Linjära ekvationssystem

1 Inledning

Vi börjar med att se lite mer på matriser i MATLAB. Sedan ser vi på matris-vektormultiplikation och tolkar den som en linjär kombination av matrisens kolonner. Avslutningsvis ser vi på linjära ekvationssystem.

2 Matriser

Vi har redan sett lite på matriser i studioövning 2. En matris är som ni vet ett rektangulärt talschema:

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}$$

med m rader och n kolonner, vi säger att den är av typ $m \times n$.

Ett matriselement a_{ij} skrivs i MATLAB med `A(i,j)` och `[m,n]=size(A)` ger matrisens typ. Med `m=size(A,1)` ges endast antal rader och med `n=size(A,2)` ges antal kolonner.

Indexeringen i är alltid som i matrisen ovan, dvs. rad- och kolonnindex börjar alltid på ett och vi kan inte ändra på det.

En matris av typ $m \times 1$ kallade vi kolonnmatris (kolonnvektor) och en matris av typ $1 \times n$ kallades radmatris (radvektor):

$$\mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} c_1 & \cdots & c_n \end{bmatrix}$$

Element nr i ges i MATLAB av `b(i)` och antalet element ges av `m=length(b)`. Även för vektorer gäller att indexeringen alltid börjar på ett. Motsvarande gäller för radvektorn **c**.

Som exempel tar vi

$$\mathbf{A} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 0 & 2 & 4 \end{bmatrix}$$

Vi skriver in detta i MATLAB enligt

```
>> A=[1 4 7 10; 2 5 8 11; 3 6 9 12]
>> b=[1; 3; 5]
>> c=[0 2 4]
```

och ser på typerna och några element med

```
>> [m,n]=size(A)
m =
     3
n =
     4

>> A(2,3)
ans =
     8
```

Prova gärna `length` och `size` på `b` och `c`. Någon skillnad? Skriv ut något element också.

En matris kan betraktas som en uppsättning av kolonner:

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1n} \\ \vdots & & \vdots & & \vdots \\ a_{m1} & \cdots & a_{mj} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_j & \cdots & \mathbf{a}_n \end{bmatrix}$$

med kolonnerna

$$\mathbf{a}_1 = \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix}, \quad \mathbf{a}_j = \begin{bmatrix} a_{1j} \\ \vdots \\ a_{mj} \end{bmatrix}, \quad \mathbf{a}_n = \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix}$$

Man kan även betrakta den som en uppsättning av rader, men vi använder oftast kolonnrepresentationen. I MATLAB plockar man ut kolonn nr j med `A(:,j)`. Här är j kolonnindex medan radindex $i = 1, \ldots, m$ representeras av tecknet kolon `:`. På liknande vis ges rad nr i av `A(i,:)`.

```
>> a1=A(:,1)
a1 =
     1
     2
     3

>> A2=A(2,:)
A2 =
     2     5     8    11
```

Uppgift 1. Skriv in följande matriser i MATLAB.

$$\mathbf{A} = \begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 4 & 5 & 6 \\ 3 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \mathbf{c} = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 7 \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} 0 & 2 & 4 \end{bmatrix}$$

(a). Skriv ut matriselementen a_{23}, b_{23}, c_2 och d_3. Prova `size` och `length`. Ändra b_{23} genom att skriva `B(2,3)=5`.

(b). Skriv ut kolonn nr 1, 2 och 3 ur matrisen **A**. Sätt in kolonnvektorn **c** som 2:a kolonn i **A** genom att skriva `A(:,2)=c`.

(c). Radera matrisen $\mathbf{B}$ (`clear B`) och skriv in den igen genom att först bilda kolonnerna

$$\mathbf{b}_1 = \begin{bmatrix} 4 \\ 3 \\ 1 \end{bmatrix}, \quad \mathbf{b}_2 = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix}, \quad \mathbf{b}_3 = \begin{bmatrix} 6 \\ 1 \\ 1 \end{bmatrix}$$

och sedan sätta in dem i matrisen $\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \mathbf{b}_3]$.

3 Matris-vektorprodukt

Matris-vektorprodukten $\mathbf{y} = \mathbf{A}\mathbf{x}$ av en $m \times n$-matris och en n-kolonnvektor är en m-kolonnvektor som ges av

$$\underset{\mathbf{y}}{\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}} = \underset{\mathbf{A}}{\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}} \underset{\mathbf{x}}{\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}} = \underset{\mathbf{Ax}}{\begin{bmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \end{bmatrix}}$$

eller elementvis

$$y_i = \sum_{j=1}^{n} a_{ij}x_j = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n$$

Matris-vektorprodukten $\mathbf{y} = \mathbf{A}\mathbf{x}$ kan beräknas i Matlab med den inbyggda matrismultiplikationen (`*`) enligt `y=A*x` eller med lite egen programmering (som bygger upp `y` elementvis)

```
>> y=zeros(m,1);
>> for i=1:m
       s=0;
       for j=1:n
           s=s+A(i,j)*x(j);
       end
       y(i)=s;
   end
```

Ett alternativt sätt att introducera matris-vektorprodukt är att definiera $\mathbf{Ax}$ som en linjärkombination av kolonnerna i $\mathbf{A}$, (se Lay avsnitt 1.4)

$$\begin{aligned} \mathbf{y} = \mathbf{A}\mathbf{x} = \begin{bmatrix} \mathbf{a_1} & \cdots & \mathbf{a_n} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} &= x_1\mathbf{a_1} + x_2\mathbf{a_2} + \cdots + x_n\mathbf{a_n} = \\ &= \begin{bmatrix} a_{11} \\ \vdots \\ a_{m1} \end{bmatrix} x_1 + \begin{bmatrix} a_{12} \\ \vdots \\ a_{m2} \end{bmatrix} x_2 + \cdots + \begin{bmatrix} a_{1n} \\ \vdots \\ a_{mn} \end{bmatrix} x_n \end{aligned}$$

I Matlab skulle vi, för t.ex. $n = 3$, skriva

```
>> y=A(:,1)*x(1)+A(:,2)*x(2)+A(:,3)*x(3)
```

och för ett större värde på n skulle vi kunna bilda linjärkombinationen enligt

```
>> y=zeros(m,1);
>> for j=1:n
        y=y+A(:,j)*x(j);
   end
```

Uppgift 2. Skriv in följande matriser i MATLAB.

$$\mathbf{A} = \begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 4 & 5 & 6 \\ 3 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

Beräkna följande produkter, både för hand, dvs. med penna och papper, och med MATLAB, dvs. med inbyggda matrismultiplikationen (*),

$$\mathbf{Ax}, \quad \mathbf{Bx}, \quad \mathbf{AB}, \quad \mathbf{ax}, \quad \mathbf{xa}, \quad \mathbf{aB}.$$

Beräkna produkten **Ax** även genom att ni skriver en egen programkod i MATLAB. Skriv snyggt och tydligt.

4 Linjära ekvationssystem

Linjära ekvationssystem kan vi lösa med MATLAB om vi först skriver dem på matrisform. Vi tar som exempel ekvationssystemet

$$\begin{cases} x_1 + 2x_2 + 3x_3 = 14 \\ 3x_1 + 2x_2 + x_3 = 10 \\ 7x_1 + 8x_2 = 23 \end{cases}$$

som kan skrivs på matrisform

$$\begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 7 & 8 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 14 \\ 10 \\ 23 \end{bmatrix}$$

dvs.

$$\mathbf{Ax} = \mathbf{b}, \quad \text{med } \mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 7 & 8 & 0 \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \text{och } \mathbf{b} = \begin{bmatrix} 14 \\ 10 \\ 23 \end{bmatrix}$$

Vi bildar koefficientmatrisen **A** och högerledsvektorn **b** med

```
>> A=[1 2 3;3 2 1;7 8 0]
A =
      1       2       3
      3       2       1
      7       8       0

>> b=[14;10;23]
b =
    14
    10
    23
```

Med kommandot **rref** (row-reduced-echelon form) i MATLAB reduceras den utökade matrisen $[\mathbf{A}\ \mathbf{b}]$ till reducerad trappstegsform. Först bildar vi den utökade matrisen $\mathbf{E} = [\,\mathbf{A}\ \mathbf{b}\,]$ med

```
>> E=[A b]
E =
     1     2     3    14
     3     2     1    10
     7     8     0    23
```

och sedan får vi den reducerade matrisen med

```
>> R=rref(E)
R =
     1     0     0     1
     0     1     0     2
     0     0     1     3
```

Lösningen ser vi i sista kolonnen i `R`.

Om vi vet att lösningen entydigt bestämd kan vi enkelt direkt beräkna den med backslash-kommandot (\) enligt

```
>> x=A\b
x =
     1
     2
     3
```

Som ytterligare ett exempel ser vi på följande ekvationssystem med oändligt många lösningar

$$\begin{cases} x_1 + 2x_2 + 3x_3 = 10 \\ 3x_1 + 2x_2 + x_3 = 14 \\ 7x_1 + 8x_2 + 9x_3 = 46 \end{cases}$$

eller på matrisform

$$\mathbf{Ax} = \mathbf{b} \qquad \begin{bmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 7 & 8 & 9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 14 \\ 46 \end{bmatrix}$$

```
>> A=[1 2 3;3 2 1;7 8 9]
A =
     1     2     3
     3     2     1
     7     8     9

>> b=[10;14;46]
b =
    10
    14
    46
```

Vi reducerar utökande matrisen med

```
>> R=rref([A b])
R =
     1     0    -1     2
     0     1     2     4
     0     0     0     0
```

Vi har en fri variabel. Om vi sätter $x_3 = t$ får vi

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2+t \\ 4-2t \\ t \end{bmatrix}$$

där t är ett godtyckligt reellt tal.

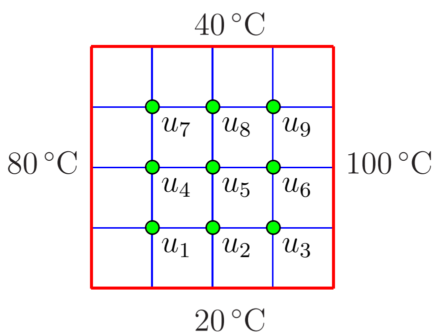
Uppgift 3. Skriv följande ekvationssystem på matrisform och lös dem sedan med `rref` respektive backslash (\).

$$\begin{cases} x_1 + 5x_2 + 9x_3 &= 29 \\ 2x_1 + 5x_3 &= 26 \\ 3x_1 + 7x_2 + 11x_3 &= 39 \end{cases} \qquad \begin{cases} x_1 + x_2 + 3x_3 + 4x_4 &= 2 \\ -2x_1 + 2x_2 + 2x_3 &= -4 \\ x_1 + x_2 + 2x_3 + 3x_4 &= 1 \\ x_1 - x_2 - 2x_3 - x_4 &= 1 \end{cases}$$

Har systemen entydig lösning? Hur läser du av resultatet från `rref` för att svara på den frågan. Skriv upp samtliga lösningar till systemen på papper.

Uppgift 4. Läs exemplet ”Balancing Chemical Equations” i Lay avsnitt 1.6. Lös sedan stökiometriuppgiften Lay 1.6: 10. Ge kommandot `format rat` så att MATLAB skriver svar med rationella tal så blir det något enklare att tolka resultatet. Med `format short` får vi sedan tillbaka standard formatet.

Uppgift 5. Vi skall beräkna temperaturen på en stålplatta där plattans kanter hålls vid temperaturer enligt figuren. Detta är en fortsättning på uppgiften Lay 1.1: 33.



Antag att temperaturen i en nodpunkt är medelvärdet av temperaturena i de närmsta nodpunkterna i *väster*, *öster*, *söder* och *norr*. Låt $u_1, u_2, \cdots, u_9$ beteckna temperaturerna i de olika nodpunkterna. Sätt upp de ekvationer som ger temperaturen i de olika nodpunkterna. Skriv det linjära ekvationssystemet på matrisform $\mathbf{Au} = \mathbf{b}$ och lös med backslash-kommandot (\) i MATLAB.