Python 的可视化功能在高等数学教学中的应用

吴娇 [1]　卓友濠 [2]　郑木华 [2]
1. 江苏大学数学科学学院；2. 江苏大学物理与电子工程学院

摘　要：高等数学是很多理工类专业必修的课程之一。当中涉及的内容较多，教师授课的形式较为单一，学生学习起来感觉异常枯燥。文章利用 Python 强大的可视化功能，绘制复杂函数的图像，增强高等数学教学的直观性，帮助学生更好地理解知识点，激发学生学习兴趣的同时提升教学效果。

关键词：Python 语言；高等数学；可视化；微积分
【DOI】10.12252/j.issn.2096-6288.2024.10.083

引言

高等数学在理工科类院校中是一门非常重要的基础课程 [1]。对于很多大学生来说，高等数学是一门非常难的课程。一方面，高等数学涉及的内容很多，包括数列、极限、微积分、常微分方程等等。另一方面，教师常常以单一的讲授法为主进行授课，学生学习起来感觉异常枯燥。教师们可以通过编程可视化的方式来激发学生的学习兴趣，丰富教学内容的同时提升课堂教学效果。

可视化教学是利用现代计算机程序快速计算和数字绘图等条件，将抽象的数学概念具体化，将复杂的函数绘制成图片，把函数的特性可视化 [1]。将可视化教学引入到高等数学教学中，可丰富教学内容，提高教学质量，更有助于增强学生对数学的感性认识，深入学习数学概念、公式、定理等。在高等数学教学中，借助引导学生在具体问题上进行自主编程操作，并通过自主进行的可视化研究过程，不仅能够培养学生的科学探索精神，还能有效提升他们的科研能力。特别是当学生参与到具体数学实例的数字化研究过程中，学生们将能够更加深入地理解知识，并通过计算机编程激发学生的思维和创造力，培养他们的解决问题的能力。

实现可视化的编程软件中，Python 已成为许多高等院校学生首选的编程语言。它具有语法简单，计算功能强大，数据可视化方便，资源库开源等特点。此外，Python 还广泛用于教学科学计算和数据处理，为学生提供了解决实际问题的能力，并在数学、物理等学科中具有广泛的应用 [2]。

本文利用 Python 的可视化以及科学计算功能，以高等数学部分习题为例，进行计算机辅助教学，加深学生对复杂函数的理解，促进高等数学的课堂教学。

一、Python 在极限运算中的可视化

极限在高等数学中可以帮助我们分析函数的趋势和性质，以及研究函数在某一点或在无穷远处的行为，因此极限在高等数学中占有非常重要的地位。此外，极限也是微积分的重要和核心的思想或概念。下面以数列极限为例，简单介绍 Python 的可视化及数据处理功能。

例 1. 求数列极限 $\lim\limits_{n\to\infty}\left(\frac{1}{n+\ln 1}+\frac{1}{n+\ln 2}+\cdots+\frac{1}{n+\ln n}\right)$

本例题可以利用 Python 编写程序，将求和式可视化，帮助学生理解数列极限。如图 1 所示。从图可得出当 n 趋于无穷时，其结果趋于 1。

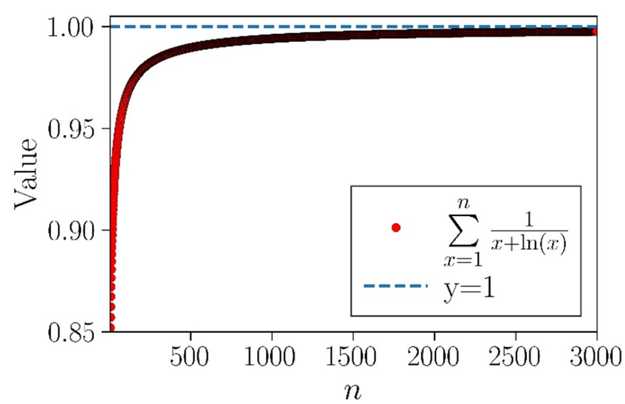

图 1　数列极限的可视化

二、 Python 在积分运算中的可视化

积分是高等数学中不可或缺的重要工具之一，也是后续内容，如微分方程、概率统计等内容的基础。通过积分，我们可以计算曲线长度、曲线下的面积、分析物理现象、解决工程问题，深化对函数行为及数学模型的理解。下面简单介绍如何利用 Python 的可视化功能帮助我们理解函数的性质和行为。

（一）Python 在定积分中的可视化以及数据处理

定积分的几何意义在于用几何图形的面积来表示函数在某个区间上的积累效应，这对于理解定积分的概念和性质非常重要。在例 3 和例 4 中，我们利用定积分比较所围图形的面积大小以及计算对数螺旋线的长度。

例 3. 比较 $\int_0^1 e^{x^2}\,dx$ 和 $\int_0^1 (1+x)^2 dx$ 的大小

我们可以用 Python 将本例题的函数图像绘制出来，并通过数形结合来比较这两个积分的大小。如图 2 所示，通过图像可以快速判断出在区间 [0,1] 内 $\int_0^1 e^{x^2}\,dx < \int_0^1 (1+x)^2 dx$。主要的代码如下：

```
# 生成 x 值的数组
```

```
x=np.arange(-2,2,0.001)
y1=np.exp(x**2)
```

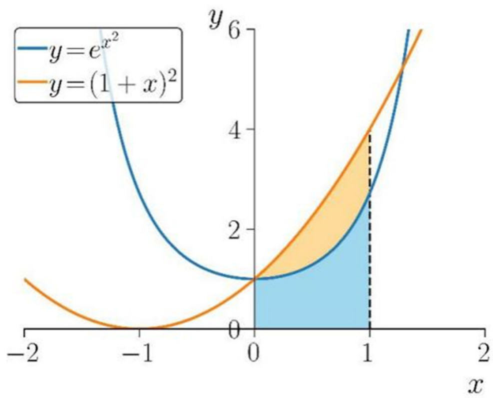

图 2 $y=e^{x^2}$ 和 $y=(1+x)^2$ 图像

```
y2=(x+1)**2# 绘制图像
fig,ax=plt.subplots()
fig.set_size_inches(6,4)
plt.plot(x,y1,linewidth=2,label=r'$y\!=\!e^{x^2}$')
plt.plot(x,y2,linewidth=2,label=r'$y\!=\!{(1+x)^2}$')
# 绘制填充区域
x0=1
ax.vlines(1.0,0,(x0+1)**2,color='black',linestyle='--')
x1=np.linspace(0,1.0,100)
plt.fill_between(x1,np.exp(x1**2),(x1+1)**2,where=(x1>=0)&(x1<= 1.0),color='orange',alpha=0.4)
plt.fill_between(x1,0,np.exp(x1**2),where=(x1>= 0) & (x1 <= 1.0),color='skyblue',alpha=0.8)
ax.yaxis.set_major_locator(tk.MultipleLocator(2))
plt.xlabel(r'$x$',loc='right')
plt.ylabel(r'$y$',loc='top',rotation=0)
ax.tick_params(axis='both',which='major')
ax.set_ylim(0 ,6)
ax.set_xlim(-2 ,2)
ax.legend()
plt.savefig('figure/figure_3.jpg',bbox_inches='tight',dpi=650)
plt.show()
```

Python 的功能并不止于此，利用 Scipy 库也可直接计算定积分的值。从而快速比较定积分的大小。

例 4. 求对数螺旋线$\rho=e^{a\theta}$在$-\frac{3\pi}{2}\leq\theta\leq\frac{3\pi}{2}, a=0.1$时的一段弧长。

解决这个问题时，虽然可以通过极坐标下的弧长元素直接积分得到答案，但这仅仅停留在答案上，对函数的图像和性质并不了解。我们可以利用 Python 绘制函数的图像、分析函数性态，更好地理解对数螺极坐标系下绘制的对数螺旋线。

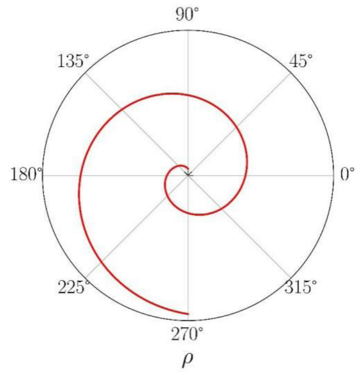

图 3 对数螺旋线在极坐标系下的图像

绘制代码如下：

```
theta=np.linspace(-3/2*np.pi,3/2*np.pi,200)
a=0.1
rho=np.exp(a*theta)
fig=plt.figure(figsize=(6,6))
ax1=fig.add_subplot(1,1,1,projection='polar')
ax1.plot(theta,rho,lw=2,color='r')
ax1.set_xlabel(r'$\rho$')
ax1.yaxis.set_ticks([])
plt.savefig('figure/figure_3.jpg',bbox_inches='tight',dpi=650)
plt.show()
```

我们也可以直接用 Python 直接积分计算

```
from scipy import integrate
def f1(theta):
a=0.1
return np.sqrt((1+a**2)*np.exp(2*a*theta))
integral_value,error=integrate.quad(f1,-3/2*np.pi,3/2*np.pi)
print("对数螺旋线的长度 :",integral_value)
```

其运行结果为：

对数螺旋线的长度 :9.826258025867817

（二）Python 在重积分上的可视化

定积分是重积分的基础，计算重积分的一个难点是确定积分区域，由于图像是三维的所以想象力薄弱的学生难以准确的描绘出积分区域[4]，利用 Python 的可视化功能能够很好地解决这个难点。通过使用 Python 绘图库如 Matplotlib，我们可以绘制函数在平面内的投影，清晰地展现积分区域的形状和范围，从而帮助学生更好地理解二重积分和三重积分。

例 5. 利用三重积分计算 $z=6-x^2-y^2$ 和 $z=\sqrt{x^2+y^2}$ 所围成的立体的体积。我们可以理论求得

$$V=\iiint_{\Omega} dv=\iint_{D_{xy}} dxdy\int_{\sqrt{x^2+y^2}}^{6-(x^2+y^2)} dz=\frac{32\pi}{3}$$

下面用 Python 绘制 $z=6-x^2-y^2$ 和 $z=\sqrt{x^2+y^2}$ 曲面图像。

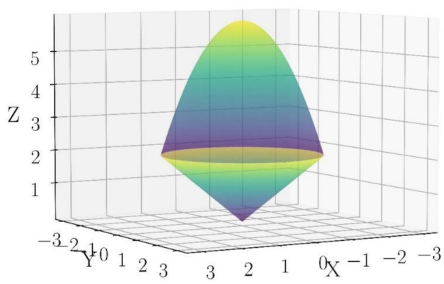

图 4 $z=6-x^2-y^2$和$z=\sqrt{x^2+y^2}$ 曲面图像

```
x=np.linspace(-3,3,1500)
y=np.linspace(-3,3,1500)
X,Y=np.meshgrid(x,y)
z1=6-X**2-Y**2
z2=np.sqrt(X**2+Y**2)
# 超出积分区域的部分设置为 nan 值
z1[X**2+Y**2>4]= np.nan
z2[X**2+Y**2>4]= np.nan
z1= np.ma.masked_invalid(z1)
z2= np.ma.masked_invalid(z2)
fig=plt.figure(figsize=(8,8))
ax=fig.add_subplot(111,projection=' 3d' )
ax.plot_surface(X,Y,z1,rstride=1,cstride=1,cmap=' viridis' ,alpha=0.5,vmin=np.nanmin(z1),vmax=np.nanmax(z1))
ax.plot_surface(X,Y,z2,rstride=1,cstride=1,cmap=' viridis' ,alpha=0.5,vmin=np.nanmin(z2),vmax=np.nanmax(z2))
ax.set_xlabel( 'X' )
ax.set_ylabel( 'Y' )
ax.set_zlabel( 'Z' )
ax.view_init(elev=5,azim=60)
plt.savefig( 'figure/figure_5.jpg' ,bbox_inches=' tight' ,dpi=650)
plt.show()
```

直接利用 Python 进行多重积分计算，求得的体积为 :33.51032163775895。

（三）Python 的其他可视化应用

对于某些复杂的 3D 函数图像，大多数学生缺乏想象能力，理解起来较为抽象。3D 图像的绘制是 Python 一大特色之一。我们可以利用 Python 的 3D 可视化功能，将函数图像绘制出来。图 5 展示了 Python 绘制的旋转抛物面 $z=x^2+y^2(0\leqslant z\leqslant 4)$ 在 xoy 面上≦≦的投影。

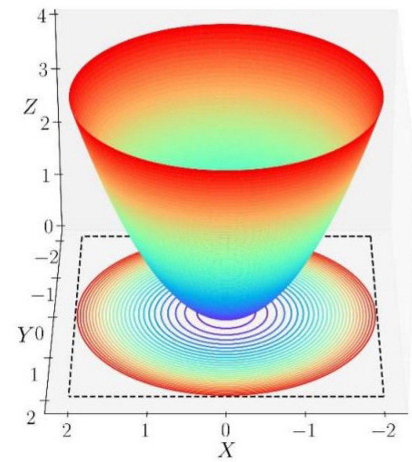

图 5 旋转抛物面 $z=x^2+y^2(0\leqslant z\leqslant 4)$ 及其在 xoy 面上的投影

结语

Python 拥有强大的可视化和数据处理功能，将 Python 运用于高等数学教学中，可将抽象的数学函数具体化、复杂的数据形象化，增强直观性，激发学生兴趣，帮助学生理解函数性态。学生可通过编程，完成相关的数据处理和可视化操作。这种实践性的方法有助于学生更直观地理解抽象的概念，同时培养他们的编程技能，提升创新能力。

参考文献

[1] 江俊勤 . 基于 Mathematica 的数字化物理学 [M]. 科学出版社 , 2015.

[2] Gezerlis A. Numerical methods in physics with Python[M]. Cambridge: Cambridge University Press, 2020.

[3] 陈向阳 . Maple 语言在高等数学教学中的应用 [J]. 广西民族大学学报 (自然科学版), 2017, 23(03): 106-108.

[4] 宋婷婷 , 王琳琳 . Python 语言在高等数学积分教学中的应用探析 [J]. 电脑知识与技术 , 2023, 19(25): 118-121.

[5] 闫新生 . Matlab 语言在高等数学教学中的应用 [J]. 山东省农业管理干部学院学报 , 2011, 28(S1): 103-105.

作者简介：吴娇，1987 年，女，汉族，安徽铜陵人，江苏大学，212013，博士研究生 / 讲师，研究方向：复杂系统与复杂网络。