 & 161 & 72 & 81 \\ 98 & 101 & 42 & 12 \\ 21 & 51 & 0 & 16 \end{bmatrix}$$

Tentukan nilai:

f(0,0) = ?

f(0.5, 0.5) = ?

f(0.25, 0.75) = ?

f(0.1, 0.9) = ?

```
f(0.00,0.00) : 161.000
```

```
f(0.50,0.50) : 97.727
```

```
f(0.25,0.75) : 105.515
```

```
f(1.00,1.00) : 42.002
```

4.6 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: Y = -3.507778141591853 + -0.002624990744829821 X1 + 7.989410387216545E-4 X2
+ 0.15415503024219504 X3
Taksiran f(x): 0.938434
```

BAB 5
PENUTUP

5.1 Kesimpulan

Menurut KBBI, matriks adalah susunan unsur matematis yang berbentuk segi empat, ditulis di antara kurung, terdiri atas kolom dan baris. Salah satu kegunaan matriks adalah untuk mengekspresikan koefisien-koefisien serta konstanta suatu Sistem Persamaan Linier (SPL) $Ax = B$ dalam bentuk *augmented*. SPL dalam bentuk matriks tesebut kemudian dapat dimodifikasi dengan menggunakan Operasi Baris Elementer (OBE), yang hasilnya bisa dipakai untuk menemukan solusi SPL, menghitung determinan matriks A, maupun mencari balikan dari matriks A, dan lain sebagainya.

Berdasarkan percobaan kami, operasi-operasi pada matriks dapat dibuat implementasinya dalam bahasa pemrograman Java. Lebih dari itu, fungsi dan prosedur yang telah dibuat bisa juga digunakan untuk menyelesaikan persoalan sehari-hari, seperti interpolasi polinom, interpolasi bicubic, serta regresi linier berganda, terbukti dengan lancarnya seluruh *test case* yang dicantumkan pada **BAB 4**.

5.2 Saran

1. Dalam mengerjakan pemrograman, sebaiknya pemrogram memiliki catatan kecil tentang fungsi, prosedur, dan juga *class* yang telah dibuat. Hal ini bertujuan untuk mempermudah kerja sama tim serta diri sendiri apabila ingin memodifikasi program.
2. Membaca spesifikasi dengan teliti sebelum melakukan pemrograman sebaiknya dilakukan bersama-sama dengan anggota kelompok yang lain. Dengan demikian, setiap orang bisa saling mengingatkan apabila ada spesifikasi yang belum terpenuhi.
3. Jika dalam pengerjaan pemrograman seseorang tiba-tiba merasa penat, ada baiknya pekerjaan dihentikan dulu sementara, dan anggota kelompok yang lain membantunya untuk menyegarkan pikiran. Setelah semuanya kembali segar, program yang dibuat biasanya menjadi lebih bagus dan cepat selesai.

5.3 Refleksi

Tugas besar ini memberikan banyak sekali pengalaman dan kebersamaan bagi kelompok kami. Selain menambah teman baru, kami juga menambah ilmu baru tentang pemrograman berbagai operasi matriks dalam bahasa Java. Di sini pula kami belajar, bahwa pengaturan waktu dan koordinasi yang baik sangat diperlukan dalam membentuk sebuah tim yang kokoh. Apabila satu orang saja tidak menjalankan tugasnya dengan benar, maka keseluruhan program bisa menjadi salah.

Dengan keterbatasan pengetahuan dan keahlian, kami pun jadi harus berusaha sangat keras mengeksplor lebih dan lebih dalam lagi mengenai tugas ini, baik dari internet, teman, juga saudara. Akhirnya, usaha kami tidak sia-sia. Tugas besar pertama kami di jurusan Teknik Informatika terselesaikan dengan baik dan tepat waktu.

Lampiran

https://github.com/NerbFox/Algeo01-21043.git

DAFTAR REFERENSI

1. Long, Tju Ji. Tanpa tanggal. *Matriks Eselon Baris dan Eselon Baris Tereduksi*. (Diakses tanggal 2 Oktober 2022 dari https://jagostat.com/aljabar-linear/bentuk-eselon-baris-dan-eselon-baris-tereduksi)
2. Tim Profematika. 9 Maret 2019. *Pengenalan Operasi Baris Elementer*. (Diakses tanggal 2 Oktober 2022 dari https://www.profematika.com/pengenalan-operasi-baris-elementer/)
3. Syarifuddin dkk. 2016. *Aljabar Linear*. Mataram: LPP Mandala.
4. Adha, Sophia Maulidatul. 6 September 2022. *Matriks – Pengertian, Operasi, Determinan, Invers, dan Contoh Soal*. (Diakses tanggal 3 Oktober 2022 dari https://akupintar.id/info-pintar/-/blogs/matriks-pengertian-operasi-determinan-invers-dan-contoh-soal)
5. Sukardi. 11 Mei 2022. *Materi, Soal, dan Pembahasan – Aturan Cramer*. (Diakses tanggal 3 Oktober 2022 dari https://mathcyber1997.com/materi-soal-dan-pembahasan-aturan-cramer/)
6. Rosidi, Mohammad. 2019. *Metode Numerik Menggunakan R Untuk Teknik Lingkungan*. (Diakses tanggal 3 Oktober 2022 dari https://bookdown.org/moh_rosidi2610/Metode_Numerik/)
7. Ivando, Albert, dkk. Tanpa tahun. *Implementasi Metode Interpolasi Bicubic Modifikasi untuk Perbaikan Citra Hasil Penskalaan*. STMIK MDP Palembang. (Diakses tanggal 3 Oktober 2022 dari https://adoc.pub/implementasi-metode-interpolasi-bicubic-modifikasi-untuk-per.html)
8. Wardana, Raditya. 23 November 2019. *Yuk, Cari Tahu Penjelasan Regresi Linear Berganda di Sini!*. (Diakses pada tanggal 3 Oktober 2022 dari https://lifepal.co.id/media/regresi-linear-berganda/)