Finalexam-part2

23307130428 姚馨悦

1、python编程

```
import numpy as np

def matrix_stats(mat):
    arr = np.array(mat)
    mean_val = np.mean(arr)
    std_val = np.std(arr)

    if std_val == 0:
        sr_val = np.nan if mean_val == 0 else (np.inf if mean_val > 0 else -np.inf)
    else:
        sr_val = mean_val / std_val

    return {
        'max': np.max(arr),
        'min': np.min(arr),
        'mean': mean_val,
        'std': std_val,
        'sr': sr_val,
        'sum': np.sum(arr),
    }

if __name__ == '__main__':

    A = np.array([[1, 2, 3], [4, 5, 6]])
    B = np.array([[-1, 0], [0, 1]])

    print(f"matrix_stats(A): {matrix_stats(A)}")
    print(f"matrix_stats(B): {matrix_stats(B)}")

    # Test Case 1: 正常输入
    C1 = [[1, 2, 3], [4, 5, 6]]
    print(C1)
    print(f"matrix_stats(C1): {matrix_stats(C1)}")

    # Test Case 2: 正常输入 - 包含负数和零
    C2 = [[-5, -3, 0], [2, 4, 8]]
    print(f"matrix_stats(C2): {matrix_stats(C2)}")

    # Test Case 3: 正常输入 - 浮点数矩阵
    C3 = [[1.5, 2.5], [3.5, 4.5]]
    print(f"matrix_stats(C3): {matrix_stats(C3)}")

    # Test Case 4: 边界输入 - 单个元素矩阵(1x1)
    C4 = [[42]]
    print(f"matrix_stats(C4): {matrix_stats(C4)}")

    # Test Case 5: 边界输入 - 一维矩阵
```

```python
    C5 = [1, 2, 3, 4, 5]
    print(f"matrix_stats(C5): {matrix_stats(C5)}")

    # Test Case 6: 异常输入 - 空矩阵
    C6 = []
    try:
        print(f"matrix_stats(C6): {matrix_stats(C6)}")
    except Exception as e:
        print(f"{type(e).__name__}: {e}")

    # Test Case 7: 异常输入 - 不规则矩阵
    C7 = [[1, 2], [3, 4, 5], [6]]
    try:
        print(f"matrix_stats(C7): {matrix_stats(C7)}")
    except Exception as e:
        print(f"{type(e).__name__}: {e}")
```

- 计算A、B的统计量：

  matrix_stats(A): {'max': np.int64(6), 'min': np.int64(1), 'mean': np.float64(3.5), 'std': np.float64(1.707825127659933), 'sr': np.float64(2.04939015319192), 'sum': np.int64(21)}

  matrix_stats(B): {'max': np.int64(1), 'min': np.int64(-1), 'mean': np.float64(0.0), 'std': np.float64(0.7071067811865476), 'sr': np.float64(0.0), 'sum': np.int64(0)}

- 发现bug：求'sr'除0时（std_val == 0）会产生异常。故额外增加判断。

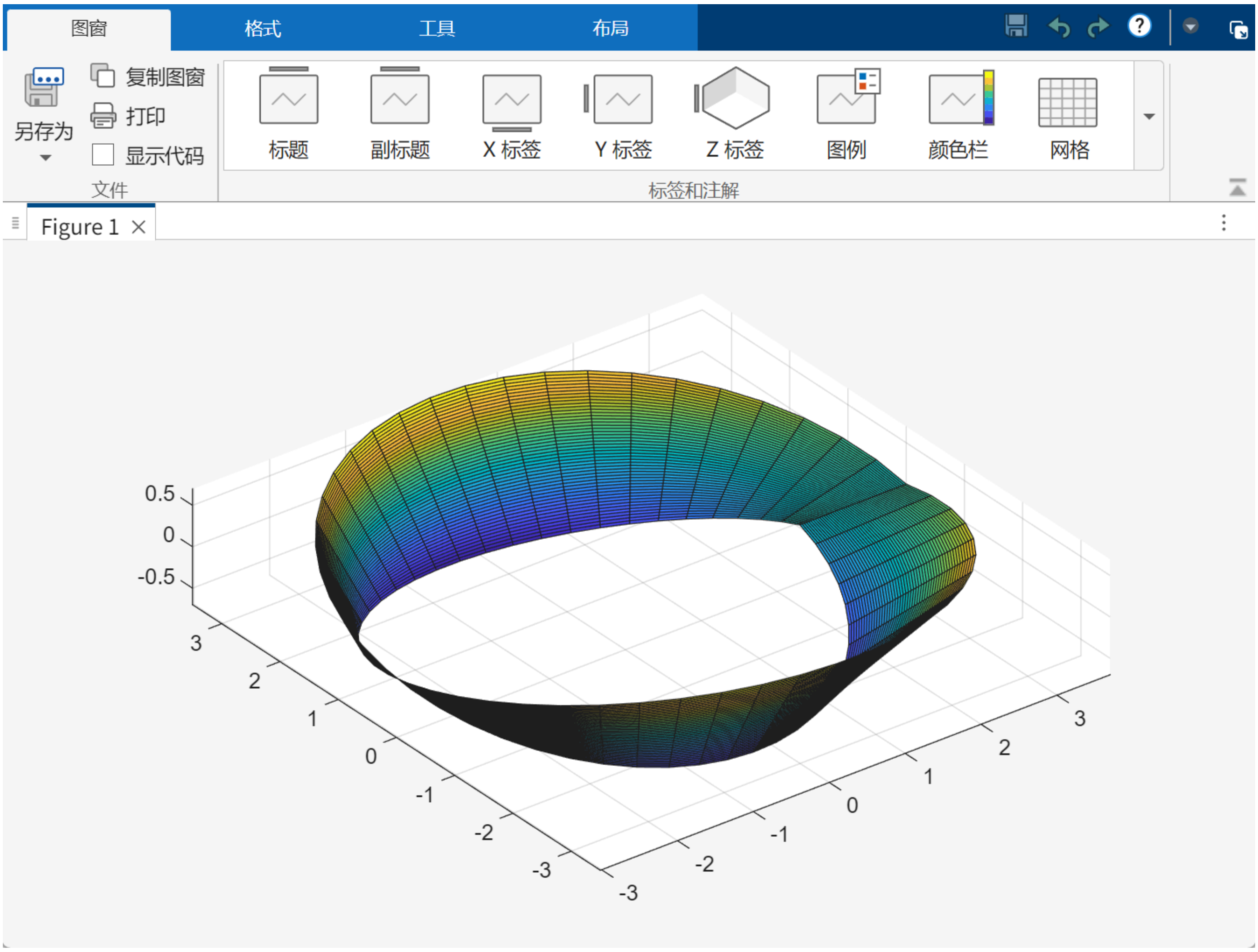
2、Matlab作图

```matlab
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/2)) .* cos(v);
Y = (R + u.*cos(v/2)) .* sin(v);
Z = u .* sin(v);

surf(X, Y, Z);
axis equal
```



3、利用Mathematica

1、求如下无穷级数的和：

```
Sum[Cos[Pi/n]/n^3,{n,1,Infinity}]
```

2、求如下定积分的值

```
Integrate[Sin[x] / (x (Exp[x]+1)^2),{x, 0,nfinity}]
```

```
In[38]:= NSum[Cos[Pi / n] / n ^ 3, {n, 1, Infinity}]

Out[38]= -0.948855

In[39]:= NIntegrate[Sin[x] / (x (Exp[x] + 1) ^ 2), {x, 0, Infinity}]

Out[39]= 0.170971
```

4、渲染

Linear Least Squares

Linear least squares (LLS) is the least squares approximation of linear functions to data. It is a set offormulations for solving statistical problems involved in linear regression, including variants forordinary (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

Connsider the linear equation

$$\mathbf{Ax}{=}\mathbf{b}$$

where $\mathbf{A} \in R^{m \times n}$ and $\mathbf{b} \in R^{m}$ 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. (1) exactly is typically replaced by finding the value of x that minimiezes some error. There are many ways that the erroe can be defined, but one of the most common is to define it as $||\mathbf{Ax} - \mathbf{b}||^2$. This produces a minimization problem, called a least squares problem

$$minimize_{x \in R^n} ||\mathbf{Ax} - \mathbf{b}||^2$$

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

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$$

where $\mathbf{A}^T$ denotes the transpose of the matrix $\mathbf{A}$.