# 数值实验报告 I

<table>
<tr><td>实验名称</td><td colspan="3">计算方法上机实践</td><td>实验时间</td><td colspan="3">2025 年 5 月 10 日</td></tr>
<tr><td>姓名</td><td>秦浩政<br>郭凯平<br>刘桂凡<br>刘佳鑫</td><td>班级</td><td>数据<br>2301</td><td>学号</td><td>2306030214<br>2306020510<br>2309050116<br>2309050117</td><td>成绩</td><td></td></tr>
</table>

**一、实验目的，内容**

**6.1 复化求积方法求数值积分**

分别用复化梯形公式，复化 Simpson 公式和高斯勒让德公式求解数值积分

**9.1 用单步法求解微分方程初值问题**

（1）欧拉方法（2）预报校正欧拉方法

**二、算法描述**

**实验 6.1**

复化梯形公式：将积分区间 [a,b] 分成 n 个等宽子区间，每个子区间上用梯形面积近似积分。

单个子区间 [xi,xi+1] 的梯形面积。

复化 Simpson 公式：将区间 [a,b] 分成 n 个子区间（n 必须为偶数），每个子区间上用二次抛物线近似积分。 单个子区间 [xi,xi+2] 的 Simpson 面积。

高斯勒让德公式：在区间 [−1,1] 上选择最优的积分点（高斯点）和权重，通过加权求和近似积分。

**实验 9.1**

欧拉公式：欧拉方法是一种最简单的数值积分方法，通过**线性近似**（即用切线代替曲线）来逼近微分方程的解。

预报校正欧拉公式：在欧拉法的基础上进行改进，用欧拉方法预测下一个点的近似值 ypred ，然后用预测值和当前值的导数平均值校正结果。

**三．程序代码**

**实验 6.1**

```
# 复化梯形公式
2 个用法
def composite_tixing(f, a, b, n):
    h = (b - a) / n
    result = 0.5 * (f(a) + f(b))
    for k in range(1, n):
        result += f(a + k * h)
    result *= h
    return result
```

```
# 复化Simpson公式
2 个用法
def composite_simpson(f, a, b, n):
    h = (b - a) / n
    s = f(a) + f(b)
    for k in range(1, n):
        xk = a + k * h
        xk2 = xk + 1 / 2 * h
        s += 2 * f(xk)
        s += 4 * f(xk2)
    s = s + 4 * f(a + 1 / 2 * h)
    s *= h / 6
    return s
```

```python
# Gauss-Legendre积分公式
2 个用法
def gauss_legendre(f, a, b, n):
    # 获取Gauss-Legendre积分点和权重
    nodes, weights = np.polynomial.legendre.leggauss(n)
    # 将积分从[a,b]映射到[-1,1]
    transformed = [((b - a) / 2 * x + (a + b) / 2) for x in nodes]
    result = (b - a) / 2 * sum(weights[i] * f(transformed[i]) for i in range(n))
    return result
```

```python
def f1(x):
    return math.sqrt(1 + x ** 2)

3 个用法
def f2(x):
    return math.exp(math.sin(x))
```

实验 9.1

```python
# 绘制结果
plt.figure(figsize=(10, 6))
plt.plot( *args: x_euler, y_euler, 'o-', label='欧拉方法')
plt.plot( *args: x_improved_euler, y_improved_euler, 's-', label='改进的欧拉方法',color = 'green')
plt.plot( *args: x_exact, y_exact, '-', label='精确解', color='red')
plt.xlabel('x')
plt.ylabel('y')
plt.title(r'单步方法求解 y\' = y - (2x)/y, y(0)=1')
plt.legend()
plt.grid(True)
plt.show()
```

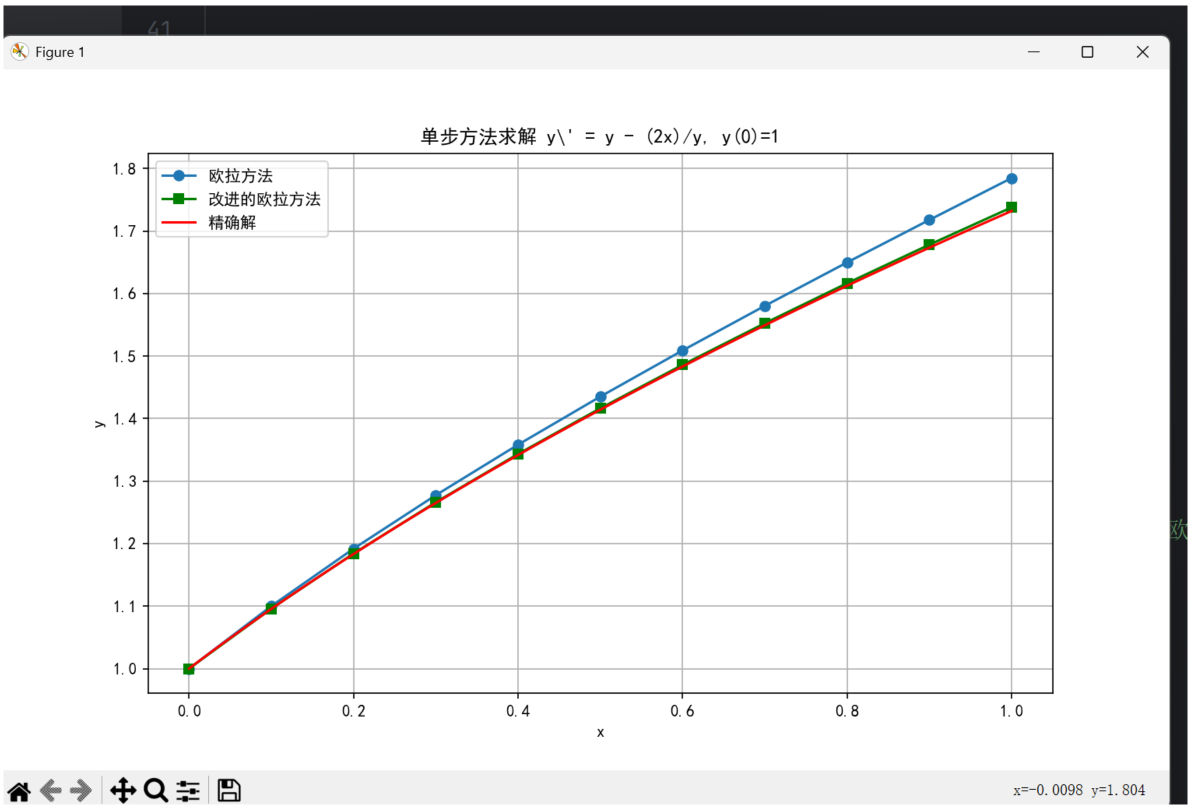
## 四. 数值结果

实验 6. 1

```
精确值是: 1.147793574696319
复化梯形公式计算∫[0,1] sqrt(1 + x^2) dx: 1.1487144663927218
复化Simpson公式计算∫[0,1] sqrt(1 + x^2) dx: 1.1477932805236797
Gauss-Legendre公式计算∫[0,1] sqrt(1 + x^2) dx: 1.147793580922107

精确值是: 6.20875803571111
复化梯形公式计算∫[0,π] exp(sin(x)) dx: 6.183057933280413
复化Simpson公式计算∫[0,π] exp(sin(x)) dx: 6.208747154919542
Gauss-Legendre公式计算∫[0,π] exp(sin(x)) dx: 6.209471456584282
```

实验 9. 1




```
欧拉方法结果:
x = 0.0, y = 1.000000
x = 0.1, y = 1.100000
x = 0.2, y = 1.191818
x = 0.3, y = 1.277438
x = 0.4, y = 1.358213
x = 0.5, y = 1.435133
x = 0.6, y = 1.508966
x = 0.7, y = 1.580338
x = 0.8, y = 1.649783
x = 0.9, y = 1.717779
x = 1.0, y = 1.784771

预报-校正欧拉方法结果:
x = 0.0, y = 1.000000
x = 0.1, y = 1.095909
x = 0.2, y = 1.184097
x = 0.3, y = 1.266201
x = 0.4, y = 1.343360
x = 0.5, y = 1.416402
x = 0.6, y = 1.485956
x = 0.7, y = 1.552514
x = 0.8, y = 1.616475
x = 0.9, y = 1.678166
x = 1.0, y = 1.737867
```

## 五．计算结果分析

**实验 6.1**

通过对比近似值和精确值的结果，以及将区间 n 等分的 n 的大小，复化梯形将区间 8 等分，复化 Simpson 和高斯勒让德都是将区间 5 等分，但是最终的结果是后面两种方法更接近真实值，这说明，复化梯形公式的收敛速度很慢，需要将区间进行多次等分才能得到一个比较好的近似效果，而复化 Simpson 和高斯勒让德公式的收敛素的比较快，但是复化梯形公式原理更加简单，不需要复杂的运算，复化 Simpson 和高斯勒让德可以很快的收敛到精确解，适用于高精度要求且函数比较复杂的情况。

**实验 9.1**

通过结果可以看到，改进后的欧拉公式，相比于原来的欧拉公式和精确解的重合度很高，预报和校正两步计算，显著提高了精度，但是欧拉公式每步只需一次函数求值，计算量小，而预报校正欧拉公式每步需要两次函数求值（预报和校正），计算量增加，但是这样显著提高了精度，

## 六. 计算中出现的问题，解决方法及体会

**实验 6.1**

**问题：** 复化梯形公式收敛太慢

复化梯形方法是一阶数值积分方法（局部截断误差为 O(h2)，全局误差为 O(h2) 随区间数增加），对于光滑函数，其精度低于高阶方法。所以如果被积函数具有较高的导数或震荡特性，复化梯形方法可能无法有效捕捉函数的变化，导致误差较大。

**解决办法:**

（1）减小步长 h 可以降低误差，但会增加计算量。 适用于对精度要求较高且计算资源充足的情况。

（2）自适应方法根据函数的变化自动调整步长，在函数变化剧烈的区域使用较小的步长，在函数变化平缓的区域使用较大的步长。（事后估计法）

**实验 9.1**

**问题：**Euler 方法的不准确

Euler 方法由于采用一阶近似，该方法仅使用当前点的斜率进行预测，未考虑函数曲率变化，导致局部截断误差较大，随着计算步数的增加，误差呈现明显的线性增长趋势。

**解决办法：**减小步长：通过缩小步长，可以降低单步误差，但会增加计算量。但是步长越小，精度越高，但计算成本增加，所以需要更高精度的情况我们采用预报校正欧拉法。

| | |
|---|---|
| 教师评语 | 指导教师：　　　　　　　　　年　　月　　日 |