答卷

汤尔扬 25300180038

2025/11/27

1 第一题

1.1 i

```python
import numpy as np

def matrix_stats(mat):
    arr = np.array(mat)
    return {
        'max': float(np.max(arr)),
        'min': float(np.min(arr)),
        'mean': float(np.mean(arr)),
        'std': float(np.std(arr)),
        'sr': float(np.mean(arr)) / float(np.std(arr)),
        'sum': float(np.sum(arr))
    }

if __name__ == '__main__':
    pass
```

以上为修改后的代码

1.2 ii

1.2.1 A

'max': 6.0

'min': 1.0

'mean': 3.5

'std': 1.707825127659933

'sr': 2.04939015319192

'sum': 21.0

1.2.2 B

'max': 1.0

'min': -1.0

'mean': 0.0

'std': 0.7071067811865476

'sr': 0.0

'sum': 0.0

1.3 iii

问题 1：要求输出数据为整数或浮点数，而原程序输出会显示 np.float64 和 np.int64，因此将这些数据统一改为浮点数来输出。（已修改）

问题 2：若所有元素相同，np.std(arr) = 0，则计算'sr' 时会报错，因此要分情况，若有 np.std(arr) = 0，则不输出'sr'。

正常输入 1：[[1,2,3],[4,5,6],[7,8,9]]

结果：'max': 9.0, 'min': 1.0, 'mean': 5.0, 'std': 2.581988897471611, 'sr': 1.9364916731037085, 'sum': 45.0

正常输入 2：[[1,2]]

结果：'max': 2.0, 'min': 1.0, 'mean': 1.5, 'std': 0.5, 'sr': 3.0, 'sum': 3.0

正常输入 3：[[1],[2]]

结果：'max': 2.0, 'min': 1.0, 'mean': 1.5, 'std': 0.5, 'sr': 3.0, 'sum': 3.0

边界输入：[[1]]

结果：ZeroDivisionError: float division by zero

异常输入：[[1,1,1],[1,1]]

结果：setting an array element with a sequence. The requested array has an inhomogeneous shape after 1 dimensions. The detected shape was (2,) + inhomogeneous part.

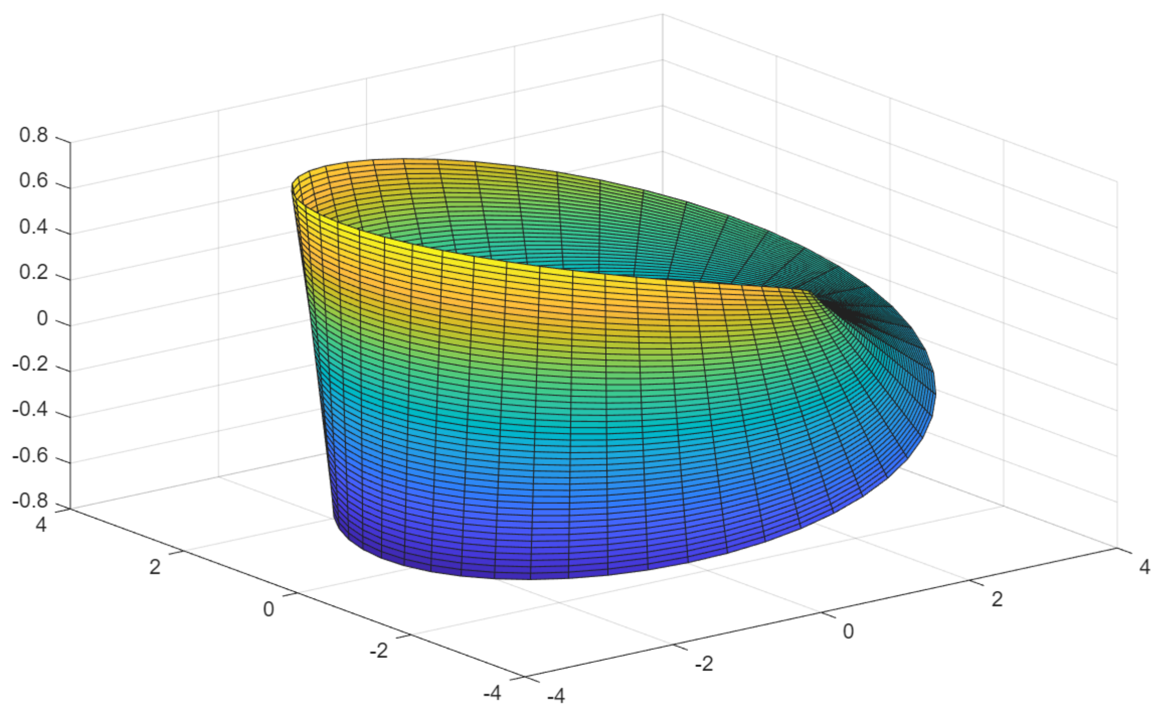
全部修改完的代码和示例：

```python
import numpy as np

def matrix_stats(mat):
    arr = np.array(mat)
    if float(np.std(arr)) != 0:
        return {
            'max': float(np.max(arr)),
            'min': float(np.min(arr)),
            'mean': float(np.mean(arr)),
            'std': float(np.std(arr)),
            'sr': float(np.mean(arr)) / float(np.std(arr)),
            'sum': float(np.sum(arr))
        }
    else:
        return {
            'max': float(np.max(arr)),
            'min': float(np.min(arr)),
            'mean': float(np.mean(arr)),
            'std': float(np.std(arr)),
            'sum': float(np.sum(arr))
        }

if __name__ == '__main__':
    print(matrix_stats([[1,2,3],[4,5,6],[7,8,9]]))
    print(matrix_stats([[1,2]]))
    print(matrix_stats([[1],[2]]))
    print(matrix_stats([[1]])) #边界
    print(matrix_stats([[1,1,1],[1,1]])) #异常,会报错
```

2 第二题

Matlab 命令:

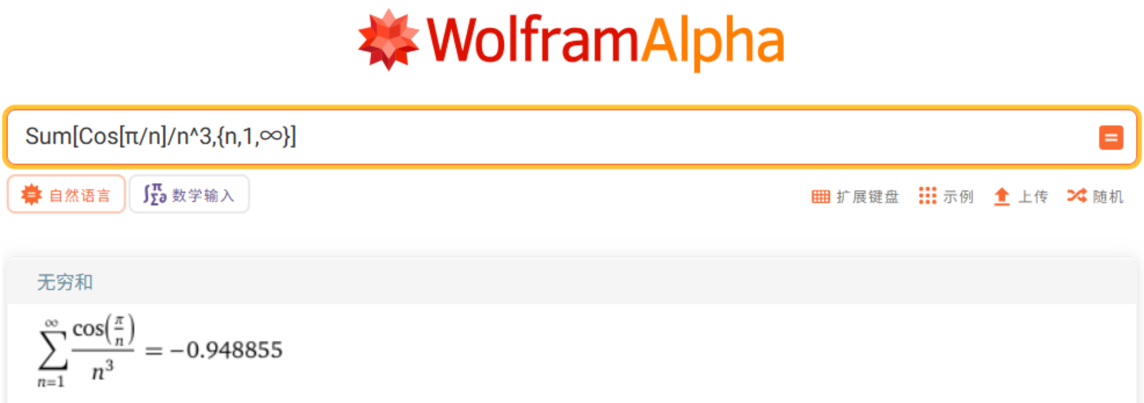
```
R = 3;
r = 0.7;

u = linspace(-r, r, 50);
v = linspace(0, 2*pi, 50);

[U, V] = meshgrid(u, v);

X = (R + U .* cos(V .* 0.5)) .* cos(V);
Y = (R + U .* cos(V .* 0.5)) .* sin(V);
Z = U .* sin(V .* 0.5);

figure;
surf(X, Y, Z);
```



3 第三题

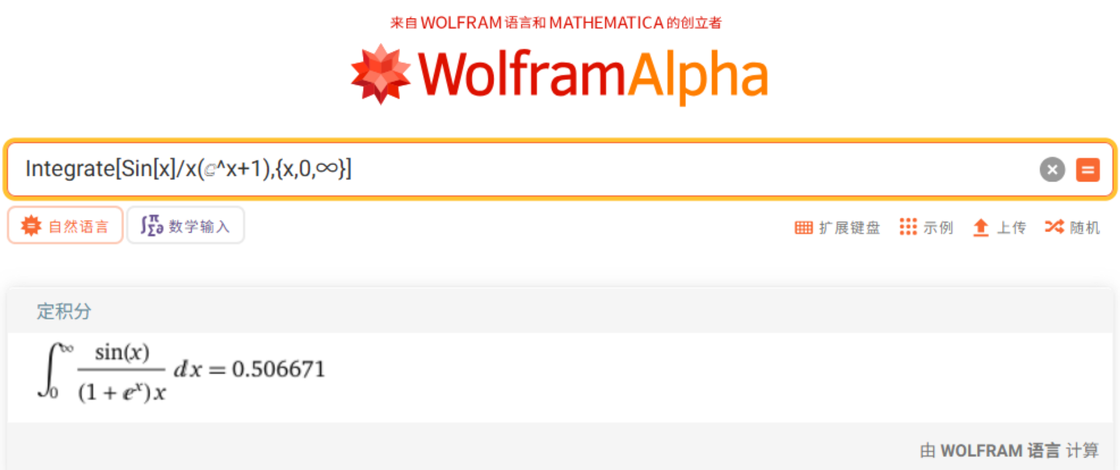
3.1 i

答案：-0.948855



3.2 ii

答案：0.506671



4 第四题

Linear Least Squares

Linear least squares (LLS) is the least squares approximation of linear functions to data. It is a set of formulations for solving statistical problems involved in linear regression, including variants for ordinary (unweighted), weighted, and generalized (correlated) residuals. Numerical methods for linear least squares include inverting the matrix of the normal equations and orthogonal decomposition methods.

Basic Formulation

Consider the linear equation

$$Ax = b \tag{1}$$

where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are given and $x \in \mathbb{R}^n$ is variable to be computed. When $m > n$, it is generally the case that Eq.(1) has no solution.
For example, there is no value of x that satisfies

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} x = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

because the first two rows require that $x = (1, 1)$, but then the third row is not satisfied. Thus, for $m > n$ the goal of solving Eq. (**??**) exactly is typically replaced by finding the value of x that minimizes some error. There are many ways that the error can be defined, but one of the most common is to define it as $\|Ax - b\|^2$.
This produces a minimization problem, called a least squares problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \|Ax - b\|^2$$

The solution to the least squares problem is computed by solving the normal equation

$$A^T Ax = A^T b$$

where A^T denotes the transpose of the matrix A.