# Grundlagen

Clc

Clear

Close

Sin

Cos

Sqrt

Log

Exp

Nthroot(-27,3) = 3

Inf % unendlich

Eps %interne Rechengenauigkeit 浮点相对精度

Nan %not a number

Linspace(start, stop, n) %创建线性间距向量 / 只能创建行向量

y1 = linspace(-5,5,7)

y1 = *1×7*

-5.0000 -3.3333 -1.6667 0 1.6667 3.3333 5.0000

X = 1 : 4;

X‘ %将行向量X转置，变为列向量

Linspace(start,stop,n)‘ %创建列向量

Transpose(X) = X’ %将X矩阵转置

Zeros(zeile, spalte) %创建m行，n列的零矩阵

Ones(zeile,spalte) %创建m行，n列的1矩阵

Eye(m) %创建m行m列的单位矩阵

Rand(m) %创建m行m列的随机数矩阵

A=ones(2,3);

B=zeros(1,3);

C=[A;B]

%创建复合矩阵

A=rand(10,6);

[r,c]=size(A) %size 确定A矩阵的行数和列数

r=10 c=6

size(A,2)

ans = 6

A =rand(10,6)

Length(A) %确定A矩阵的最大行数/列数（也可以说确定行向量/列向量的长度）

=10

B=rand(10,11);

Length(B)

=11

A = ‘MATLAB’ %创建字符串‘MATLAB’

A(3) = T

B=‘Hello’；

C = [A, B]

C = MATLABHello

## Index

A= rand(2,3)

A(2,3) %显示A矩阵第2行第3列的数

= XXX

A(end,1)

= XXX %显示A矩阵最后一行第一列的数

A=magic(3) %创建一个三行三列的魔方矩阵

A([2,3],2) %找到第2，3行和第2列的数

A([1,end],[1,end]) %找到第一行和最后一行，第一列和最后一列的数

A(:,[2,3]) %找到A矩阵中所有行，第2和3列的数

A(end,:) %找到A矩阵中最后一行的数

A(4,2) = 7; %将数字7赋值到A矩阵的第4行和第2列

A(2,:) = [2 3 4] %将[2 3 4]赋值到A矩阵的第二行

z.B.

>> max([10 20 30])

ans =

30

>> [m,n]=max([10,20,30])

m =

30

n =

3

>> [~,n]=max([10 20 30])

n =

3

## Operatoren

+ - \* / ^ .\* ./ .^

\*： 表示矩阵之间的相乘

.\* : 表示矩阵元素之间的相乘

x + 2y = 13

3x + 4y = 17

• Überführen in Matrixform: A \* x = b

• MATLAB berechnen lassen: x = A\b

%解线性方程组

## Statistische Operationen auf Matrizen

Max

Min

Mean %求中值

Median %找中位数

Mode %数组中出现最多次数的值

Std %Standardabweichung求标准差

Var %Varianz求方差

Sum %求各元素的和

Prod(A) %求A矩阵中，每列元素的乘积

Sort(A) %将矩阵A每列元素按顺序排列

Sortrows(A,1) %将矩阵A中的第一列按顺序排列

## Logik

True : 1

False : 0

== > < >= <= ~=(ungleich)

>> r = rand(3,4)

r =

0.8147 0.9134 0.2785 0.9649

0.9058 0.6324 0.5469 0.1576

0.1270 0.0975 0.9575 0.9706

>> tf = (r < 0.5)

tf =

0 0 1 0

0 0 0 0

1 1 0 0

>> r = r .\* tf

r =

1. 0 0.2785 0

0 0 0 0.1576

0.1270 0.0975 0 0

NaN： not a number

Inf: unendlich

Isnan(x) % 检查x中的不是数的元素

Strcmp(a,b) %比较字符串a,b，区分大小写 若完全相同，则输出1，否则输出0

Sctcmpi(a,b) %比较字符串a,b, 不区分大小写 若相同，则输出1，否则输出0

Strncmp(a,b,n) %比较字符串a,b的前n个字符，区分大小写，若相同出1，否则出0

Strncmpi(a,b,n) %比较字符串a,b的前n个字符，不区分大小写，若相同出1，否则出0

& und

| oder

~ not

&& % expr1 && expr2 表示使用短路行为的逻辑 AND 运算。即，如果 expr1 为逻辑值 0 (false)，将不计算 expr2 的结果。每个表达式的计算结果都必须为标量逻辑值。

|| % expr1 || expr2 表示使用短路行为的逻辑 OR 运算。即，如果 expr1 为逻辑值 1 (true)，将不计算 expr2 的结果。每个表达式的计算结果都必须为标量逻辑值。

Any all %确定数组中的单一元素和所有元素的逻辑关系

Find(x>10) %找到x数组中大于10的所有元素

z.B.

>> x = [4,5,2,0,1,4]

x =

4 5 2 0 1 4

>> y1=any(x==3)

y1 =

logical

0

>> y2=any(x==4)

y2 =

logical

1

>> y3=all(x>-1)

y3 =

logical

1

## Cell数组{}

z.B

>> C = {1,[],3;'foo',rand(2,4),{1,2}}

C =

[ 1] [] [ 3]

'foo' [2x4 double] {2x1 cell}

## Structure

## Table

Datenum: 将日期和时间转换为日期序列值，例如一天为86400秒（0-86399）

Datestr: 将日期和时间转换为字符串格式

## Erstellen von Grafiken建立图像

Plot(x,y) color: k-black marker: o x + \* s d . linestyle: -- : -. –

Xlabel(‘XXX’)

Ylabel(‘XXX’)

Title(‘XXX’)

Legend(‘XXX’)

Mehr Grafiken

Figure(1)

Plot(x,y)

...

Subplot(x,y,1)

Subplot(x,y,2)

Subplot(x,y,3)

…

## Aufbau einer Grafik

Figure – Axes – Plot objects

## Laden & Speichern

Load() %load m文件

Xlsread() %loadExcel文件

Readtable() %load csv文件的Table文本

Csvread() %load csv文件的数字文本

Save() %存储为m文件

Xlswrite() %存储为Excel文件

Writetable() %存储为Table文件

Csvwrite() %存储为CSV文件

Fprintf %存储在Text文件中

## Eingabe / Ausgabe

**Eingabe**

Input %请求用户输入。。。

**Ausgabe**

Disp(‘…’) %以Text或Arrays的形式显示

Fprintf(‘Eis:%d$\n’,2) % 输出 Eis = 2 $ Bildschirm的形式

Warning(‘…’)

Error(‘…’) %输出警告/错误

**If-else**

Z.B.

>> value = 95;

>> if (value < 40)

cor = 7;

elseif (value < 60)

cor = 4;

elseif (value < 90)

cor = 1;

else cor = 0

end

cor =

0

**Switch-case**

Z.B.

>> x=4;

>> switch x

case 1

disp('x is 1')

case 2

disp('x is 2')

otherwise

disp('x is 3')

end

x is 3

**if – else vs. switch – case**

switch str

case {'a','b'}

disp('It is a or b');

case 'c' disp('it is c');

otherwise disp('not valid');

end

%%

if strcmp(str,'a') || strcmp(str,'b')

disp('It is a or b');

elseif strcmp(str,'c') disp('it is c');

else disp('not valid');

end

**for – loop**

v = [1 13 2 5];

for idx = v

disp(2\*idx)

end

>> Untitled2

2

26

4

10

**Infinite – loop**

x = 3;

while (x>2)

x = 2\*x;

end

# Simulink

初值问题，不要忘记设置。以及龙格库塔方法，fixed-step等

# Signalverarbeitung

#### 内插法（interpolation）

Interp1(x, y, x\_interpolation, methode) % 一维插值，x, y, x的新定义域，方法（‘linear,pchip oder spline’）

Interp2, interp3, … , intern % 多维等距点插值

Griddata(x, y, x\_interpolation, y\_interpolation, methode) % 多维散点插值

X\_i = linspace(x\_min, x\_max) % 建立等距网格点

Spline 对于振动 oszillierenden verlauf， 优先选择

#### 外插法（Extrapolation）

Variable = interp1(x, y, v\_i, ‘linear’, ‘extrap’) % 外插法插值

#### 逼近法（Approximation）

**Koeffizienten = polyfit(x, y, Ordnung) % Polynomial curve fitting**

**Polyval(Koeffizienten, x\_approximation) % Polynomial evaluation**

#### 调节采样（Anpassung der Abtastrate）

Downsample() % 降低采样率

Upsample() %提高采样率

Decimate() %抽取 - 按整数因子降低采样率

Interp() % 插值 - 通过整数因子提高采样率

Resample() %将统一或非均匀数据重新采样为新的固定速率

## Analyse im Zeitbereich(时域分析)

#### Korrelation(互相关)

Xcorr(x,y) %

#### Autokorrelation

Xcorr(x,’scaling’)

#### Integration von Messdaten

Sum

Cumsum

Trapz

Cumtrapz

#### Differentiation von Messdaten

Diff(Messwerte, Ordnung)

## Analyse im Frequenzbereich (频域分析)

Fft % 快速傅里叶变换

#### Erstellung eines Fensters

Fenstervektor = hann(L) % hanning 窗口

Hamming / blackman / kaiser

z.B.

Signal\_hann = Signal.\*hann(length(Signal))‘ ;

Visualisierung eines Fenster 可视化窗口

Wvtool(Fenstervektor)