No. Selasa
Date 13/2. 2024

Latihan 1 Metode Numerik

Nama : Antonius Aditya Rizky Wijaya
NIM : G5402221003

2.32.) Given a nonsingular system of linear equation $Ax=b$, what effect on the solution vector x results from each of the following actions?

a. Permuting the rows of $[A \;\; b]$

Menukar baris dari $[A\,b]$ akan menghasilkan vektor solusi yang berbeda karena urutan persamaan dalam sistem juga berubah. Namun, jika hanya baris yang ditukar dan tidak ada perubahan elemen-elemen didalamnya, maka vektor solusi akan tetap konsisten.

b Permuting the columns of A

Menukar kolom-kolom dari matriks A akan mengubah koefisien dari setiap variabel dalam sistem. Akibatnya vektor solusi juga akan berubah.

c. Multiplying both sides of the equation from the left by a nonsingular matrix M

Mengalikan kedua sisi dengan matriks nonsingular M akan menghasilkan solusi yang sama, misalkan $Ax=b$, kemudian dikalikan dengan matriks nonsingular M dari kiri, maka $(MA)x=(Mb)$, solusinya akan tetap x karena $(MA)x = M(Ax) = Mb$

2.33) Suppose that both sides of a system of linear equations $Ax=b$ are multiplied by a nonzero scalar α.

a) Does this change the true solution x?

Jika kedua sisi dari sistem persamaan linear $Ax=b$ dikalikan dengan sebuah skalar tak nol, maka solusi sebenarnya tidak akan berubah. Ini karena setiap solusi akan memenuhi $Ax=b$ juga memenuhi $\alpha Ax = \alpha b$.

b. Does this change the residual vector $r = b - Ax$ for a given x?

Hal ini tidak akan mengubah vektor residual $r = b - Ax$.

$\alpha r = \alpha(b - Ax) = \alpha b - \alpha Ax = \alpha(b - Ax) = \alpha r$

c. What conclusion can be drawn about assessing the quality of a computed solution?

Dapat disimpulkan bahwa mengalikan kedua sisi persamaan dengan skalar α tak nol, tidak mempengaruhi solusi dari hasil sebenarnya maupun vektor residual.

2.34) Suppose that both sides of a system of linear equations $Ax = b$ are premultiplied by a nonsingular diagonal matrix

(a) Does this change the true solution x?

Tidak, mengalikan kedua sisi SPL $Ax = b$ dengan sebuah matriks diagonal nonsingular tidak akan mengubah solusi x. Ini karena perkalian dengan matriks diagonal hanya mengubah skala setiap persamaan dalam sistem tanpa mempengaruhi hubungan relatif antara variabel-variabel.

(b) Can this affect the conditioning of the system?

Tidak, hal ini tidak mempengaruhi kondisi sistem. Kondisi sistem tergantung pada matriks koefisien A dan tidak berubah oleh perkalian dengan matriks diagonal nonsingular.

(c) Can this affect the choice of pivots in Gaussian elimination?

Tidak, ini tidak akan mempengaruhi pilihan pivot dalam eliminasi Gaussian. Pivot dipilih berdasarkan nilai absolut dari elemen-elemen di dalam matriks koefisien A, dan mengalikan kedua sisi dengan matriks diagonal hanya mengubah skala persamaan tetapi tidak mengubah nilai-nilai ini. Sehingga, pilihan pivot dalam eliminasi Gaussian tetap sama.

2.35) Specify an elementary elimination matrix that zeros the last two components of the vector. $\begin{bmatrix} 3 \\ 2 \\ -1 \\ 4 \end{bmatrix}$

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 4 \end{array}\right] \underset{\sim}{E_{43}(4)} \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 4 & 0 & 0 \end{array}\right] \underset{\sim}{E_{32}(1/2)} \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 1/2 & 1 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \end{array}\right]$$

∴ Sehingga untuk menghasilkan vektor dengan dua komponen terakhir nol dari vektor $\begin{bmatrix} 3 \\ 2 \\ -1 \\ 4 \end{bmatrix}$ adalah mengalikannya dengan matriks $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1/2 & 1 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}$

didapat $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1/2 & 1 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix} \times \begin{bmatrix} 3 \\ 2 \\ -1 \\ 4 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 0 \\ 0 \end{bmatrix}$

2.38) (a) What is the difference between partial pivoting and complete pivoting in Gaussian elimination?

* Dalam partial pivoting, setiap langkah dari eliminasi Gaussian melibatkan pertukaran baris sehingga elemen dengan nilai absolut terbesar berada diposisi pivot. Namun, hanya kolom yang dipertimbangkan saat pemilihan pivot, tidak memperhatikan elemen-elemen diluar kolom.
* Dalam complete pivoting, tidak hanya pertukaran baris yang dilakukan, tetapi juga pertukaran kolom. Ini berarti, selama eliminasi elemen dengan nilai absolut terbesar diantara semua sisa elemen yang belum diolah di dalam submatriks menjadi pivot.

(b) State one advantage of each type of pivoting relative to the order.

* Keuntungan partial pivoting yaitu lebih efisien secara komputasi karena hanya memerlukan pertukaran baris. Biasanya partial pivoting cukup untuk memperbaiki kegagalan eliminasi Gaussian yang disebabkan oleh elemen dengan nilai absolut yang sangat kecil di bawah pivot.
* Keuntungan complete pivoting yaitu lebih stabil numeriknya karena mempertimbangkan seluruh sisa matriks di dalam submatriks saat memilih pivot. Selain itu, dapat mengurangi kesalahan pembulatan yang disebabkan oleh nilai kecil diluar kolom.

2.39) Consider the following matrix A, whose LU factorization we wish to compute using Gaussian elimination:

$$A = \begin{bmatrix} 4 & -8 & 1 \\ 6 & 5 & 7 \\ 0 & -10 & -3 \end{bmatrix}$$

What will the initial pivot element be if

(a) No pivoting is used?

Elemen pivot awal adalah 4.

(b) Partial pivoting is used?

Kita mencari elemen dengan nilai absolut terbesar di kolom pertama. Elemen tersebut adalah 6 dibaris kedua. Kita tukar baris pertama dengan baris kedua sebelum memulai eliminasi. Jadi elemen pivot awal adalah 6.

(c) Complete pivot used?

Kita mencari elemen dengan nilai absolut terbesar di submatriks yang belum diproses. Elemen tersebut adalah 10 dibaris ketiga. Kita tukar baris petama dengan baris ketiga dan kolom kedua dengan kolom pertama. Jadi elemen pivot awal = 10.

No.

Date

2.40) Give two reasons why pivoting is essential for a numerically stable implementation of Gaussian elimination!

1.) Menghindari pembagian dengan nol :

Dalam eliminasi Gaussian, pembagian dilakukan dengan elemen pivot. Jika elemen pivot sangat kecil atau bahkan nol, ini dapat menyebabkan pembagian oleh nol atau pembagian yang sangat dekat dengan nol. Pivoting memungkinkan kita untuk memilih pivot yang memiliki nilai absolut terbesar antara elemen-elemen yang tersedia, sehingga mengurangi risiko pembagian oleh nol atau pembagian yang sangat kecil.

2.) Meningkatkan stabilitas numerik :

Dalam beberapa kasus, matriks koefisien dapat menjadi buruk terkondisi, yang artinya bahwa perubahan kecil pada input dapat menyebabkan perubahan besar pada output. Pivoting membantu mengurangi kemungkinan terjadi situasi ini. Dengan menggunakan pivot yang sesuai, kita dapat mengurangi penyebaran kesalahan akibat kesalahan pembulatan atau kecilnya elemen matriks.

2.42) (a) What is the inverse of the following matrix?

$$A|I = \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & m_1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & m_2 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right] \begin{array}{c} E_{32}(-m_1) \\ E_{42}(-m_2) \end{array} \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -m_1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -m_2 & 0 & 1 \end{array}\right] = I|A^{-1}$$

$$\therefore \text{jadi } A^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -m_1 & 1 & 0 \\ 0 & -m_2 & 0 & 1 \end{bmatrix}$$

(b) How might such a matrix arise in computational practice?

Matrik seperti ini mungkin muncul dalam praktik komputasi, terutama dalam konteks metode eliminasi Gaussian untuk menyelesaikan persamaan linear. Misal jika kita ingin menggunakan metode eliminasi gaussian untuk memecahkan persamaan linear dengan matrik koefisien :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & m_1 & 1 & 0 \\ 0 & m_2 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

Matriks koefisien ini muncul ketika ada dua variabel tambahan x_3 dan x_4, yang mungkin terikat oleh hubungan tertentu dengan variabel lain dalam sistem.

No.

Date

2.93) (a.) Can every nonsingular $n \times n$ matrix A be written as a product, $A = LU$, where L is a lower triangular matrix and U is an upper triangular matrix?

Ya, setiap matriks segi $n \times n$ yang nonsingular dapat ditulis sebagai hasil dari sebuah perkalian matriks segitiga bawah L dengan matriks segitiga atas U, yang dikenal sebagai Faktorisasi LU. (Diagonal L harus 1).

(b.) If so, what is an algorithm for accomplishing that? If not, give a counterexample to illustrate.

Algoritmanya dapat dijelaskan dengan contoh berikut:
Misal diberikan matriks segi 3×3 definit positif simetrik

$$A = \begin{pmatrix} 2 & -2 & 1 \\ -2 & 3 & 2 \\ 1 & 2 & 10 \end{pmatrix} \underset{E_{31}(1/2)}{\overset{E_{21}(-1)}{\sim}} \begin{pmatrix} 2 & -2 & 1 \\ 0 & 1 & 3 \\ 0 & 3 & 19/2 \end{pmatrix} \overset{E_{32}(3)}{\sim} \begin{pmatrix} 2 & -2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1/2 \end{pmatrix}$$

didapat $U = \begin{pmatrix} 2 & -2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1/2 \end{pmatrix}$ dan $L = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1/2 & 3 & 1 \end{pmatrix}$

Jika diperiksa $\begin{pmatrix} 2 & -2 & 1 \\ -2 & 3 & 2 \\ 1 & 2 & 10 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1/2 & 3 & 1 \end{pmatrix} \begin{pmatrix} 2 & -2 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1/2 \end{pmatrix}$

$$A = LU //$$

2.94) Given an $n \times n$ non singular matrix A and a second $n \times n$ matrix B, what is the best way to compute the $n \times n$ matrix $A^{-1}B$?

Menurut saya, cara terbaik untuk menentukan matriks $n \times n$ $A^{-1}B$ dari matriks segi A dan matriks segi B adalah, pertama dengan mencari invers dari matriks A menggunakan operasi baris dasar atau metode eliminasi Gaussian kemudian mengalikan invers A tersebut dengan matriks B.