

M0031M, Linjär algebra och differentialekvationer

Föreläsning 6

Ove Edlund

LTU

2016-09-06

Några elementära funktioner i Matlab

Exempel

- **exp** – Beräknar e^x , **exp(3.2)** = $e^{3.2}$
- **log** – Naturliga logaritmen \ln , **log(1.0)** = $\ln 1.0$
- **sin, cos, tan** – ex, **sin(2.3)** = $\sin 2.3$
- **asin, acos, atan** – ex, **atan(1.0)** = $\arctan 1.0 = \tan^{-1} 1.0$
- **sqrt** – kvadratroten, **sqrt(2.0)** = $\sqrt{2}$
- **abs** – Absolutbeloppet, **abs(-3.1)** = $|-3.1|$

Alla dessa går att applicera på vektorer och matriser. Beräkningen utförs då **element för element**. T.ex.

```
>> abs([-1.2 2.4 -2.2])  
ans =  
1.2000 2.4000 2.2000
```

Ytterligare elementära funktioner i Matlab

Logaritmer och rötter

- **log** – Naturlig logaritm \ln , **log(12)** = $\ln 2$
- **log10** – Bas 10 logaritm \lg , **log10(2)** = $\lg 2$
- **log2** – Bas 2 logaritm \log_2 , **log2(2)** = $\log_2 2$
- **sqrt** – kvadratroten, **sqrt(2.0)** = $\sqrt{2}$
- **nthroot** – n -te roten, **nthroot(8,3)** = $\sqrt[3]{8}$

Komplexa tal

- **i** – Imaginära enheten, $z = 4 + 2*i$
- **real** – Realdelen, **real(z)**
- **imag** – Imaginärdelen, **imag(z)**
- **abs** – Absolutbeloppet, **abs(z)**
- **angle** – Argumentet, **angle(z)**

Plus, gånger osv.

Element för element

x=a:b:c Vektorn **x** innehåller element från **a** till **c** i steg om **b**. Ex **0:0.5:2**
= [0 0.5 1.0 1.5 2]

+ , **-** Adderar/subtraherar matriser och vektorer element för element

.* , **./** Multiplicerar/dividerar matriser och vektorer element för element

.^ Upphöjer matriser och vektorer element för element

Exempel

```
>> 2.^[1 2 1 3]
```

```
ans =
```

```
2 4 2 8
```

```
>> [1 2 1 3].^2
```

```
ans =
```

```
1 4 1 9
```

Operationer på vektorer

Vektorer

plot(x,y) Ritar en kurva med koordinaterna hämtade från **x** och **y**

dot(x,y) Skalärprodukten. (Samma som **x'*y**)

cross(x,y) Vektorprodukten för vektorer i \mathbb{R}^3

norm(x) Längden på vektorn, dvs $\|x\|$

length(x) Dimensionen, dvs antal element i **x**.

Mer information fås med **help kommando**

Plus, gånger osv.

Matrisräkning

- + , - Matrisaddition/subtraktion (samma som tidigare!!!)
- * Matrimultiplikation
- / Löser system. \mathbf{B}/\mathbf{A} motsvarar $B A^{-1}$.
- \ Löser system. $\mathbf{A}\backslash\mathbf{B}$ motsvarar $A^{-1}B$.
- ^ Matris-upphöjt-i. \mathbf{A}^n motsvarar A^n , där n är ett heltal.
- ' Transponerar matriser (& vektorer). $\mathbf{A}' = A^T$

Linjära ekvationssystem $Ax = b$ löses av $x = A^{-1}b$. I Matlab skriver vi $\mathbf{x}=\mathbf{A}\backslash\mathbf{b}$

Manipulera matriser

Några exempel

- $\mathbf{A}(3,2)$ avser elementet på rad 3 och kolonn 2 i \mathbf{A}
- $\mathbf{A}(:,4)$ avser den kolonnvektor som ges av kolonn 4 i \mathbf{A}
- $\mathbf{A}(2,:)$ avser den radvektor som ges av rad 2 i \mathbf{A}

- $\mathbf{A}(3,2)=4.0$ ändrar elementet på rad 3 och kolonn 2 till att vara 4.0
- $\mathbf{A}(:,4)=2*\mathbf{A}(:,4)$ dubblar alla element i kolonn 4

Operationer på matriser

Matriser

randn(m,n) Skapar en $m \times n$ -matris med normalfördelade slumpstal

eye(n) Skapar en enhetsmatris med dimension $n \times n$

eig(A) Egenvärdena hos **A**

inv(A) Inversen, dvs A^{-1} .

Undvik, om det går att använda \ eller / !!!!

rref(A) Ta matrisen till reducerad trappstegsform (reduced row echelon form)

det(A) Determinanten

[m,n]=size(A) ger **m** antal rader och **n** antal kolonner i **A**

Mer information fås med **help kommando**

Matlabscript

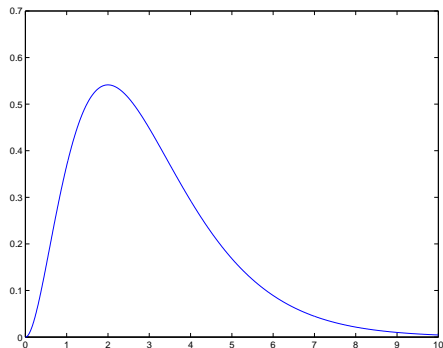
Kommandosekvenser som man vill spara och köra vid senare tillfällen lägger man i en scriptfil i Matlab. Exempel:

```
exscript.m
```

```
t = 0:0.01:10;  
y = t.^2.*exp(-t);  
plot(t,y)
```

Användning:

```
>> exscript
```



Summor och produkter

- Beräkna $\sum_{k=0}^{31} (k^2 + 1) \sin\left(\frac{\pi k}{8}\right)$
- Beräkna $\prod_{k=1}^4 \frac{10 - k + 1}{k} = \frac{10 \cdot 9 \cdot 8 \cdot 7}{1 \cdot 2 \cdot 3 \cdot 4} = \binom{10}{4}$ dvs 10 över 4

I Matlab använde vi

... elementvisa operationer på vektorer för att beräkna termer/faktorer, och sedan

- **sum** – Summera elementen i en vektor.
- **prod** – Beräkna produkten av elementen i en vektor
- **(nchoosek** – n över k , binomialkoefficienten)

Skapa sida i rapporten

Givet ett Matlabscript, tex `exscript.m` kan man skapa en pdf-fil som innehåller själva scriptet och en körning av det, med

```
publish('exscript','pdf')
```

Inlämningsuppgifterna ska lösas med script och redovisas med filerna som fås av `publish`.

Polynom i Matlab

Polynom, som t.ex.

$$2x^2 - 3x + 1$$

representeras i Matlab av en vektor med polynomkoefficienterna:

$$\gg p = [2 -3 1]$$

Polynomfunktioner

polyval(p, xs) Evaluerar polynomet med koefficienter **p** i alla punkter i vektorn **xs**.

roots(p) Beräknar alla nollställen till polynomet med koefficienter **p**.

p = polyfit(x, y, n) Finner det polynom av gradtal **n** som bäst överensstämmer med datapunkterna i **x**, **y**.

Egna funktioner

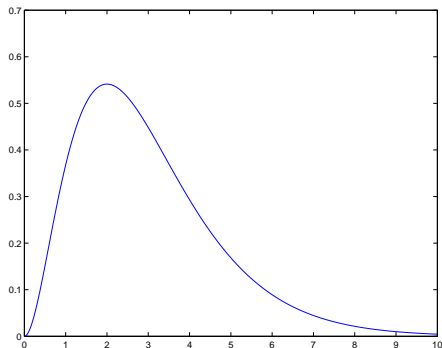
Exempel på en funktiondefinition

```
exfun.m
```

```
function y = exfun(x)  
y = x.^2.*exp(-x);
```

Användning:

```
>> t = 0:0.01:10;  
>> plot(t, exfun(t))
```



Anonyma funktioner

Alternativt kan funktionshandtag användas för att skapa anonyma funktioner

```
>> exfun2 = @(x)x.^2.*exp(-x);  
>> t = 0:0.01:10;  
>> plot(t, exfun2(t))
```

