

## Sage 快速参考: 微积分

William Stein

Sage Version 3.4

<http://wiki.sagemath.org/quickref>

GNU Free Document License, extend for your own use

### 内置常数和函数

常数:  $\pi = \text{pi}$   $e = \text{e}$   $i = \text{I} = \text{i}$

$\infty = \text{oo} = \text{infinity}$   $\text{NaN} = \text{NaN}$   $\log(2) = \text{log2}$

$\phi = \text{golden\_ratio}$   $\gamma = \text{euler\_gamma}$

$0.915 \approx \text{catalan}$   $2.685 \approx \text{khinchin}$

$0.660 \approx \text{twinprime}$   $0.261 \approx \text{merten}$   $1.902 \approx \text{brun}$

近似值:  $\text{pi.n(digits=18)} = 3.14159265358979324$

内置函数:  $\sin$   $\cos$   $\tan$   $\sec$   $\csc$   $\cot$   $\sinh$

$\cosh$   $\tanh$   $\text{sech}$   $\text{csch}$   $\text{coth}$   $\log$   $\ln$   $\exp \dots$

### 定义数学表达式

构造未定元:

`var("t u theta")` or `var("t,u,theta")`

用  $*$  表示乘法,  $\wedge$  表示乘方:

$2x^5 + \sqrt{2} = 2*x^5 + \text{sqrt}(2)$

排版: `show(2*theta^5 + sqrt(2))`  $\rightarrow 2\theta^5 + \sqrt{2}$

### 数学函数

数学函数 (可以积分, 微分等):

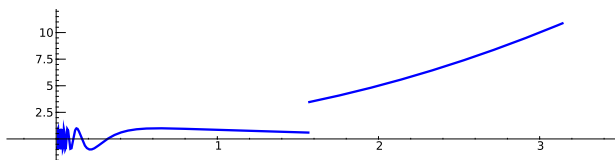
`f(a,b,theta) = a + b*theta^2`

theta 的“形式”函数:

`f = function('f',theta)`

分段函数:

`Piecewise([(0,pi/2),sin(1/x)],[(pi/2,pi),x^2+1])`



### Python 函数

定义:

`def f(a, b, theta=1):`

`c = a + b*theta^2`

`return c`

内联函数:

`f = lambda a, b, theta = 1: a + b*theta^2`

### 化简与展开

下述  $f$  均为数学 (符号) 函数 (不是 Python 函数):

化简: `f.simplify_exp()`, `f.simplify_full()`,  
`f.simplify_log()`, `f.simplify_radical()`,  
`f.simplify_rational()`, `f.simplify_trig()`

展开: `f.expand()`, `f.expand_rational()`

### 方程

关系:  $f = g: f == g$ ,  $f \neq g: f != g$ ,  
 $f \leq g: f <= g$ ,  $f \geq g: f >= g$ ,  
 $f < g: f < g$ ,  $f > g: f > g$

求解  $f = g$ : `solve(f == g, x)`, 和  
`solve([f == 0, g == 0], x,y)`

`solve([x^2+y^2==1, (x-1)^2+y^2==1],x,y)`

解:

`S = solve(x^2+x+1==0, x, solution_dict=True)`

`S[0]["x"]` `S[1]["x"]` 是解

精确根: `(x^3+2*x+1).roots(x)`

实根: `(x^3+2*x+1).roots(x,ring=RR)`

复根: `(x^3+2*x+1).roots(x,ring=CC)`

### 因式分解

分解因式: `(x^3-y^3).factor()`

列出 (因式, 幂指数) 对:

`(x^3-y^3).factor_list()`

### 极限

$\lim_{x \rightarrow a} f(x) = \text{limit}(f(x), x=a)$

`limit(sin(x)/x, x=0)`

$\lim_{x \rightarrow a^+} f(x) = \text{limit}(f(x), x=a, \text{dir}='plus')$

`limit(1/x, x=0, dir='plus')`

$\lim_{x \rightarrow a^-} f(x) = \text{limit}(f(x), x=a, \text{dir}='minus')$

`limit(1/x, x=0, dir='minus')`

### 微分

$\frac{d}{dx}(f(x)) = \text{diff}(f(x), x) = f.\text{diff}(x)$

$\frac{\partial}{\partial x}(f(x, y)) = \text{diff}(f(x, y), x)$

`diff = differentiate = derivative`

`diff(x*y + sin(x^2) + e^(-x), x)`

### 积分

$\int f(x)dx = \text{integral}(f, x) = f.\text{integrate}(x)$   
`integral(x*cos(x^2), x)`

$\int_a^b f(x)dx = \text{integral}(f, x, a, b)$   
`integral(x*cos(x^2), x, 0, sqrt(pi))`

$\int_a^b f(x)dx \approx \text{numerical\_integral}(f(x), a, b)[0]$   
`numerical\_integral(x*cos(x^2), 0, 1)[0]`

`assume(...)`: 当求积分被询问时使用

`assume(x>0)`

### Taylor 和部分分式展式

$a$  点处  $n$  次 Taylor 多项式:

$\text{taylor}(f, x, a, n) \approx c_0 + c_1(x-a) + \dots + c_n(x-a)^n$

`taylor(sqrt(x+1), x, 0, 5)`

部分分式:

`(x^2/(x+1)^3).partial_fraction()`

### 数值解和最优化

数值解: `f.find_root(a, b, x)`

`(x^2 - 2).find_root(1, 2, x)`

最大化: 寻找  $(m, x_0)$  使  $f(x_0) = m$  最大

`f.find_maximum_on_interval(a, b, x)`

最小化: 寻找  $(m, x_0)$  使  $f(x_0) = m$  最小

`f.find_minimum_on_interval(a, b, x)`

最小化: `minimize(f, start_point)`

`minimize(x^2+x*y^3+(1-z)^2-1, [1,1,1])`

### 多变量微积分

梯度: `f.gradient()` or `f.gradient(vars)`

`(x^2+y^2).gradient([x,y])`

Hessian: `f.hessian()`

`(x^2+y^2).hessian()`

Jacobian 矩阵: `jacobian(f, vars)`

`jacobian(x^2 - 2*x*y, (x,y))`

### 无穷级数求和

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

还未实现, 但你可以使用 *Maxima*:

`s = 'sum (1/n^2,n,1,inf), simpsum'`

`SR(sage.calculus.calculus.maxima(s))`  $\rightarrow \pi^2/6$