Mátrixok, vektorok

A Matlab alapvető változói a mátrixok.
Mátrix = egy kétdimenziós tömb:

$$\underbrace{A = \begin{bmatrix} * & * & \cdots & * & * \\ * & * & \cdots & * & * \\ \vdots & & & & \\ * & * & \cdots & * & * \end{bmatrix}}_{m \text{ oszlop}} \left.\vphantom{\begin{bmatrix} * \\ * \\ \vdots \\ * \end{bmatrix}}\right\} n \text{ sor}$$

Ekkor A egy $n \times m$-es tömb (mátrix)

Sorvektorok

Sorvektor: olyan mátrix, melynek egyetlen sora van:

$$a = \begin{bmatrix} * & * & \cdots & * & * \end{bmatrix}$$

Sorvektorok létrehozása Matlab-ban

szögletes zárójelek között elemeinek felsorolásával, pl:

- az elemeket vesszővel választjuk el:

  `a =[-1.2, 3.1, 4.7, 1.9]`

- vagy az elemeket szóközzel választjuk el:

  `a=[-1.2 3.1 4.7 1.9]`

A vektor elemeinek számozása 1-gyel kezdődik, `a(i)` az *a* vektor *i*-edik eleme.

A : (kettőspont) operátor

Ha a vektor elemei szabályos lépésközzel követik egymást, akkor használhatjuk a kettőspont operátort:

- a $b = [1, 2, 3, 4, 5]$ vektor:

  ```
  b = 1:5
  ```

- a $c = [5, 4, 3, 2, 1]$ vektor:

  ```
  c = 5:-1:1
  ```

- a $d = [2, 2.2, 2.4, 2.6, 2.8, 3]$ vektor

  ```
  d=2:0.2:3
  ```

Általában:

```
x=elsoelem:lepeskoz:utolsoelem
```

ahol a lépésköz negatív is lehet, vagy

```
x=elsoelem:utolsoelem
```

ekkor a lépésköz 1.

A linspace függvény

Ha megadott számú, egyenlő lépésközű elemet szeretnénk megadni:

```
x=linspace(elsoelem,utolsoelem,elemekszama)
```

Pl. az

```
e=linspace(1,2,6)
```

utasítás egy 6 elemű vektort ad meg, melynek első eleme 1, utolsó eleme 2, és az elemek egyforma lépésközzel követik egymást:

$$e = [1, 1.2, 1.4, 1.6, 1.8, 2]$$

Ha a linspace függvényt csak két argumentummal hívjuk, akkor 100 elemű vektort kapunk:

```
x=linspace(elsoelem,utolsoelem)
```

Oszlopvektorok

Oszlopvektor: olyan mátrix, melynek egyetlen oszlopa van:

$$a = \begin{bmatrix} * \\ * \\ \vdots \\ * \\ * \end{bmatrix}$$

Oszlopvektorok létrehozása Matlab-ban

- szögletes zárójelek között elemeinek felsorolásával, az elemeket pontosvesszővel választjuk el:
  `a =[-1; 3; 0; 1]`
- vagy egy sorvektort transzponálunk:
  `a=[-1, 3, 0, 1]; b=a'`

A vektor elemeinek számozása 1-gyel kezdődik, `a(i)` az a vektor i-edik eleme.

Transzponálás

Sorból oszlopot, oszlopból sort csinál:

- $$a = [-1, 3, 0, 1] \quad \Longrightarrow \quad a' = \begin{bmatrix} -1 \\ 3 \\ 0 \\ 1 \end{bmatrix}$$
- $$a = \begin{bmatrix} -1 \\ 3 \\ 0 \\ 1 \end{bmatrix} \quad \Longrightarrow \quad a' = [-1, 3, 0, 1]$$

(**valójában a ' jel konjugált transzponáltat eredményez, a konjugálás nélküli transzponátás:** `a.'` **vagy** `transpose(a)`**. Ez csak akkor jelent különbséget, ha a vektor elemei nem valós számok.**)

Példák

`a=[1;2;3]`
vagy
`a=[1, 2, 3]'`
vagy
`a=(1:3)'`
	$a = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

`a=(1:0.2:2)'`
vagy
`a=(linspace(1,2,6))'`
	$a = \begin{bmatrix} 1 \\ 1.2 \\ 1.4 \\ 1.6 \\ 1.8 \\ 2 \end{bmatrix}$

Néhány hasznos függvény

- size
  Megadja egy mátrix sorainak és oszlopainak számát.
  Pl. $a = [1, -2, 0, 5]$ esetén `size(a)` értéke: `[1,4]`
  $a = [1; -2; 0; 5]$ esetén `size(a)` értéke: `[4,1]`
- numel
  Megadja a mátrix elemeinek számát.
  Pl. az előző két a vektorral `numel(a)` értéke 4.
- ones
  `ones(n,m)` egy $n \times m$-es, csupa 1-esből álló mátrix.
  Pl. `ones(1,5)` egy 5 elemű, csupa 1-es sorvektor.
  `ones(5,1)` egy 5 elemű, csupa 1-es oszlopvektor.
- zeros
  `zeros(n,m)` egy $n \times m$-es, csupa 0-ból álló mátrix.
  Pl. `zeros(1,5)` egy 5 elemű, csupa 0 sorvektor.
  `zeros(5,1)` egy 5 elemű, csupa 0 oszlopvektor.

Vektorok darabolása

Sor- és oszlopvektorokra is:

`v(2)`	a v vektor 2. eleme
`v([2,4])`	a v vektor 2. és 4. eleméből álló vektor
`v(2:4)`	a v vektor 2., 3. és 4. eleméből álló vektor
`v(end)`	a v utolsó eleme
`v(2)=[]`	elhagyja a vektor 2. elemét
`v([2,4])=[]`	elhagyja a vektor 2. és 4. elemét
`v(2:4)=[]`	elhagyja a vektor 2., 3. és 4. elemét

A fenti utasítások eredménye aszerint lesz sor-, vagy oszlopvektor, hogy a v sor-, vagy oszlopvektor volt-e.

Emlékeztető: a vektor elemeinek számozása 1-gyel kezdődik.

Vektorelemek módosítása

Az előzőeket felhasználva módosíthatjuk egy vektor elemeit. Pl.: ha

$$v = [-1, 4, 6, 0, -3, 5]$$

akkor (pirossal jelölve a módosított elemeket)

- `v(2)=-5` $\Longrightarrow$ $v = [-1, -5, 6, 0, -3, 5]$
- `v([2,4])=[-5,1]` $\Longrightarrow$ $v = [-1, -5, 6, 1, -3, 5]$
- `v([2,4])=1` $\Longrightarrow$ $v = [-1, 1, 6, 1, -3, 5]$
- `v(2:4)=[-5,-2,1]` $\Longrightarrow$ $v = [-1, -5, -2, 1, -3, 5]$
- `v(2:4)=1` $\Longrightarrow$ $v = [-1, 1, 1, 1, -3, 5]$

Ha a vektor egy részét módosítani akarjuk, akkor egy ugyanolyan méretű vektorral kell egyenlővé tennünk, vagy egyetlen számmal. Utóbbi esetben minden módosítandó elemet arra a számra cserél.

Fontos! A `v(9)=4` utasítás eredménye az `v=[-1,4,6,0,-3,5,0,0,4]` vektor (a legkisebb olyan vektor, amelyben van értelme a 9. elemre hivatkozásnak, a nemdefiniált elemeket 0-kal tölti fel. Megváltozik a vektor mérete, erre nem figyelmeztet!)

Vektorok összefűzése

Emléleztető: szögletes zárójelben az elemeket

- vesszővel vagy szóközzel választjuk el $\Longrightarrow$ sorvektor
- pontosvesszővel választjuk el $\Longrightarrow$ oszlopvektor

Ugyanígy építhetünk össze vektorokat is, ha a méretük engedi (az eredménynek ,,téglalap alakúnak" kell lenni).

Hüvelykujj-szabály: ha a szögletes zárójelen belül az utolsó elem után

- vesszővel, vagy szóközzel elválasztva írjuk az új elemet, akkor azt az utolsó elem után helyezi el (ha lehetséges)
- pontosvesszővel elválasztva írjuk az új elemet, akkor sort tör, az új elemet az eddigiek alá helyezi (ha lehetséges)

Vektorok összefűzése

`[a b]` vagy `[a,b]`	két sorvektor egymás után fűzése
`[-4 a 3 -1]`	sorvektor bővítése újabb elemekkel

Példa: Ha

$$\mathtt{a} = \begin{bmatrix} 5 & -1 & 2 \end{bmatrix}, \mathtt{b} = \begin{bmatrix} 2 & -7 & 3 & -1 \end{bmatrix},$$

akkor

$$\mathtt{[a\ \ b]} = \begin{bmatrix} 5 & -1 & 2 & 2 & -7 & 3 & -1 \end{bmatrix}$$

$$\mathtt{[-4\ \ a\ \ 3\ \ -1]} = \begin{bmatrix} -4 & 5 & -1 & 2 & 3 & -1 \end{bmatrix}$$

Amit nem lehet: Az `a` és `b` vektorokat nem helyezhetem egymás alá, mert nem egyforma hosszúak.

Ha egyforma hosszúak lennének, akkor egymás alá rakhatnánk őket $\Longrightarrow$ egy 2 sorból álló mátrixot kapnánk.

Vektorok összefűzése

`[m;n]`	két oszlopvektor egymás után fűzése
`[1;m;-3]`	oszlopvektor bővítése újabb elemekkel

Példa: Ha

$$\mathtt{m} = \begin{bmatrix} -2 \\ 1 \\ 6 \end{bmatrix}, \mathtt{n} = \begin{bmatrix} 0 \\ 4 \end{bmatrix},$$

akkor

$$\mathtt{[m;n]} = \begin{bmatrix} -2 \\ 1 \\ 6 \\ 0 \\ 4 \end{bmatrix}, \quad \mathtt{[1;m;-3]} = \begin{bmatrix} 1 \\ -2 \\ 1 \\ 6 \\ -3 \end{bmatrix}$$

Amit nem lehet: Az `m` és `n` vektorokat nem tehetjük egymás mellé, mert nem ugyanolyan hosszúak.

Ha egyforma hosszúak lennének, akkor egymás mellé rakhatnánk őket $\Longrightarrow$ egy 2 oszlopból álló mátrixot kapnánk.

Aritmetikai műveletek vektorokkal

Ha `a` és `b` két ugyanolyan méretű vektor, akkor

- `a+b` ill. `a-b` a két vektor elemenkénti összege, ill. különbsége
- `a+1` az `a` minden eleméhez hozzáad 1-et
- `a.^2` az `a` minden elemét négyzetre emeli
- `a.*b` az `a` és `b` vektorok elemenkénti szorzata,
- `a./b` az `a` és `b` vektorok elemenkénti hányadosa
- `1./a` az `a` elemenkénti reciproka

Az utolsó négy esetben a műveleti jel előtti pont a művelet elemenkénti végrehajtását eredményezi. A pont nélküli műveletek a hagyományos lineáris algebrai műveleteket jelentik.

`sin`, `cos`, `tan`, `exp`, `log`, `sqrt`, `abs`, stb. mind elemenként hajtódik végre.

`NaN` : Not a Number (pl. 0/0, `Inf/Inf`)

Példák

$$\mathtt{a} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}, \mathtt{b} = \begin{bmatrix} -2 & 4 & 1 \end{bmatrix}$$

- $\mathtt{a+b} = \begin{bmatrix} -1 & 6 & 4 \end{bmatrix}$
- $\mathtt{a+1} = \begin{bmatrix} 2 & 3 & 4 \end{bmatrix}$
- $\mathtt{a.\hat{}\,2} = \begin{bmatrix} 1 & 4 & 9 \end{bmatrix}$
- $\mathtt{a.*b} = \begin{bmatrix} -2 & 8 & 3 \end{bmatrix}$
- $\mathtt{a./b} = \begin{bmatrix} -0.5 & 0.5 & 3 \end{bmatrix}$
- $\mathtt{1./a} = \begin{bmatrix} 1 & 0.5 & 0.3333 \end{bmatrix}$

Ha a pontot nem szerepel a $*$ előtt, akkor:

```
>> a*b
```

Error using * Incorrect dimensions for matrix multiplication. Check that the number of columns in the first matrix matches the number of rows in the second matrix. To perform elementwise multiplication, use '.*'.

Néhány hasznos függvény

- `min(x)` és `max(x)` az x vektor legkisebb és legnagyobb eleme
- `sort(x)` az x elemeit növekvő sorrendbe rendezi
- `sort(x,'descend')` az x elemeit csökkenő sorrendbe rendezi
- `flip(x)` az x elemeit fordított sorrendben sorolja fel
- `length(x)` az x vektor sor és oszlop száma közül a nagyobb
- `numel(x)` az x elemeinek száma
- `sum(x)` az x vektor elemeinek összege
- `prod(x)` az x vektor elemeinek szorzata
- `mean(x)` az x vektor elemeinek átlaga
- `x(3)` az x vektor harmadik eleme
- `x(1:3)` az x vektor első három eleme
- `x(3:end)` az x vektor minden elemei a harmadiktól az utolsóig

1. feladat

(a) Az elemek egyenkénti begépelése nélkül állítsa elő az alábbi vektorokat!

(1) $a = (0, 1, \cdots, 30)$
(2) $b = (2, 4, 6, \ldots, 100)$,
(3) $c = (2, 1.9, 1.8, \cdots, 0.1, 0)$
(4) $d = (0, 3, 6, \ldots, 27, 30, -100, 30, 27, \cdots, 6, 3, 0)$
(5) $e = \left(\frac{1}{2}, \frac{1}{3}, \cdots, \frac{1}{20}\right)$
(6) $f = \left(\frac{1}{2}, \frac{2}{3}, \cdots, \frac{19}{20}\right)$

(b) Legyen x egy adott 100 elemű sorvektor. Az x vektorból állítsa elő azt az y vektort, melynek elemei

(1) az x vektor elemei fordított sorrendben felsorolva,
(2) az x vektor első 5 eleme,
(3) az x vektor elemei ugyanolyan sorrendben, kivéve az x 4. elemét
(4) az x vektor elemei ugyanolyan sorrendben, kivéve az x 5., 72. és 93. elemét
(5) az x vektor páratlan sorszámú elemei
(6) az x vektor 2., 5., 17. és 81. eleme.

2. feladat

Legyen x egy adott sorvektor. A `for` utasítás használata nélkül az x vektorból állítsa elő azt az y vektort, melynek i-edik eleme

(1) $x(i) + 2$

(2) $x(i)^2$

(3) $1/x(i)$

(4) $\sin(x(i)^3 - 1)$

(5) $x(i) - i$