ESKİŞEHİR OSMANGAZİ ÜNİVERSİTESİ
Mühendislik Mimarlık Fakültesi
İnşaat Mühendisliği Bölümü
E-Posta: ogu.ahmet.topcu@gmail.com
Web: http://mmf2.ogu.edu.tr/atopcu

Bilgisayar Destekli

Nümerik Analiz

Ders notları 2014

Ahmet TOPÇU

m=n

Üst üçgen matris

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ . & . & . & \dots & . \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ . \\ b_n \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ & & a_{33}^{(3)} & \dots & a_{3n}^{(2)} \\ & & & \dots & . \\ & & & & a_{nn}^{(n-1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(n-1)} \end{bmatrix}$$

Carl Friedrich **GAUSS** (1777 –1855)

5

DENKLEM SİSTEMİ ÇÖZÜMÜ, DİREKT METOTLAR

Basit GAUSS indirgeme metodu

5. BASİT GAUSS[1] İNDİRGEME METODU

n bilinmeyenli n denklemli A x=b sisteminde A ve b verilmiş olsun. x in hesaplanması istenmektedir. det A≠0 ise çözüm vardır ve tektir.

$$\begin{array}{llllll} a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3+ & \dots & +a_{1n}x_n & =b_1 \\ a_{21}x_1 & +a_{22}x_2 & +a_{23}x_3+ & \dots & +a_{2n}x_n & =b_2 \\ a_{31}x_1 & +a_{32}x_2 & +a_{33}x_3+ & \dots & +a_{3n}x_n & =b_3 \\ . & . & . & \dots & . & . \\ a_{n1}x_1 & +a_{n2}x_2 & +a_{n3}x_3+ & \dots & +a_{nn}x_n & =b_n \end{array} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ . & . & . & \dots & . \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ . \\ b_n \end{bmatrix} \qquad (5.1)$$

Basit GAUSS indirgeme metodu bu denklem sistemini n-1 adımda katsayılar matrisi bir üst üçgen olan eşdeğer denklem sistemine indirger:

$$\begin{array}{llllll} a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3+ & \dots & +a_{1n}x_n & =b_1 \\ & a_{22}^{(1)}x_2 & +a_{23}^{(1)}x_3+ & \dots & +a_{2n}^{(1)}x_n & =b_2^{(1)} \\ & & +a_{33}^{(2)}x_3+ & \dots & +a_{3n}^{(2)}x_n & =b_3^{(2)} \\ & & & \dots & . & . \\ & & & & a_{nn}^{(n-1)}x_n & =b_n^{(n-1)} \end{array} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ & & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ & & & \dots & . \\ & & & & a_{nn}^{(n-1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(n-1)} \end{bmatrix}$$

(5.2)

Burada $a_{ij}^{(k)}$ ve $b_i^{(k)}$ sayıları k. adımda A nın değişen sayılarını göstermektedir. Bu denklem sisteminden **aşağıdan yukarı doğru** hesap yapılarak x vektörünün tüm elemanları belirlenir.

İndirgeme metodunun ana ilkesi şudur: 1. adımda A nın 1. kolonunun diyagonalinin altındaki sayılar sıfırlanır. 2. adımda A nın 2. kolonunun diyagonalinin altındaki, 3. adımda A nın 3. kolonunun diyagonalinin altındaki, …, n-1. adımda A nın n-1. kolonunun diyagonalinin altındaki sayılar sıfırlanır.

1. kolonun diyagonali altındaki sayıları sıfırlamak için A nın 1. satırı ***özenle seçilmiş bazı sayılar*** ile çarpılır ve 2., 3., …, n. denklemden çıkarılır. 2. kolonun diyagonali altındaki sayıları sıfırlamak için A nın 2. satırı ***özenle seçilmiş bazı sayılar*** ile çarpılır ve 3., 4., …, n. denklemden çıkarılır, v.s. Bu işlemin nasıl yapılacağı aşağıda adım adım açıklanacaktır.

1. Adım:

Özenle seçilmiş sayılar

1. satır $l_{21}=a_{21}/a_{11}$ ile çarpılır 2. satırdan çıkarılır. Bunun sonucunda a_{21} sıfır olur.
1. satır $l_{31}=a_{31}/a_{11}$ ile çarpılır 3. satırdan çıkarılır. Bunun sonucunda a_{31} sıfır olur.
…
1. satır $l_{n1}=a_{n1}/a_{11}$ ile çarpılır n. satırdan çıkarılır. Bunun sonucunda a_{n1} sıfır olur.

Özenle seçilen sayılarda bölen olarak görünen a_{11} **sayısına pivot eleman** denir. 1.adım sonunda x_1 bilinmeyeni son n-1 denklemden indirgenmiş olur. 5.1 denklem sistemi

İndirgenen bilinmeyen

$$\begin{array}{llllll} a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3+ & \dots & +a_{1n}x_n & =b_1 \\ 0 & +a_{22}^{(1)}x_2 & +a_{23}^{(1)}x_3+ & \dots & +a_{2n}^{(1)}x_n & =b_2^{(1)} \\ 0 & +a_{32}^{(1)}x_2 & +a_{33}^{(1)}x_3+ & \dots & +a_{3n}^{(1)}x_n & =b_3^{(1)} \\ . & . & . & \dots & . & . \\ 0 & +a_{n2}^{(1)}x_2 & +a_{n3}^{(1)}x_3+ & \dots & +a_{nn}^{(1)}x_n & =b_n^{(1)} \end{array} \rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \dots & a_{3n}^{(1)} \\ . & . & . & \dots & . \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \dots & a_{nn}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(1)} \\ . \\ b_n^{(1)} \end{bmatrix} \qquad (5.3)$$

Sıfırlanan sayılar

şeklini alır. Burada $a_{ij}^{(1)}$ ve $b_i^{(1)}$ sayıları 1. adımda A nın değişen yeni sayılarını göstermektedir. Genelleştirilirse, bu sayılar

$$l_{i1}=a_{i1}/a_{11}, \quad a_{ij}^{(1)}=a_{ij}-l_{i1}a_{1j} \quad ve \quad b_i^{(1)}=b_i-l_{i1}b_1, \qquad i=2,3,\dots,n, \quad j=1,2,\dots,n$$

ile hesaplanmaktadır.

[1] Carl Friedrich **GAUSS** (1777 –1855), Alman: 1798 civarı

2. Adım:

1. adım sonunda oluşan 5.3 sisteminin 1. denklemi hariç geriye kalan n-1 denkleminden x_2 bilinmeyeni benzer şekilde yok edilir.

2. satır $l_{32} = a_{32}^{(1)} / a_{22}^{(1)}$ ile çarpılır 3. satırdan çıkarılır. Bunun sonucunda $a_{32}^{(1)}$ sıfır olur.

2. satır $l_{42} = a_{42}^{(1)} / a_{22}^{(1)}$ ile çarpılır 4. satırdan çıkarılır. Bunun sonucunda $a_{42}^{(1)}$ sıfır olur.

...

Özenle seçilmiş sayılar

2. satır $l_{n2} = a_{n2}^{(1)} / a_{22}^{(1)}$ ile çarpılır n. satırdan çıkarılır. Bunun sonucunda $a_{n2}^{(1)}$ sıfır olur.

Özenle seçilen sayılarda bölen olarak görünen $a_{22}^{(1)}$ **sayısına pivot eleman** denir. 2.adım sonunda x_2 bilinmeyeni son n-2 denklemden indirgenmiş olur. 5.3 denklem sistemi

$$
\begin{array}{cccccc}
a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3+ & \dots & +a_{1n}x_n & =b_1 \\
0 & +a_{22}^{(1)}x_2 & +a_{23}^{(1)}x_3+ & \dots & +a_{2n}^{(1)}x_n & =b_2^{(1)} \\
0 & +0 & +a_{33}^{(2)}x_3+ & \dots & +a_{3n}^{(2)}x_n & =b_3^{(2)} \\
. & . & . & \dots & . & . \\
0 & +0 & +a_{n3}^{(2)}x_3+ & \dots & +a_{nn}^{(2)}x_n & =b_n^{(2)}
\end{array}
\qquad
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\
0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\
0 & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\
. & . & . & \dots & . \\
0 & 0 & a_{n3}^{(2)} & \dots & a_{nn}^{(2)}
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix}
=
\begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(2)} \end{bmatrix}
\qquad (5.4)
$$

İndirgenen bilinmeyenler

Sıfırlanan sayılar

şeklini alır. Burada $a_{ij}^{(2)}$ ve $b_i^{(2)}$ sayıları 2. adımda A nın değişen yeni sayılarıdır. Genelleştirilirse, bu sayılar

$$
l_{i2} = a_{i2}^{(1)} / a_{22}^{(1)}, \quad a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i2}a_{2j}^{(1)} \quad ve \quad b_i^{(2)} = b_i^{(1)} - l_{i2}b_2^{(1)}, \qquad i = 3,4, \dots, n, \quad j = 2, 3, \dots, n
$$

ile hesaplanmaktadır.

Diğer adımlar:

2. adım sonunda oluşan 5.4 sisteminin 1. ve 2. denklemi hariç, geriye kalan n-2 denklemindeki x_3, x_4, …, x_{n-1} bilinmeyenleri benzer şekilde yok edilir. **n-1. adım sonunda** katsayılar matrisi bir üst üçgen olan 5.2 eşdeğer denklem sistemine dönüşür. Bu denklem sisteminden aşağıdan yukarı doğru hesap ile x_1, x_2, …, x_n bilinmeyenleri sırayla bulunur:

İndirgenen bilinmeyenler

$$
\begin{array}{cccccc}
a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3+ & \dots & +a_{1n}x_n & =b_1 \\
 & a_{22}^{(1)}x_2 & +a_{23}^{(1)}x_3+ & \dots & +a_{2n}^{(1)}x_n & =b_2^{(1)} \\
 & & +a_{33}^{(2)}x_3+ & \dots & +a_{3n}^{(2)}x_n & =b_3^{(2)} \\
 & & & \dots & . & . \\
 & & & & a_{nn}^{(n-1)}x_n & =b_n^{(n-1)}
\end{array}
\qquad
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\
 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\
 & & a_{33}^{(3)} & \dots & a_{3n}^{(2)} \\
 & & & \dots & . \\
 & & & & a_{nn}^{(n-1)}
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix}
=
\begin{bmatrix} b_1 \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(n-1)} \end{bmatrix}
$$

Hesap yönü

$$
\begin{aligned}
x_1 &= (b_1 - a_{12}x_2 - a_{13}x_3 - \dots - a_{1n}x_n) / a_{11} \\
x_2 &= (b_2^{(1)} - a_{23}^{(1)}x_3 - \dots - a_{2n}^{(1)}x_n) / a_{22}^{(1)} \\
x_3 &= (b_3^{(2)} - \dots - a_{3n}^{(2)}x_n) / a_{33}^{(2)} \\
&\dots \\
x_n &= b_n^{(n-1)} / a_{nn}^{(n-1)}
\end{aligned}
$$

Sıfırlanan sayılar

a_{11}, $a_{22}^{(1)}$, $a_{33}^{(2)}$,…, $a_{nn}^{(n-1)}$ diyagonal elemanlarına **pivot eleman** denir.

Gözlem:

- A katsayılar matrisi **n-1 adım** sonunda bir üst üçgen matrise dönüştürülmektedir.
- Oluşan yeni denklem sistemi verilen denklem sistemi ile eşdeğerdir.
- Tüm işlemler A, b ve x matrisi üzerinde yapılmaktadır. A ve b nın ilk değerleri kaybolmaktadır.
- Pivot elemanlar ≠ 0 olmalıdır, aksi halde çözüme devam edilemez.
- Kasayılar matrisi A nın determinantı indirgenmiş sistemin determinantına eşittir: $\det A = a_{11} \cdot a_{22}^{(1)} \cdot a_{33}^{(2)} \cdot .. \cdot a_{nn}^{(n-1)}$
- Pivot elemanlardan biri sıfır ise det A=0 olur. Bu ise A nın satır veya kolonlarının doğrusal bağımlı olduğu anlamına gelir.
- det A=0 ise A tekildir, çözüm yoktur veya tek değildir. Rank A <n dir.
- det A ≠ 0 ise çözüm vardır ve tektir. A düzenlidir (regülerdir). Rank A =n dir.

Örnek:

$$
\begin{aligned}
6x_1 - 2x_2 + 2x_3 + 4x_4 &= 16\\
12x_1 - 8x_2 + 6x_3 + 10x_4 &= 26\\
3x_1 - 13x_2 + 9x_3 + 3x_4 &= -19\\
-6x_1 + 4x_2 + x_3 - 18x_4 &= -34
\end{aligned}
\rightarrow
\begin{bmatrix} 6 & -2 & 2 & 4\\ 12 & -8 & 6 & 10\\ 3 & -13 & 9 & 3\\ -6 & 4 & 1 & -18 \end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{bmatrix}
=
\begin{bmatrix} 16\\ 26\\ -19\\ -34 \end{bmatrix}
\rightarrow \underline{A}\underline{x} = \underline{b}, \qquad \underline{x} = ?
$$

Tüm işlemler <u>A</u> matrisi üzerinde yapılacaktır. Diyagonal altındaki sayıların sıfır olacağı bilindiğinden bu hücrelere l_{ij} değerleri saklanacaktır.

1. Adım:

Pivot eleman

$$
\begin{bmatrix} 6 & -2 & 2 & 4\\ 2 & -4 & 2 & 2\\ 0.5 & -12 & 8 & 1\\ -1 & 2 & 3 & -14 \end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{bmatrix}
=
\begin{bmatrix} 16\\ -6\\ -27\\ -18 \end{bmatrix}
$$

$l_{21}=12/6=2$

$l_{31}=3/6=0.5$

$l_{41}= - 6/6= -1$

$l_{21}=2$ sayısı 1.satır ile çarpıldı, 2.satırdan çıkartıldı.

$l_{31}=0.5$ sayısı 1.satır ile çarpıldı, 3.satırdan çıkartıldı.

$l_{41}= -1$ sayısı 1.satır ile çarpıldı, 4.satırdan çıkartıldı.

1. diyagonal altındaki elemanlar pivot elemana bölündü

2. Adım:

Pivot eleman

$$
\begin{bmatrix} 6 & -2 & 2 & 4\\ 2 & -4 & 2 & 2\\ 0.5 & 3 & 2 & -5\\ -1 & -0.5 & 4 & -13 \end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{bmatrix}
=
\begin{bmatrix} 16\\ -6\\ -9\\ -21 \end{bmatrix}
$$

$l_{32}=-12/(-4)=3$

$l_{42}=2/(-4)=-0.5$

$l_{32}=3$ sayısı 2.satır ile çarpıldı, 3.satırdan çıkartıldı.

$l_{42}= -0.5$ sayısı 2.satır ile çarpıldı, 4.satırdan çıkartıldı.

2. diyagonal altındaki elemanlar pivot elemana bölündü

3. Adım:

Pivot eleman

$$
\begin{bmatrix} 6 & -2 & 2 & 4\\ 2 & -4 & 2 & 2\\ 0.5 & 3 & 2 & -5\\ -1 & -0.5 & 2 & -3 \end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{bmatrix}
=
\begin{bmatrix} 16\\ -6\\ -9\\ -3 \end{bmatrix}
$$

$l_{43}=4/2=2$

$l_{43}= 2$ sayısı 3.satır ile çarpıldı, 4.satırdan çıkartıldı.

3.diyagonal altındaki elemanlar pivot elemana bölündü

İndirgenmiş denklem sistemi ve aşağıdan yukarıya doğru hesap:

$$
\begin{bmatrix} 6 & -2 & 2 & 4\\ & -4 & 2 & 2\\ & & 2 & -5\\ & & & -3 \end{bmatrix}
\begin{bmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{bmatrix}
=
\begin{bmatrix} 16\\ -6\\ -9\\ -3 \end{bmatrix}
$$

Hesap yönü

$$
\left.
\begin{aligned}
x_1 &= (16-(-2)\cdot 1 - 2\cdot(-2) - 4\cdot 1)/6 = 3\\
x_2 &= (-6 - 2\cdot(-2) - 2\cdot 1)/(-4) = 1\\
x_3 &= (-9-(-5)\cdot 1)/2 = -2\\
x_4 &= -3/(-3) = 1
\end{aligned}
\right\}
\underline{x} = \begin{bmatrix} 3\\ 1\\ -2\\ 1 \end{bmatrix}
$$

Gözlem:

- Tüm işlemler <u>A</u> , <u>b</u> ve <u>x</u> matrisi üzerinde yapılmıştır. <u>A</u> ve <u>b</u> nın verilmiş ilk değerleri kaybolmuştur.
- <u>A</u> katsayılar matrisi **n-1=4-1=3 adım** sonunda bir üst üçgen matrise dönüşmüştür.
- Oluşan yeni denklem sistemi verilen denklem sistemi ile eşdeğerdir.
- Pivot elemanlar ≠ 0 dır.
- Katsayılar matrisi <u>A</u> nın determinantı indirgenmiş sistemin determinantına eşittir. $\det \underline{A} = 6\cdot(-4)\cdot 2\cdot(-3) = 144 \neq 0$ dır.
- det <u>A</u> ≠ 0 olduğundan <u>A</u> düzenli(regüler) bir matristir, satır ve kolonları doğrusal bağımlı değildir, rank <u>A</u>=n=4 tür.
- Tek çözüm vardır ve bulunmuştur.

Pivot eleman seçimi ve satır değiştirme

l_{ij} sayılarının hesabında bölen olarak karşımıza çıkan $a_{11},\ a_{22}^{(1)},\ a_{33}^{(2)},\ldots,\ a_{nn}^{(n-1)}$ sayılarına "**pivot eleman**", bu elemanların bulunduğu satır ve kolona "**pivot satırı**" ve "**pivot kolonu**" denir. Pivot kelimesi kilit, rol oynayan, yönlendiren olarak tercüme edilebilir. Pivot eleman hesabın kaderini belirler. Çünkü, indirgemenin sürdürülebilmesi için pivot elemanların sıfırdan farklı olması gerekir, aksi durumda hesaba devam edilemez. Örneğin

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ -2 & 1 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix},\quad \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ -2 & 1 & -1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix},\quad \begin{bmatrix} -2 & 1 & -1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$$

denklem sistemlerinin üçünün de çözümü aynı; x_1=1, x_2=3, x_3=-2 dır. Çünkü farklı gibi görülen her üç denklem sistemi gerçekte birbirinin aynıdır, sadece satırlarının yerleri değiştirilmiştir. Satırların yer değiştirmesi çözümü değiştirmediğinden çözüm de aynı kalır.

Ancak; birinci denklem sistemi çözülmek istenirse, daha ilk adımda, çıkmaza girilir. Çünkü 1. diyagonal eleman(yani pivot), sıfırdır. İkinci denklem sisteminde 1. diyagonal eleman(pivot) sıfırdan farklıdır, indirgemenin ilk adımı sorunsuz yürütülebilir. Üçüncü denklem sisteminde de ilk adım sorun yaratmaz.

Pivot elemanın sıfır olması halinde, örneğin k. Adımda, a_{kk}=0 ise; $\underline{A}$ matrisinin k. Kolonunda diyagonalin altındaki sayılar arasında pivot eleman aranır ve k. satır ile sonraki satırlardan biri değiştirilir, pivot elemanın sıfırdan farklı olması sağlanır ve hesaba devam edilir.

Sıfırdan farklı pivot elemanın bulunamaması durumunda; örneğin k. Adımda diyagonal ve diyagonalin altındaki tüm elemanlar sıfır ise, $a_{kk=}\ a_{(k+1)k}$=,..., = a_{nk}=0; satır değiştirmek de bir işe yaramaz. Bunun anlamı; $\underline{A}$ matrisinin tekil, yani det $\underline{A}$=0 olduğu, çözümün bulunmadığı veya sayısız çözümün olduğudur.

Pivot elemanın sıfırdan farklı fakat mutlak değerce çok küçük olması da nümerik sorun yaratır. l_{ij} sayıları $\underline{A}$ nın elemanları pivot elemana bölünerek hesaplandığından, pivot elemanın çok küçük olması bölümün çok büyük bir sayı olmasına, bununla çarpılan/toplanan sayıların daha da büyümesine ve yuvarlama hatalarına neden olur. Büyük denklem sistemlerinde bu işlemler milyonlarca defa yapıldığından, yuvarlama hataları birike birike sonucun tamamen yanlış hesaplanmasına neden olur.

Yuvarlama hatalarını azaltmak için, gerekirse her adımda, satırlara yer değiştirilir. Mutlak değerce en büyük sayı diyagonal ve diyagonalin altındaki sayılar arasında aranır, bu sayının bulunduğu satır ile pivot satırı değiştirilir. Örnek: yukarıdaki 1. denklem sistemi çözülürken 1. adımda 1. diyagonal ve diyagonalin altında mutlak değerce en büyük sayı 3. satırdadır: |-2|. Bu nedenle 1. denklem ile 3. denkleme yer değiştirilir ve indirgemeye devam edilir. Böylece pivot elemanların en büyük sayı olması, olası yuvarlama hatalarının en az olması sağlanmış olur.

Determinant:

İndirgeme sonrası oluşan yeni denklem sistemi verilen denklem sistemi ile eşdeğerdir. Bu nedenle katsayılar matrisi $\underline{A}$ nın determinantı indirgenmiş sistemin determinantına eşittir. Satırlara yer değiştirmek determinantın değerini değiştirmez fakat işaretini değiştirir. Pivot arama nedeniyle indirgeme sırasında p defa satır değiştirilmiş ise determinant da p defa işaret değiştireceğinden

$$\det \underline{A} = (-1)^p \cdot a_{11} \cdot a_{22}^{(1)} \cdot a_{33}^{(2)} \cdot .. \cdot\ a_{nn}^{(n-1)} \tag{5.5}$$

olur.

İndirgeme tamamlanabilmiş, n-1 adım sonra denklem sisteminin katsayılar matrisi üst üçgen olan eşdeğer denklem sistemine dönüştürülebilmiş ise determinant (5.5) den hesaplanır ve det $\underline{A}$ ≠0 dır. Ancak, indirgeme tamamlanamamış ise; örneğin k. Adımda pivot eleman bulunamamış ise, indirgeme işlemine devam edilemez, det $\underline{A}$=0 dır.

det $\underline{A}$ ≠0 ise: $\underline{A}$ matrisi tekil değildir, düzenlidir(regülerdir) denir. Çözüm vardır ve tektir.

det A=0 ise: $\underline{A}$ matrisi tekildir(singülerdir), düzensizdir denir. Çözüm yoktur veya sonsuz çözüm vardır. Denklemler uyumlu ise sonsuz çözüm vardır, uyumsuz ise çözüm yoktur.

Rank:

İndirgeme tamamlanabilmiş ise rank $\underline{A}$=n dir. $\underline{A}$ nın satır ve kolonları düzenlidir. k.adımda sıfırdan farklı pivot eleman bulunamamış ise, $\underline{A}$ nın rankı r=k-1 dir. $\underline{A}$ nın r satırı doğrusal bağımsızdır geriye kalan d=n-r satır veya kolonları doğrusal bağımlıdır; d ye rank artığı denir.

Örnek:

$$\begin{bmatrix} 6 & -2 & 2 & 4 \\ 12 & -8 & 6 & 10 \\ 3 & -13 & 9 & 3 \\ -6 & 4 & 1 & -18 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 16 \\ 26 \\ -19 \\ -34 \end{bmatrix}, \qquad \underline{x} = ?$$

Çözüm için her adımda pivot arama yapılacak, gerekirse satırlar değiştirilecektir. El hesabında ondalık işaretinden sonra 6 hane yürütülecektir.

1. Adım: 1. diyagonal ve altındaki mutlak değerce en büyük sayı 2. satırda |12| dir. 1. ve 2. satırlar değiştirilecek ve indirgeme yapılacaktır.

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 6 & -2 & 2 & 4 \\ 3 & -13 & 9 & 3 \\ -6 & 4 & 1 & -18 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ 16 \\ -19 \\ -34 \end{bmatrix}$$

1. ve 2.satır değiştirildikten sonra.

Pivot eleman

$l_{21}=6/12=0.5$

$l_{31}=3/12=0.25$

$l_{41}=-6/12=-0.5$

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 0.5 & 2 & -1 & -1 \\ 0.25 & -11 & 7.5 & 0.5 \\ -0.5 & 0 & 4 & -13 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ 3 \\ -25.5 \\ -21 \end{bmatrix}$$

l_{21}= 0.5 sayısı 1.satır ile çarpıldı, 2.satırdan çıkartıldı.

l_{31}= 0.25 sayısı 1.satır ile çarpıldı, 3.satırdan çıkartıldı.

l_{41}= -0.5 sayısı 1.satır ile çarpıldı, 4.satırdan çıkartıldı.

1. diyagonal altındaki elemanlar pivot elemana bölündü

2. Adım: 2. diyagonal ve altındaki mutlak değerce en büyük sayı 3.satırda|-11|dir. 2. ve 3. satırlar değiştirilecek ve indirgeme yapılacaktır.

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 0.25 & -11 & 7.5 & 0.5 \\ 0.5 & 2 & -1 & -1 \\ -0.5 & 0 & 4 & -13 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ -25.5 \\ 3 \\ -21 \end{bmatrix}$$

2. ve 3.satır değiştirildikten sonra.

Pivot eleman

$l_{32}=2/(-11)= -0.181818\ldots$

$l_{42}=0/(-11)=0$

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 0.25 & -11 & 7.5 & 0.5 \\ 0.5 & -0.181818 & 0.363635 & -0.909091 \\ -0.5 & 0 & 4 & -13 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ -25.5 \\ -1.636359 \\ -21 \end{bmatrix}$$

l_{32}= -0.181818 sayısı 2.satır ile çarpıldı, 3.satırdan çıkartıldı.

l_{42}= 0 olduğundan işlem gerekmez.

2. diyagonal altındaki elemanlar pivot elemana bölündü

3. Adım: 3. diyagonal ve altındaki mutlak değerce en büyük sayı 4. satırda |4| tür. 3. ve 4. satırlar değiştirilecek ve indirgeme yapılacaktır.

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 0.25 & -11 & 7.5 & 0.5 \\ -0.5 & 0 & 4 & -13 \\ 0.5 & -0.181818 & 0.363635 & -0.909091 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ -25.5 \\ -21 \\ -1.636359 \end{bmatrix}$$

3. ve 4.satır değiştirildikten sonra.

Pivot eleman

l_{43}=0.363635/4= 0.090909

$$\begin{bmatrix} 12 & -8 & 6 & 10 \\ 0.25 & -11 & 7.5 & 0.5 \\ -0.5 & 0 & 4 & -13 \\ 0.5 & -0.181818 & 0.090909 & 0.272726 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 26 \\ -25.5 \\ -21 \\ 0.272730 \end{bmatrix}$$

l_{43}= 0.090909 sayısı 3.satır ile çarpıldı, 4. satırdan çıkarıldı.

3. diyagonal altındaki eleman pivot elemana bölündü

Aşağıdan yukarı doğru hesap:

$$\left.\begin{aligned} x_1 &= (26-(-8)\cdot 1.000034 - 6\cdot(-1.999951) - 10\cdot 1.000015)/12 = 2.999986 \\ x_2 &= (-7.5\cdot(-1.999951) - 0.5\cdot 1.000015)/(-11) = 1.000034 \\ x_3 &= (-21-(-13)\cdot 1.000015)/4 = -1.999951 \\ x_4 &= 0.272730/0.272726 = 1.000015 \end{aligned}\right\} \underline{x} = \begin{bmatrix} 2.999986 \\ 1.000034 \\ -1.999951 \\ 1.000015 \end{bmatrix}$$

Gerçek çözüm: $\underline{x}$={3 1 -2 1} dir.

Hata vektörü:

$$\begin{bmatrix} 3 \\ 1 \\ -2 \\ 1 \end{bmatrix} - \begin{bmatrix} 2.999986 \\ 1.000034 \\ -1.999951 \\ 1.000015 \end{bmatrix} = \begin{bmatrix} 0.000014 \\ -0.000034 \\ -0.000049 \\ -0.000015 \end{bmatrix}$$

Görüldüğü gibi, pivot arama yapılmasına, 6 ondalık hane yürütülmesine rağmen, az da olsa hata vardır. Bilgisayarda ondalık sayı hesabında genelde 15-16 hane kullanılır. Denklem sistemi 15-16 hane kullanarak bilgisayarda çözülürse hata çok azalacak, fakat belki gene de tam sıfır olmayacaktır.

Bu örneğin gerçek çözümünü bildiğimiz için hata vektörünü de hesaplayabildik. Uygulamada ise gerçek çözüm bilinmez(bilinseydi hesaplamaya gerek kalmazdı).

Gerçek çözüm bilinmediğine göre aşağıdaki sorular gündeme gelmektedir:

- Hesaplanan çözüm doğru mudur?
- Hata nasıl ölçülecektir?
- Hata ne kadardır?
- Hata kaynağı nedir?

Bu soruların cevabı verilemeyen çözüm doğru varsayılamaz. İzleyen sayfalarda bu sorular irdelenecektir.

Nümerik hesaplarda hata kaynakları, hatanın ölçüsü

Fiziksel bir problemin sayısal olarak çözümünde aşağıdaki nedenlerle az ya da çok hatalar oluşur.

1. Modelleme hatası: Fiziksel problemin çözülebilmesi için oluşturulan matematik model gerçek problemi tam olarak yansıtamıyordur. Varsayılan geometri ve sınır değerler farklıdır.

2. Çözüm yönteminden doğan hatalar: Nümerik çözüm için seçilen metot fiziksel problem veya onun modeline uygun değildir.

3. Program veya programlama hataları: Çözümde kullanılan program kod hataları içeriyordur veya problemin çözümüne uygun değildir.

4. Bilgisayar hataları: Bilgisayarda sayılar yuvarlatılarak depolanır, sonsuz terimli serilerden hesaplanan değerler için sonlu terim kullanılır. Nadiren de olsa, donanım hatalı olabilir. Voltajda ani değişimler farkında olunamayan hatalı sonuçlar doğurabilir.

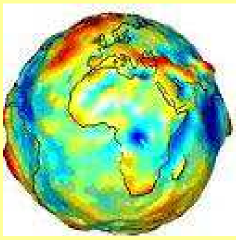
5. İnsani hatalar: Programı kullanan kişi programı yeterince tanımıyordur, gerekli ön ayarları yapmamış veya hatalı yapmıştır. Verileri hatalı girmiştir veya yuvarlatarak vermiştir. Sonuçları yanlış yorumlamıştır.

Basit bir fiziksel problem örneği ile yukarıda sayılan hataların bazılarına açıklık getirmeğe çalışalım.

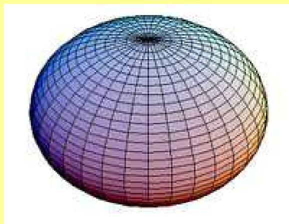
Fiziksel problem: Dünya yüzey alanının hesabı

Matematik model: Fiziksel problemin matematik modeli öncelikle a) Gerçeğe en yakın olmalı b) basit olmalı c) Hesaplanabilir olmalı

Dünyanın ideal bir geometrisi yoktur. Yuvarlaktır deriz, küre olduğunu kastederiz. Uzaydan çekilmiş fotoğrafa baktığımızda küre gibi görünmektedir. Fakat biliyoruz ki; yüksek dağlar ve derin çukurlar vardır, kutuplarda basık, ekvatorda şişik bir geometriye sahiptir(geoid) .


Fiziksel problem: pürüzlü yüzey, düzgün olmayan geometri


Küre model: uzaydan görünüş


Daha gerçek model: Geoid

Basitliği nedeniyle, dünyayı küre olarak modelleyelim:

Kürenin yüzey alanı: $A=4\,\pi\, r^2$

r yarıçapını ve π sayısını ne alalım? Ekvatorda r=6378.1 km, kutuplarda r=6356.8 km dir. Yarı çapı ortalama değer alarak modelleyelim, r=(6378.1+6346.8)/2=6367.5 km alalım.

$\pi=3.14$ alalım.

$A=4 \cdot 3.14 \cdot 6367.5^2=509245907$ km^2 olur.

Bulduğumuz bu değerin tam doğru olmadığı, yaklaşık olduğu açıktır. Doğru değerin hesaplanamayacağı da açıktır. Hata kaynağı: a) Küre varsayımı doğru değildir. b) Ekvator ve kutupta ölçülmüş yarıçaplar ölçüm hatası içerir. c) Yarıçapı ortalama değer alarak bir hata daha oluşturduk. d) $\pi=3.14$ değerini yuvarlatarak hata oluşturduk. e) Hesaplanan A değerinin yuvarladık, virgülden sonraki hanelerini attık.

Hata ne kadardır? Gerçek değeri bilemediğimize göre, hatanın ne kadar olduğunu da bilememekteyiz. Daha gerçekçi modellerle (geoid) hesaplanmış değer kaynaklarda A=510072000 km^2 olarak verilmektedir. Bu değer de matematik anlamda doğru olamaz, çünkü gene bir takım varsayımlar yapılarak hesaplanmıştır. Ancak, burada kullandığımız küre modelinden daha doğrudur. A=510072000 km^2 değerini doğru kabul ederek hata analizi yapabiliriz. Önce hata ölçüsü tanımı yapalım.

Tanım: bir x fiziksel büyüklüğünün hesaplanmış değeri x_{hesap} ve gerçek değeri $x_{gerçek}$ olsun. Bu iki değer arasındaki farka **hata** veya **mutlak hata** denir.

Hata(mutlak): $\Delta x = x_{hesap} - x_{gerçek}$

Mutlak hatanın gerçek değere oranına **hata yüzdesi** veya **bağıl** hata denir.

Hata yüzdesi(bağıl hata): $\varepsilon = \dfrac{\Delta x}{x_{gerçek}} = \dfrac{x_{hesap} - x_{gerçek}}{x_{gerçek}}$

Hata yüzdesi (bağıl hata) gerçek değerin birim hatası olarak da yorumlanabilir: $\varepsilon = \dfrac{\Delta x}{x_{gerçek}} \rightarrow \Delta x = \varepsilon\, x_{gerçek}$

Şimdi dünyanın yüzey alanı ile ilgili hesabımıza dönelim, hata analizi yapalım: $A_{gerçek}$=510072000 km^2 ve A_{hesap}=509245907 km^2 alarak:

Hata(mutlak): $\Delta A = A_{hesap} - A_{gerçek} = 509245907 - 510072000 = -826093 \text{km}^2$

Hata yüzdesi(bağıl hata): $\varepsilon = \dfrac{\Delta A}{A_{gerçek}} = \dfrac{A_{hesap} - A_{gerçek}}{A_{gerçek}} = \dfrac{-826093}{510072000} \approx -0.0002$

Buradaki eksi işareti hesaplanan alanın gerçek alandan 826093 km^2 daha küçük kaldığını gösterir, Hata %0.02 kadardır. 826093 km^2 değeri hatanın çok büyük olduğunu düşündürebilir. Fakat hata yüzdesi sadece %0.02 (on binde 2) olduğundan, küçük kabul edilmelidir.

Bu basit örnekte de görüldüğü gibi, hata yüzdesi(bağıl hata) daha iyi fikir vermektedir. Nümerik hesaplarda **hata** yerine **hata yüzdesi(bağıl hata)** ile çalışmak daha uygun olacaktır.

Gerçek değerin bilinmediği fiziksel problemlerde hata kontrolü

Nümerik analiz problemlerinde genelde gerçek çözüm bilinmez. Fakat hesaplanan değerin veya değerlerin sağlaması gereken bir koşul veya koşullar vardır. Hesaplanan değerin bu koşulları sağlayıp sağlamadığına bakılarak hesaplanan değerin veya değerlerin doğruluğu veya hatası kontrol edilir. Basit bir örnek verelim:

$3x^2 - 1=0$ denklemini sağlayan x nedir? $x_1= \pm 0.5774$ olarak hesapladığımızı varsayarak hatayı hesaplayalım. Bu değer $3x^2 - 1=0$ denklemini sağlamak zorundadır: $3(0.5774)^2 - 1=0.000172 \neq 0$ olduğundan, sağlamıyor. Hata 0.000172 dir.

Yuvarlama hatasının nedeni nedir, nasıl azaltılabilir?

Hesap makineleri ve bilgisayarlar bir sayıyı depolamak için belli sayıda hane kullanırlar, 8, 10 veya 16 hane gibi. Makineye verilen sayının hane sayısı makinenin depolayabileceği hane sayısından daha fazla ise, sayının sonundaki bazı rakamları kaybolur. Örnekleyelim: kullandığınız el hesap makinesine 1234567890 sayısını yazın ve eşit tuşuna basın. Makineniz 10 hane gösterebiliyorsa ekranda 1234567890 göreceksiniz. Şimdi 123456789012345 yazın ve eşit tuşuna basın. Muhtemelen $1.23456789 \cdot 10^{14}$ göreceksiniz. Makineye verdiğiniz sayı bu değildi, son haneleri kayboldu!

Verdiğiniz: 123456789012345
Makinenin depoladığı: $1.23456789 \cdot 10^{14}$ =123456789000000
Hata: 123456789012345 – 123456789000000=12345
Hata yüzdesi: 12345/123456789012345=10^{-10}

0.123456789012345 sayısını yazıp eşit tuşuna basın, sonuç: 0.123456789 olacaktır. Son haneler gene kayboldu!

1/3 yazın eşit tuşuna basın, sonuç: 0.333333333 olacaktır. Halbuki 1/3=0.33333333333333……
tür. Ondalık işaretinden sonra sonsuz sayıda 3 rakamı vardır. Makinenizin sonsuz hanesi yoktur, depolayabildiğinden daha fazla haneyi kesmektedir.

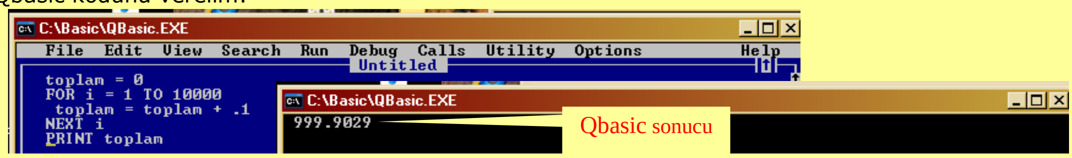
π sayısını makineniz muhtemelen π=3.141592654 olarak gösteriyordur, son hane yuvarlatılmıştır. 30 haneli değeri π=3.14159265358979323846264338328 dir, gerçek değeri bugüne kadar hesaplanamamıştır!

Yukarıdaki örnekleri, varsa, cep telefonunuzun veya saatinizin hesap makinesi ile de deneyiniz!

Bilgisayara 10 tabanlı sayılar veririz. Bilgisayar ise iki tabanlı sayılarla (sıfır ve bir ile) çalışır. Verilen ondalık sayıları iki tabanlı (binary) sayıya çevirir ve depolar. Örnek: 2010 sayısı bilgisayarda 11111011010 olarak depolanır. Çevirme nedeniyle hane (bit) kaybı olur. Buna yönelik çarpıcı bir örnek verelim: 0.1 ondalık sayısının 10000 defa toplamı

$$Toplam = \sum_{i=1}^{10000} 0.1 = 0.1 \cdot 10000 = 1000$$

olması gerektiği açıktır. Kullandığınız programlama dili ile bu toplamı bilgisayara yaptırarak şaşırtıcı sonucu görünüz! Qbasic kodunu verelim:



Bilgisayarın verdiği sonuç 999.9029 hatalıdır! Bilsayar 0.1 sayısını hatalı depolamıştır. Demek ki bilgisayar sayıları hatasız depolayamıyor, yuvarlıyor ve kesiyor. Yuvarlama hatalarını tamamen önlemek mümkün değildir. Milyarlarca sayı dört işleme tabi tutulduğunda bu hataların giderek büyüyebileceği, belki de hesaplanmak istenen sonucun tamamen yanlış olmasına neden olacağı açıktır. Yuvarlanmış bir sayı hataya neden olurken yuvarlanmış bir sayı ile hesaplanan bir diğer sayı yuvarlandığında hata büyüyebilir de azalabilir de.

Hataları elden geldiğince azaltmak için:

1.Makine veya bilgisayarda mümkün olduğunca çok haneli hesap yapılmalı: El hesap makinelerinde hane sayısını değiştirmek mümkün değildir, fakat satın alırken çok haneli olanı seçilebilir. Bilgisayarda hane sayısını, kullanılan yazılıma bağlı olarak, ayarlama imkânı vardır: Tek hassasiyet(4 Byte SINGLE precision) yerine çift hassasiyet(8 Byte DOUBLE precision) hatta dörtlü hassasiyet (16 Byte QUAD precision**,** EXTENDED) kullanarak.

2.Büyük sayılar ile diğer büyük sayıların çarpımı önlenmelidir:

- Büyük denklem sistemlerinin çözümünde uygun pivot eleman seçilerek bunu gerçekleştirmek mümkündür.
- a, b, c sayılarından d=ab/c hesaplanacak varsayalım. Sayıların büyük veya küçük olduğuna bağlı olarak hesabın aşağıdaki şekilde yapılması sayı taşma ve yuvarlama riskini azaltır:

  |a| ve |b| sayılarından biri büyük ve diğerleri küçükse d=(ab)/c
  |b| ve |c| sayılarının her ikisi de büyük veya her ikisi de küçükse d=a(b/c)
  |a| ve |c| sayılarının her ikisi de büyük veya her ikisi de küçükse d=(a/c)b

  Bilgisayar önce parantez içindeki ifadeyi hesapladığından, parantez içi 1 sayısına yakın çıkacaktır.
- e^x gibi ifadeler yuvarlama ve sayı taşması açısından risklidir, uygun hesap sırası seçilmelidir, örnek:

  $a=e^x/e^t$ yerine $a=e^{(x-t)}$, $a=y^n/e^{nx}$ yerine $a=(y/e^x)^n$ kullanılmalıdır.
- Yüksek dereceden polinomlar sayı taşması ve yuvarlama hataları açısından risklidir, örnek:

  $y=a_1x^5+ a_2x^4+ a_3x^3+ a_4x^2+ a_5x+ a_6$ yerine $y=((((a_1x+a_2)x+a_3)x+a_4)x+a_5)x+a_6$ daha uygundur.

Vektör ve matris normları, kondisyon sayısı kavramı

Denklem sistemlerinin çözümünde hata analizi için norm ve kondisyon sayıları kullanılır. Norm ve kondisyon sayısı çözümün ne denli sağlıklı olduğu bilgisini verebilir.

Norm bir sayıdır ve $\|\ \|$ işareti ile gösterilir. Mesala $\underline{x}$ vektörünün normu $\|\underline{x}\|$ ile gösterilir. Uygulamada sıkça kullanılan vektör ve matris normları aşağıda verilmiştir.

Vektör normları:

$\underline{x}=[x_1 \quad x_2 \quad \dots \quad x_n]$ vektörünün normları:

$\|\underline{x}\|_1=|x_1|+|x_2|+\dots+|x_n|$ (l_1 normu denir)

$\|\underline{x}\|_2=\sqrt{\underline{x}\,\underline{x}^T}=\sqrt{x_1^2+x_2^2+\dots+x_n^2}$ (l_2 normu veya Öklid normu veya $\underline{x}$ in uzunluğu denir)

$\|\underline{x}\|_\infty=\max|x_i|$ (l_∞ normu veya maximum norm denir)

Örnek: $\underline{x}_{gerçek}=[2 \quad -3 \quad 0 \quad 1 \quad -4]$ vektörünün hesaplanmış vektörü

$\underline{x}_{hesap}=[2.0001 \quad -2.9998 \quad 10^{-6} \quad 0.9999 \quad -4.0002]$ olsun.

Normlar:

$$\|\underline{x}_{gerçek}\|_1=2+3+0+1+4=10$$
$$\|\underline{x}_{gerçek}\|_2=\sqrt{4+9+0+1+16}=5.477226$$
$$\|\underline{x}_{gerçek}\|_\infty=\max(2,3,0,1,4)=4$$

$$\|\underline{x}_{hesap}\|_1=2.0001+2.9998+10^{-6}+0.9999+4.0002=10.000001$$
$$\|\underline{x}_{hesap}\|_2=\sqrt{2.00001^2+2.9998^2+0^2+(10^{-6})^2+0.9999^2+4.0002^2}=5.477289$$
$$\|\underline{x}_{hesap}\|_\infty=\max(2.00001,2.9998,0,0.9999,4.0002)=4.0002$$

Normların oranları:

$$\frac{\|\underline{x}_{gerçek}\|_1}{\|\underline{x}_{hesap}\|_1}=\frac{10}{10.000001}=0.9999999$$
$$\frac{\|\underline{x}_{gerçek}\|_2}{\|\underline{x}_{hesap}\|_2}=\frac{5.477226}{5.477289}=0.9999885$$
$$\frac{\|\underline{x}_{gerçek}\|_\infty}{\|\underline{x}_{hesap}\|_\infty}=\frac{4}{4.0002}=0.99995000$$

Dır. $\underline{x}_{hesap}$ tam doğru olsaydı bu normların hepsi de 1 olurdu. Demek ki herhangi bir normun oranı 1 e ne denli yakın ise $\underline{x}_{hesap}$ o denli doğrudur. Örneğimizde oranlar 1 e çok yakın olduğundan $\underline{x}_{hesap}$ doğru kabul edilebilir. Burada verilenler dışında başka vektör normları da vardır. Uygulamada l_2 (Öklid) normu veya basitliği nedeniyle l_∞ (maksimum) normu genelde tercih edilir.

Yukarıdaki $\underline{x}_{hesap}$ vektörünün doğruluğu hata vektörü kullanılarak da yapılır:

Hata vektörü:

$$\underline{h} = \underline{x}_{hesap} - \underline{x}_{gerçek} = [2.0001 \quad -2.9998 \quad 10^{-6} \quad 0.9999 \quad -4.0002] - [2 \quad -3 \quad 0 \quad 1 \quad -4]$$

$$\underline{h} = [0.0001 \quad 0.0002 \quad 10^{-6} \quad -0.0001 \quad -0.0002]$$

Hata vektörünün uzunluğu (Öklid normu)≈0 olmalıdır:

$$\sqrt{\underline{h}\underline{h}^T} = \sqrt{0.0001^2 + 0.0002^2 + (10^{-6})^2 + 0.0001^2 + 0.0002^2} = 10^{-8} \approx 0$$

Olduğundan $\underline{x}_{hesap}$ doğru varsayılabilir. Hemen şunu da belirtelim: Hata vektörünün terimlerinin veya normunun küçük olması çözümün doğru olduğunun önemli bir bir göstergesidir, fakat yeterli bir koşul değildir. İzleyen örneklerde bu durum açıklık kazanacaktır.

Matris normları:

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$ matrisinin normları:

$\|\underline{A}\|_{\max} = \max|a_{ij}|$ (matrisin mutlak değerce en büyük elemanı)

$\|\underline{A}\|_1 = \max\|\underline{A}_j\|$ ($\|\underline{A}_j\|$ *j.* Kolonun l_1 normu)

$\|\underline{A}\|_\infty = \max\|\underline{A}_i\|$ ($\|\underline{A}_i\|$ *i.* Satırın l_1 normu)

Matris kondisyon sayısı:

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$ matrisinin kondisyon sayısı $k(\underline{A})$:

$k(\underline{A}) = \|\underline{A}\| \|\underline{A}^{-1}\|$ ile tanımlanır.

$k(\underline{A}) \approx 1$ ise $\underline{A}$ iyi kondisyonludur denir, $\underline{A}^{-1}$ hatasız hesaplanmış anlamına gelir.

Bir diğer kondisyon sayısı **Hadamard**[1] kondisyon sayısıdır:

$$k_h(\underline{A}) = \frac{|\det \underline{A}|}{V}$$

$$V = a_1 a_2 \dots a_n$$

$$a_i = \sqrt{a_{i1}^2 + a_{i2}^2 + \dots + a_{in}^2} \quad (i.\ satırın\ Öklid\ normu)$$

$0 \le k_h(\underline{A}) \le 1$ arasındadır. $k_h(\underline{A})=0$ ise det $\underline{A}$=0 matris tekildir. $k_h(\underline{A})$ değeri 1 e ne denli yakın ise matris o denli iyi kondisyonlu, 0 ra ne denli yakınsa o denli kötü kondisyonludur. $k_h(\underline{A})<0.01$ ise kötü kondisyonlu, $k_h(\underline{A})>0.1$ iyi kondisyonlu varsayılabilir. $0.01 \le k_h(\underline{A}) \le 0.1$ aralığında ise ne kötü ne de iyi kondisyonludur, durum şüphelidir.

Ters matris hesabı çok işlem gerektirdiğinden ve denklem sistemi çözümünde det $\underline{A}$ ara değer olarak zaten hesaplanabildiğinden, Hadamard kondisyon sayısını kullanmak daha pratiktir.

----
[1] Jacques Salomon **Hadamard** (1865-1963), Fransız

Örnek:

$\underline{A}=\begin{bmatrix}10 & 2\\ -1 & 9\end{bmatrix}$ Matrisinin norm ve kondisyon sayılarını belirleyelim

$\|\underline{A}\|_{max}=10$, $\|\underline{A}\|_1=\max(11,11)=11$, $\|\underline{A}\|_\infty=\max(12,10)=12$

$$\underline{A}^{-1}=\begin{bmatrix}0.097826 & -0.021739\\ 0.010870 & 0.108696\end{bmatrix}$$

$\|\underline{A}^{-1}\|_{max}=0.108696$, $\|\underline{A}^{-1}\|_1=\max(0.108696, 0.130435)=0.130435$, $\|\underline{A}^{-1}\|_\infty=\max(0.119565, 0.119540)=0.119565$

$$k(\underline{A})_{max}=\|\underline{A}\|_{max}\|\underline{A}^{-1}\|_{max}=10\cdot 0.108696=1.09$$
$$k(\underline{A})_1=\|\underline{A}\|_1\|\underline{A}^{-1}\|_1=11\cdot 0.130435=1.43$$
$$k(\underline{A})_\infty=\|\underline{A}\|_\infty\|\underline{A}^{-1}\|_\infty=12\cdot 0.119565=1.43$$

HADAMARD kondisyon sayısı:

$$\det\underline{A}=92$$
$$a_1=\sqrt{10^2+2^2}=10.20,\ a_2=\sqrt{1^2+9^2}=9.06,\ V=10.2\cdot 9.06=92.41$$
$$k_h(\underline{A})=\frac{92}{92.41}=0.9956$$

Yorumlanırsa, kondisyon sayıları 1 e yakın olduğundan A matrisi iyi kondisyonludur. Örnek olması açısından burada tüm norm ve kondisyon sayıları hesaplanmıştır. Uygulamada sadece bir norm ve kondisyon sayısı ile yetinilir.

Hasta matris(ill-conditioned matrix) ve hasta denklem sistemi

Kötü kondisyonlu matrise hasta matris(İng.: ill-conditioned) denir. Bu tür matrislerin determinantları çok küçüktür, matris tekile yakındır. Matrisin elemanlarındaki çok küçük bir değişiklik(yuvarlama hatası nedeniyle) matrisin tersinin tamamen yanlış hesaplanmasına neden olur. Dolayısıyla Hasta matris katsayılı denklem sisteminin çözümü de hatalı olacaktır.

Örnek:

$$\underline{A}\underline{x}=\underline{b}\rightarrow\begin{bmatrix}1.2969 & 0.8648\\ 0.2161 & 0.1441\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}=\begin{bmatrix}0.8642\\ 0.1440\end{bmatrix},\quad \underline{x}=?$$

Önce A nın HADAMARD kondisyon sayısını bulalım: $k_h(\underline{A})=\frac{|\det\underline{A}|}{V}$

Det A=10^{-8}

Determinant çok küçük! A nın hasta matris olduğunun ilk işaretini almış olduk.

$$a_1=\sqrt{1.2969^2+0.8648^2}=1.56,\ a_2=\sqrt{0.2161^2+0.1441^2}=0.26,\ V=a_1a_2=1.56\cdot 0.26=0.62,\ k_h(\underline{A})=\frac{10^{-8}}{0.62}=1.6\cdot 10^{-8}$$

K_h(A)=$1.6\cdot 10^{-8}\approx 0$ olduğundan A matrisi kötü kondisyonludur, hasta bir matristir. A x=b denklem sistemi de hasta bir denklem sistemidir, çözümü yaparsak hatalı sonuç alacağımıza işaret etmektedir. Gauss ile çözerek sonucu görelim:

Basit Gauss indirgemesi:

$$\begin{bmatrix}1.2969 & 0.8648\\ 0.166628 & 1.06\cdot 10^{-7}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}=\begin{bmatrix}0.8642\\ 0.82\cdot 10^{-7}\end{bmatrix}$$

$$x_1=(0.8642-0.8648\cdot 0.773585)/1.2969=0.150516$$
$$x_2=1.06\cdot 10^{-7}/0.82\cdot 10^{-7}=0.773585$$

$$\underline{x} = \begin{bmatrix} 0.150516 \\ 0.773585 \end{bmatrix}$$

Hata vektörü:

$$\underline{h} = \underline{b} - \underline{A}x = \begin{bmatrix} 0.8642 \\ 0.1440 \end{bmatrix} - \begin{bmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{bmatrix} \begin{bmatrix} 0.150516 \\ 0.773585 \end{bmatrix} = \begin{bmatrix} -5.08 \cdot 10^{-7} \\ -1.06 \cdot 10^{-7} \end{bmatrix}$$

Hata vektörünün Öklid normu :

$$\|\underline{h}\|_2 = \sqrt{\underline{h}^T \underline{h}} = \sqrt{(5.08 \cdot 10^{-7})^2 + (1.06 \cdot 10^{-7})^2} = 5.2 \cdot 10^{-7}$$

Yorum:
Hata vektörünün terimleri küçük görünüyor. Normu da öyle. O halde x vektörü doğrudur diyebilir miyiz?. HAYIR! x vektörünün tek bir rakamı dahi doğru değil! Gerçek çözüm:

$$\underline{x} = \begin{bmatrix} 2 \\ -2 \end{bmatrix}$$

dir, yerine koyarak hata var mı? Görelim:

$$\underline{h} = \underline{b} - \underline{A}x = \begin{bmatrix} 0.8642 \\ 0.1440 \end{bmatrix} - \begin{bmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{bmatrix} \begin{bmatrix} 2 \\ -2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$ sıfır hata ile sağlanıyor.

O halde hesapladığımız x=[0.150516 0.773585]^T çözümü, hata vektörü küçük olmasına rağmen, tamamen yanlıştır. Bunun nedeni: A matrisinin, dolayısıyla denklem sisteminin hasta olmasıdır.

ÖZET:

A x=b doğrusal denklem sisteminde

- A kötü kondisyonlu (hasta) ise A^{-1} (A nin tersi), büyük bir olasılıkla, hatalıdır.
- Doğrusal denklem sistemi de hastadır
- x çözümü, büyük bir olasılıkla, hatalıdır.
- |det A| <<1 hasta matrise işaret eder.

Hasta matris ile yapılan nümerik hesapların sonucuna güvenilmez. Öncelikle alınacak önlem, matrisin neden hasta olduğunu araştırmaktır. Modelleme hataları, uygun olmayan nümerik yöntem, uygun olmayan birim kullanılması ve veri hataları hasta matris oluşmasının genel nedenleri arasında sayılabilir. Matrisin kurulmasında bu hatalar yapılmamışsa, bilgisayar çözümünde mutlaka çift hassasiyet (8 Byte-DOUBLE PRECESI0N) hatta, programlama dili izin veriyorsa, dörtlü hassasiyet (16 Byte-QUAD precision**,** EXTENDED) kullanılmalıdır.