数据处理常用库: pandas

1. 什么是 Pandas?

Pandas 是一个开源的 Python 数据分析库，提供了高性能、易于使用的数据结构和数据分析工具。它主要用于处理和分析结构化数据，如表格数据。

2. Pandas 的核心数据结构

2.1 Series

Series 是 Pandas 中的一维标签数组，类似于带标签的一维数组。它是一种灵活的数据结构，可以容纳不同类型的数据。

```
import pandas as pd



data = [10, 20, 30, 40, 50]

s = pd.Series(data, index=['a', 'b', 'c', 'd', 'e'])

print(s)
```

2.2 DataFrame

DataFrame 是 Pandas 中的二维表格数据结构，类似于 Excel 表格。每个列可以是不同的数据类型。

```
data = {'Name': ['Alice', 'Bob', 'Charlie'],

'Age': [25, 30, 35],

'Country': ['USA', 'Canada', 'UK']}

df = pd.DataFrame(data)
```

```
print(df)
```

3. 读取文件内容

Pandas 支持从多种文件格式中读取数据，包括 CSV 和 Excel 文件。

3.1 读取 CSV 文件

```
csv_file = 'data.csv'

df = pd.read_csv(csv_file)

print(df)
```

3.2 读取 Excel 文件

```
excel_file = 'data.xlsx'

df = pd.read_excel(excel_file)

print(df)
```

4. 数据处理和分析

4.1 数据选择与过滤

```
# 选择列

ages = df['Age']
```

```
# 条件过滤

filtered_df = df[df['Age'] > 28]
```

4.2 数据聚合和分组

```
# 按国家分组并计算平均年龄

grouped = df.groupby('Country')

average_age = grouped['Age'].mean()
```

5. 异常处理

Pandas 提供了处理缺失值和异常值的功能。

```
# 处理缺失值

df.dropna() # 删除包含缺失值的行

df.fillna(value) # 用指定值填充缺失值



# 异常值处理

df[df['Age'] > 100] = np.nan
```

6. 实战: 数据预处理

我们通常结合使用pandas与numpy

```
import numpy as np
```

```
import pandas as pd



# 创建一个示例数据集

data = {'Age': [25, 30, np.nan, 35, 40],

'Income': [50000, np.nan, 60000, 75000, 80000],

'Gender': ['M', 'F', 'M', 'F', 'M']}



df = pd.DataFrame(data)



# 使用平均值策略处理缺失值

df['Age'].fillna(df['Age'].mean(), inplace=True)

df['Income'].fillna(df['Income'].mean(), inplace=True)



# 将 Gender 字段进行离散化

df['Gender'] = pd.Categorical(df['Gender'], categories=['M', 'F'],
ordered=True)

df['Gender'] = df['Gender'].cat.codes



# 将pandas对象转为numpy数组方便后续计算

data = df.to_numpy()

print('处理后的数据:')

print(df)
```

```
print(f'data={data}')
```

处理后的数据:

```
Age Income Gender

0 25.0 50000.0 0

1 30.0 68750.0 1

2 32.5 60000.0 0

3 35.0 75000.0 1

4 40.0 80000.0 0
```

```
data = [[2.500e+01 5.000e+04 0.000e+00]

[3.000e+01 6.625e+04 1.000e+00]

[3.250e+01 6.000e+04 0.000e+00]

[3.500e+01 7.500e+04 1.000e+00]

[4.000e+01 8.000e+04 0.000e+00]]
```