

4. Roots of algebraic and transcendental equations

代数方程式と超越方程式の根

- 代数方程式（多項式）と超越方程式の根
- ユーザー定義の関数
- 超越方程式の根
- Symbolic computation 文字式処理

Roots of polynomial equations: roots 多項式の根(roots)

- 2次方程式 $x^2-x-2=(x-2)(x+1)=0$ の解を求めるには次のように入力する

```
>> C=[1,-1,-2];  
>> roots(C)  
ans =  
     2  
    -1
```

$f(x)=0$ の解 x を求める

$f(x)$ が $a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ の場合

```
C=[an, an-1, ..., a1, a0];
```

```
roots(C)
```

- 3次方程式 $x^3+1=0$ の解を求める場合は次のようになる

```
>> C=[1,0,0,1];  
>> roots(C)  
ans =  
-1.00000 + 0.00000i  
 0.50000 + 0.86603i  
 0.50000 - 0.86603i
```

Next: $f(x)$ がもっと複雑な場合には？

User-defined functions

- 任意の関数を自分で定義できる。 スクリプトファイルに以下のように書く

```
function [y1,...,yN] = myfun(x1,...,xM)
y1 = ...
...
end
```

- スクリプトファイルmyfun.mを保存

```
#myfun.m
function y = myfun(x)
y = x^2+sin(x)-1;
end
```

functionのファイルを保存
function名前とファイル名
は同じにする

- 実行は次のようにタイプする： 関数名(変数の値)

```
>> myfun(0)
ans = -1
>> myfun(1)
ans = 0.84147
```

Functionファイルの保存場所に注意
実行するディレクトリと同じ場所に
保存

User-defined function

関数の値

関数の名前

変数

```
function y=myfun(x)
```

```
y=x^2+sin(x)-1;
```

```
end
```

myfun(0) とコマンドを入力すると

x=0を代入したyの値

0^2+sin(0)-1 の値が得られる

Matlabの場合は別のスクリプトファイルを用意しなくても
スクリプトの一番下に記述する方法もある

実行内容

```
myfun(0)  
myfun(0)
```

関数

```
function y=myfun(x)  
y=x^2+sin(x)-1;  
end
```

Anonymous function

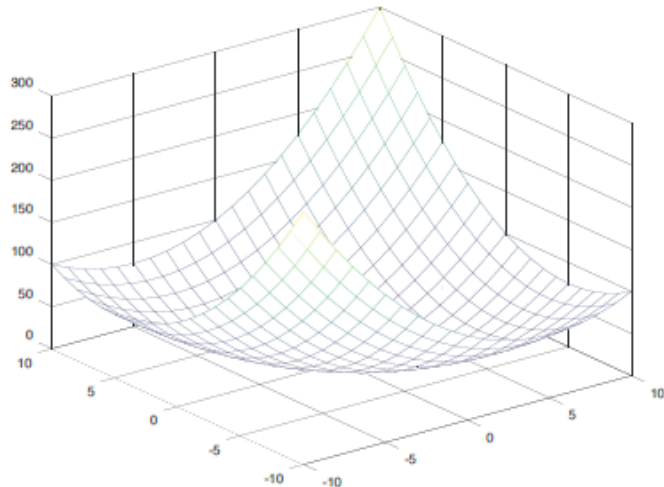
- Anonymous function: user function以外の関数の作成方法（スクリプトファイルなしで書く）

```
>> myfun1 = @(x) (x^2+sin(x)-1);  
>> myfun1(1)  
ans = 0.84147
```

関数名=@(変数) (関数の中身);

- 2変数以上の関数の定義

```
>> myfun2 = @(x,y) (x.^2+y.^2+x.*y);  
>> [X,Y] = meshgrid(-10:10);  
>> mesh(X,Y,myfun2(X,Y))
```



X^2ではなくx.^2と書いているのはxが行列やベクトルの場合でも取り扱えるように定義するため

超越方程式の根の計算：fsolve

$f(x)=0$ の解 x を探す

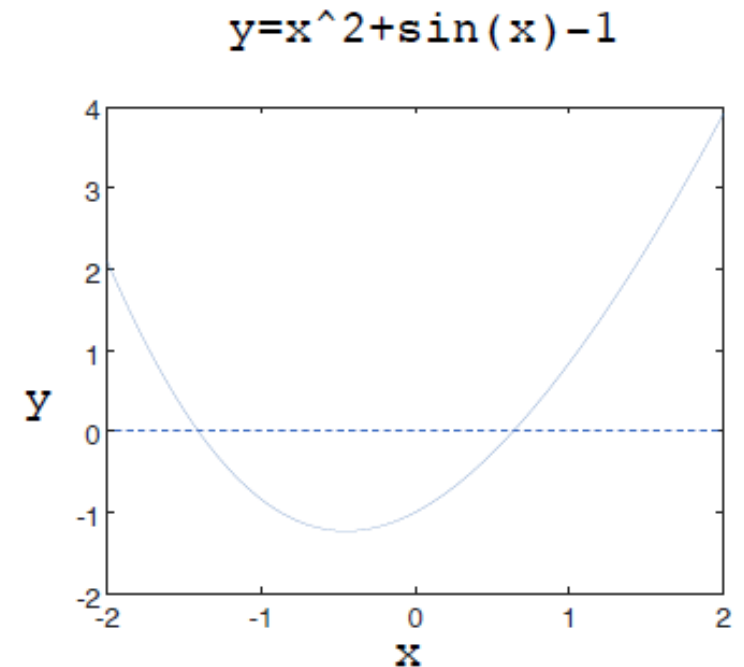
`fsolve(@(x) f(x), x0)`

関数 初期値

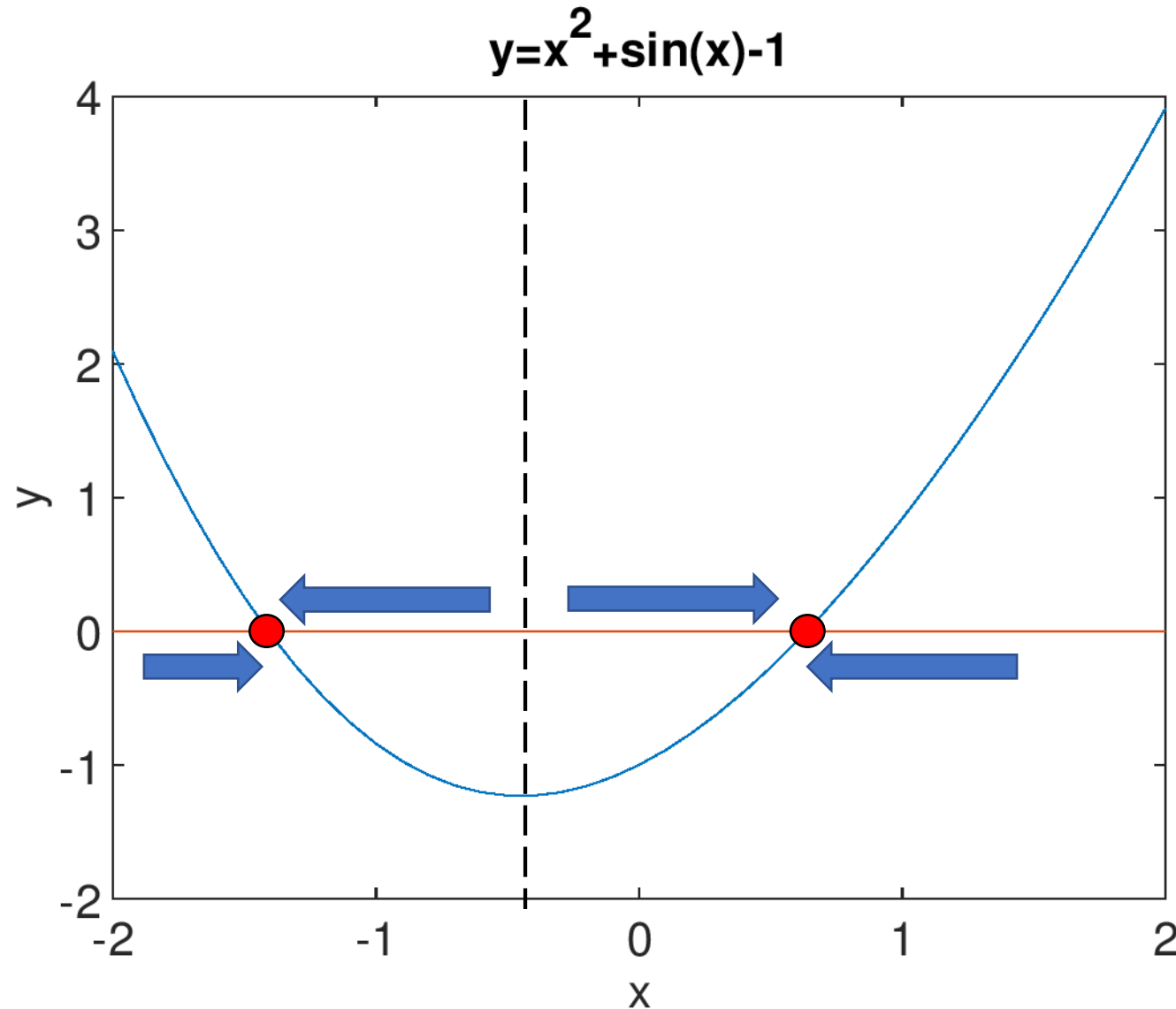
- $x^2+\sin(x)-1=0$ の解を探す

```
>> fsolve(@(x) x^2+sin(x)-1, 1.0)
ans = 0.63673
>> fsolve(@(x) x^2+sin(x)-1, -1.0)
ans = -1.4096
```

- fsolve tries to find a root starting from given initial value
- It can fail to find any root; the success depends on the equation and the provided initial values



初期値の重要性



計算には初期値が必要
(初期値の近傍から根を
探し始める)

根が複数あるとき

初期値によって得られる解
が異なる

解が得られないことも

x_0 での関数 $f(x_0)$ の値と x_0 での
関数の傾きから解は x_0 より大か
小か判断して x_0 の値を変更
それを繰り返して解を得る

Symbolic package

- Symbolic : 文字式を取り扱うこともできる
 - Function `solve` in MATLAB has not been implemented as of today
- OctaveにSymbolicをインストールするには
<https://github.com/cbm755/octsympy> で必要なpackageをダウンロードする
- Symbolic packageを使う場合は、はじめにコマンドウィンドウに

```
>> pkg load symbolic
```

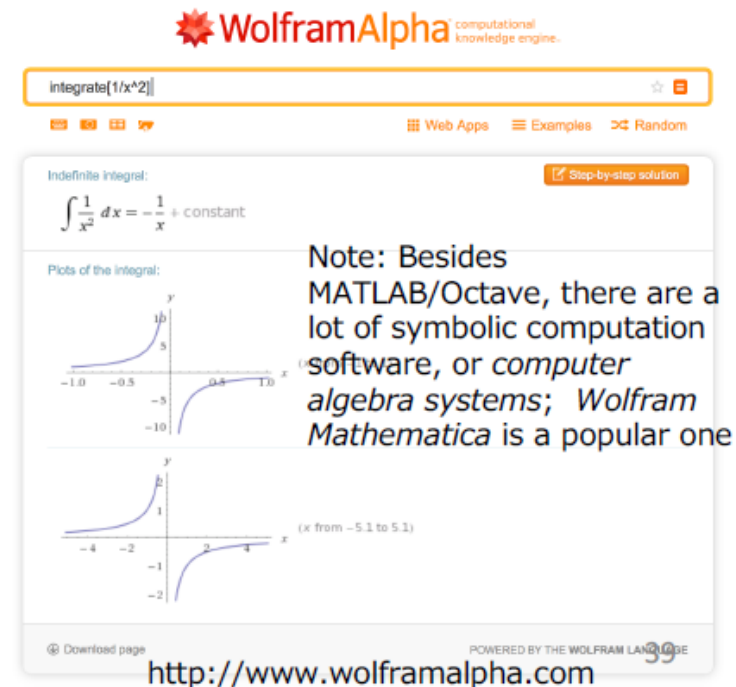
 と入力

- `syms` : 取り扱う文字式の変数を宣言する

```
>> syms x
```

- Symbolic を用いると . . .

```
>> x^2+sin(x)-1  
ans = (sym)  
      2  
      x  + sin(x) - 1
```



Symbolic package: 因数分解

- 多項式の因数分解: factor

```
>> syms x
>> f=x^3+13*x^2-105*x+171;
>> factor(f)
ans = (sym)
      2
      (x - 3) *(x + 19)
```

```
>> syms x y
>> f=x^3*y-3*x^3-4*x^2*y+12*x^2-3*x*y+9*x+18*y-54;
>> factor(f)
ans = (sym)
      2
      (x - 3) *(x + 2)*(y - 3)
```

Symbolic package: 微分

- Symbolic differential: diff

```
>> diff(x^2+sin(x)-1)
ans = (sym) 2*x + cos(x)
```

```
>> diff(exp(-x*sin(x)))  
ans = (sym)  
  
-x*sin(x)  
(-x*cos(x) - sin(x))*e
```

Symbolic package: 不定積分

- 不定積分 : int

```
>> int(x^2+sin(x)-1)
ans = (sym)
```

$$\frac{x^3}{3} - x - \cos(x)$$

```
>> int(sin(log(x)))
ans = (sym)
```

$$\frac{x \sin(\log(x))}{2} - \frac{x \cos(\log(x))}{2}$$

Exercises 4.1

- 次の方程式の“全ての”根を求めよ

$$(\sin x)^2 \exp\left(-\frac{x}{2}\right) - 0.1 = 0, \quad (x \geq 0)$$

数値的解法(fsolveを用いる)には初期値が必要であり, 初期値によって求められる解が異なる場合がある.

➡ グラフを描いて解の値に見当を付けたうえで初期値を決める.

```
>> x=0:0.01:10;  
>> y=sin(x).^2.*exp(-x/2) - 0.1;  
>> y0=zeros(1,length(x));  
>> plot(x,y,x,y0)
```