

用 Sagemath 计算微积分

王卓群

(西北农林科技大学 理学院, 陕西 杨凌 712100)

摘 要: Sagemath(简称 Sage)作为一款具有广阔前景的数学软件,它覆盖了数学研究的大部分领域,包括微积分、代数、几何、组合数学、数论以及统计。Sage 也可以应用在与数学相关领域的教学和科研工作中,它使用基于互联网的云计算,也可以实行网格进算和并行计算。主要通过介绍 Sage 的主要功能和开发背景,同时介绍一些 Sage 的基本命令,最后用高等数学书上面一些比较经典的微积分计算作为例子,比如求极限,导数,泰勒展开等,对 Matlab 和 Sage 进行对比,使人们深入了解 Sage 的功能和特性,使教学工作者可以更好地在课堂上面演示出抽象的结果。

关键词: Sagemath;微积分;数学软件;Matlab

中图分类号: TP312

文献标识码: A

文章编号: 1672-7800(2011)011-0066-03

1 始前的准备

Sage 采用浏览器作为 GUI 界面,融入了云计算的思想。学生不用再为写一次作业而装几 G 多的 Matlab 而苦恼了,他们可以不在电脑上面安装 Sage 而使用它。

1.1 进入 Sage

首先登录 Sage 的官方网站: www.sagemath.org 也可以通过搜索引擎通过搜索 Sagemath 进行寻找。

1.2 设计理念

威廉·斯坦在设计 Sage 时意识到了几个重要的事实。其一,如果要从头开始做一个可替代 Magma、Maple、Mathematica 和 Matlab 的软件将需要几百或几千年的工夫。其二,有不同的语言(常见的有 C, C++, Fortran 和 Python)编写的大量现成的大型开源数学软件可用。因此, Sage 将所有专用的数学软件集成到一个通用的接口而不是从头开发。通过选择一些选项就可以用到需要的专用数学软件,包括在统计里应用很广的 R 语言,代数方面的 GAP,同时可以在这里面直接使用 Python,这个设计也是 Sage 的特色之一。下面只用到“sage”这个选项。

2 用 Sage 计算微积分

下面是一些具体的例子,其中全部摘自《高等数学》(同济第六版),用来说明 Sage 在遇到微积分计算的时候,应该怎么去使用它。同时为了说明 Sage 和 Matlab 在语法,输出结果以及个别功能上面的差异,一些程序同时附有 Matlab 程序,方便大家进行比较。

2.1 求极限

sage: # 函数极限 1, 无穷大是 2 个小写的字母 o

```
sage: (sin(x)/x).limit(x=oo)
0
```

sage: # 也可以这样来做

```
sage: limit(sin(x)/x, x=oo)
0
```

Matlab 程序,与 Sage 的语法基本相同 >> limit(sin(x)/x, x, inf)

```
>> ans =
```

```
0
```

sage: # 左极限

```
sage: limit((x+1)/(x-7), x = 7, dir='plus')
+Infinity
```

sage: # 右极限

```
sage: limit(cos(x)/(sin(x)-1), x=pi/2, dir='minus')
-Infinity
```

2.2 求导数

sage: # 求导 1

```
sage: (sin(x)).derivative(x)
cos(x)
```

sage: # 或者这样

```
sage: diff(sin(x), x)
cos(x)
```

求导的语法基本也是一样的。

matlab 程序

```
>> diff(sin(x), x)
```

```
>> ans =
```

```
cos(x)
```

sage: # 求导 2

```
sage: diff(e^(x^3), x).show()
```

```
*
```

作者简介:王卓群(1989—),男,河南洛阳人,西北农林科技大学理学院本科生,研究方向为计算机软件技术和应用。

$$3x^2 e^x$$

matlab 程序

```
>> syms e x
```

```
>> diff(e^(x^3), x)
```

```
ans =
```

```
3 * e^(x^3) * x^2 * log(e)
```

```
>> pretty(ans)
```

```
3
```

```
x^2
```

```
3 e x log(e)
```

sage: # 求高阶导数

```
1048576x^2 e^(2x) + 20971520xe(2x) + 99614720e^(2x) *
```

```
sage: diff((x^2) * e^(2 * x), x, 20). show()
```

matlab 程序

```
>> syms e x
```

```
>> diff((x^2) * e^(2 * x), x, 20)
```

```
ans =
```

```
99614720 * e^(2 * x) * log(e)^18 + 1048576 * e^(2 * x) * x^2 * log(e)^20 + 20971520 * e^(2 * x) * x * log(e)^19
```

2.3 泰勒多项式

```
sage: f = sin(x)
```

```
sage: f1 = f.taylor(x, 0, 3)
```

```
sage: f1
```

```
-1/6 * x^3 + x
```

2.4 求不定积分

```
sage: # 不定积分
```

```
sage: f = (sin(x/2))^2
```

```
sage: g = f.integrate(x)
```

```
sage: show(g)
```

```
1/2 x - 1/2 sin(x)^2
```

matlab 程序

```
>> f = (sin(x/2))^2;
```

```
>> g = int(f);
```

```
>> pretty(g)
```

```
x sin(x)
```

```
— — — — —
```

```
2 2
```

2.5 求定积分

```
sage: # 定积分, 积分区间为 0 到 3。
```

```
sage: tstart = 0
```

```
sage: tend = 3
```

```
sage: integrate((x^2)/((x^2-3*x+3)^2), x, tstart, tend)
```

```
8/9 * pi * sqrt(3) + 1
```

```
sage: integrate((x^2)/((x^2-3*x+3)^2), x, 0, 3).
```

```
show()
```

```
8/9 pi sqrt(3) + 1 *
```

sage: # 第一项是结果, 第二项是误差项

```
sage: numerical_integral((x^2)/((x^2-3*x+3)^2), 0, 3)
```

```
(5.8367983046245806, 6.480147867887288e-14)
```

matlab 程序

```
>> int((x^2)/((x^2-3*x+3)^2), x, 0, 3)
```

```
ans =
```

```
(8 * pi * 3^(1/2))/9 + 1
```

sage: # 反常积分, 不收敛的情况

```
sage: integrate((1/x^2), x, -1, 1)
```

Traceback (click to the left of this block for traceback)

```
ValueError: Integral is divergent.
```

提示错误, 因为积分是发散的

matlab 程序

```
>> int((1/x^2), x, -1, 1)
```

```
ans =
```

```
Inf
```

这次 matlab 的计算有错误, 书上答案同样为发散, 而不是无穷大。

sage: # 反常积分, 收敛的情况

```
sage: integrate((1/sqrt(x*(x+1)^3)), x, 0, oo)
```

```
2
```

matlab 程序

```
>> int((1/sqrt(x*(x+1)^3)), x, 0, inf)
```

```
ans =
```

```
2
```

2.6 多元函数的微积分

2.6.1 偏导数

```
sage: f(x, y) = x^2 + 3 * x * y + y^2
```

```
sage: diff(f, x), diff(f, y)
```

```
((x, y) |--> 2 * x + 3 * y, (x, y) |--> 3 * x + 2 * y)
```

2.6.2 多元复合函数求偏导

```
sage: var('x, y, u, v')
```

```
sage: u = x * y
```

```
sage: v = x + y
```

```
sage: z = e^u * sin(v)
```

```
ye^xy sin(x+y) + e^(xy) cos(x+y)
```

sage: # 对 x 求导数

```
sage: z.diff(x). show()
```

sage: # 对 y 求导数

```
sage: z.diff(y). show()
```

```
xe^xy sin(x+y) + e^(xy) cos(x+y)
```

matlab 程序

```
>> syms u v x y e
```

```
>> u = x * y;
```

```
>> v = x + y;
```

```
>> z = e^u * sin(v);
```

```
>> diff(z, 'x')
```

```

ans =
e^(x*y) * cos(x + y) + e^(x*y) * y * sin(x + y)
* log(e)
>> diff(z,y)
ans =
e^(x*y) * cos(x + y) + e^(x*y) * x * sin(x + y)
* log(e)

```

2.6.3 重积分

二重积分就是对函数积分两次,语法和积分一样,注意积分的上下限范围。

```

sage: # 二重积分
sage: integrate(integrate(x*y,y,1,x),x,1,2)
9/8

```

三重积分也是一样的。

sage: # 三重积分,一个比较复杂的例子

```

sage: var('x,y,z')
sage: integrate(integrate(integrate(x,z,0,1-x-2*y),y,0,(1-x)/2),x,0,1)
1/48

```

2.6.4 含参变量的积分

sage: # Sage 定积分解不出来的情况

```

sage: integrate((ln(1+x)/(1+x^2)),x,0,1).show()

```

```

*
\int_0^1 \frac{\log(x+1)}{x^2+1} dx

```

matlab 程序

```

>> int((log(1+x)/(1+x^2)),x,0,1)
ans =
(pi*log(2))/8

```

这里 matlab 的计算结果正确。

2.7 曲线积分

```

sage: var('k,a,t,x,y,z')
sage: r=vector([a*cos(t),a*sin(t),k*t]) # 参
数坐标

```

```

sage: ds=diff(r,t).norm()

```

```

sage: ds.show()

```

```

\sqrt{-a\sin(t)^2 + |a\cos(t)|^2 + |k|^2(a^2\sin(t)^2 + a^2\cos
(t)^2 + k^2t^2)}

```

```

sage: f(x,y,z)=x^2+y^2+z^2 # 建立被积函数

```

```

sage: integrand=f(*r) * ds

```

```

sage: integrand.show()

```

```

\sqrt{-a\sin(t)^2 + |a\cos(t)|^2 + |k|^2(a^2\sin(t)^2 + a^2\cos
(t)^2 + k^2t^2)}

```

```

sage: tstart=0

```

```

sage: tend=2*pi # 计算 0 到 2*pi 的曲线积分

```

```

sage: integral(integrand, t,tstart, tend).show()

```

```

\frac{2}{3} \sqrt{a^2+k^2}(4\pi^3k^2+3ka^2)

```

2.8 无穷级数的求和

sage: # 证明级数 $1+2+3+\dots+\dots+n+\dots$ 是发散的。

```

sage: assume(k > 1)

```

```

sage: k, n = var('k,n')

```

```

sage: sum(k, k, 1, oo)

```

Traceback (click to the left of this block for trace-back)

```

ValueError: Sum is divergent. sage: # 级数要是收敛,就求和

```

```

sage: sum(1/k-1/(k+1), k, 1, oo)

```

```

1

```

```

sage: # 几何级数

```

```

sage: assume(abs(q) < 1)

```

```

ssage: sum(a*q^k, k, 0, oo)

```

```

-a/(q-1)

```

```

sage: forget()

```

```

sage: assume(q > 1)

```

```

sage: sum(a*q^k, k, 0, oo)

```

Traceback (click to the left of this block for trace-back)

```

ValueError: Sum is divergent.

```

3 结束语

如果想进一步了解 Sage 的其它功能和用法,可以在 <http://www.sagemath.org/zh/> 上面看到 Sage 的中文帮助手册,进一步去学习它。

最后需要说明的是,就目前的情况来看,Sage 还不能说是完美的。其一,因为 Sage 是开源软件的关系,所以在有些命名和用法规则上面还没用去统一,造成了使用中存在一些困扰;其二,也是最主要的,同时是阻碍 Sage 在成为流行软件的最大原因:Sage 不能直接安装在 Windows 系统下面,不过可以使用虚拟机或者在 linux 下使用,或者和文章中的方法一样,直接登录它的网站去进行计算。

参考文献:

- [1] SAGE, SAGE Mathematical Software, Version 4.7[EB/OL] <http://www.sagemath.org>
- [2] SAGE, SageTutorial-CH[EB/OL]. <http://www.sagemath.org/zh/>
- [3] SAGE, Sage Talk in Northwest A&F University, P. R. China[EB/OL]. <http://www.sagenb.org/home/pub/2868/>
- [4] 同济大学数学系. 高等数学(第六版)[M]. 北京:高等教育出版社, 2011.
- [5] 宋世德,郭满才. 数学实验(第二版)[M]. 北京:中国农业出版社, 2007.

(责任编辑:余 晓)