

MatLab数值计算

符号计算

符号变量的创建：

```
f = sym('符号字符串', flag), e.g. f = sym('a * x ^ 2 + b * x + c', 'real'/'unreal') syms a b c x (空格), f = a * x ^ 2 + b * x + c
```

符号函数：任意函数或多项式

符号方程：

```
代数方程：eq = sym('a * x ^ 2 + b * x + c = 0') 一阶微分方程：eq = sym('Dy - y = x')(Dy = dy/dt或dy/dx) 二阶微分方程：eq = sym('D2y - y = x')(D2y = d2y/dt2或d2y/dx2)
```

sym函数的其他应用：sym()可以进行符号和数值之间的转换，即sym(f, flag)，其中 flag为'f','r','d'。

其他运算符：det(), diag(), inv(), rank(), exp()/expm(), 合并同类项 collect(), 嵌套形式honor(), 因式分解factor(), 展开expand(), 简化多项式simplify(), 分式通分numden()([n, d] = numden(sym(x/y + y/x)))

注意：符号关系运算时只能用'=='和'~='两个运算符进行比较，不可以用其他不等号

符号微积分

```
% 符号表达式的极限
limit(F, x, a) % 当x->a时·符号表达式F的极限
limit(F, a)    % 符号表达式F的默认自变量趋近于a时的极限
limit(F)       % 默认自变量趋近于0时的极限
limit(F, x, a, 'right'/'left') % 左极限与右极限

% 符号表达式的微分
diff(F, 'x')    % 令x为自变量·对F求微分 (F也可以是矩阵)
diff(F, n)      % 对F中的默认自变量求n阶微分
diff(F, 'x', n) % 对x求n阶微分

% 符号表达式的积分
int(F) % 对默认符号求F的不定积分
int(F, x) % 对x求F的不定积分
int(F, a, b) % 对默认变量求F在[a, b]上的积分
int(F, x, a, b) % 对x求F在[a, b]上的积分

% 符号表达式的级数求和
symsum(s, a, b) % 求s中默认变量从a到b的有限和 e.g. f1 = sym('x^2'); symsum(f1, 0, n - 1)
symsum(s, x, a, b) % 求s中x从a到b的有限和
```

符号方程求解

```
% 符号代数方程求解
syms x;
eq = a * x ^ 2 + b * x + c;
solve(eq); % 默认求解eq = 0
% solve('x^2 + 2*x + 1 = 0');
solve(eq1, eq2, eq3); % 求解方程组
S = solve(eq1, eq2, eq3, var1, var2); % 若方程数量不够，则解出var1,var2的符号表达式
disp(S.var1), disp(S.var2);

% 符号微分方程求解
dsolve(f1, f2, ..., 'C', 'x'); % 对微分方程组的变量x求解，C为初始条件，系统默认变量为t
dsolve('Dx = -a * x', 'x(0) = 1', 's')

% 傅里叶变换
syms t w
f = t * exp(-t^2);
fourier(f, t, w);
```