

Pandas 数据清洗与数据分析

jupyter打开输入，然后运行：

```
!pip install pandas matplotlib -i https://pypi.tuna.tsinghua.edu.cn/simple
```

模块一 缺失值检测与处理

任务 1 检测缺失值位置

```
# 导入库
import pandas as pd
import numpy as np

# 自动生成测试成绩表
df = pd.DataFrame({
    '语文': [80, 90, None, 70],
    '数学': [60, 75, 88, 92],
    '英语': [55, np.nan, np.nan, 90]
})

# 检测缺失值：True=缺失，False=正常
print("缺失值位置标记：")
print(pd.isnull(df))
```

任务 2 统计缺失值数量与占比

```
import pandas as pd
import numpy as np

df = pd.DataFrame({
    '语文': [80, 90, None, 70],
    '数学': [60, 75, 88, 92],
    '英语': [55, np.nan, np.nan, 90]
})

# 定义缺失值统计函数
def 统计缺失值(df):
    缺失数量 = df.isnull().sum()
    缺失占比 = df.isnull().sum() / len(df) * 100
    res = pd.DataFrame({
        '缺失数量': 缺失数量,
        '缺失占比(%)': 缺失占比
    })
    return res

# 执行统计
print("缺失值统计报表：")
```

```
print(统计缺失值(df))
```

任务 3 删除含缺失值的行

```
import pandas as pd
import numpy as np

df = pd.DataFrame({
    '语文': [80, 90, None, 70],
    '数学': [60, 75, 88, 92],
    '英语': [55, np.nan, np.nan, 90]
})

# 删除有缺失值的整行
新表格 = df.dropna()
print("删除缺失值后: ")
print(新表格)
```

任务 4 填充缺失值（固定值 + 平均值）

```
import pandas as pd
import numpy as np

df = pd.DataFrame({
    '语文': [80, 90, None, 70],
    '数学': [60, 75, 88, 92],
    '英语': [55, np.nan, np.nan, 90]
})

# 1. 固定值66填充
填充1 = df.fillna(66)
print("=== 固定值填充 ===")
print(填充1)

# 2. 列平均值填充
填充2 = df.fillna(df.mean()).round(1)
print("\n=== 平均值填充 ===")
print(填充2)
```

模块二 重复值检测与处理

任务 5 检测重复行

```
import pandas as pd

# 学生信息表
df = pd.DataFrame({
    '姓名': ['张三', '李四', '王五', '赵六', '赵六'],
    '年龄': [20, 21, 22, 23, 23],
    '班级': ['1班', '1班', '2班', '2班', '2班']
})

# 标记重复行: True=重复
print("重复行标记: ")
print(df.duplicated())
```

任务 6 删除重复行

```
import pandas as pd

df = pd.DataFrame({
    '姓名': ['张三', '李四', '王五', '赵六', '赵六'],
    '年龄': [20, 21, 22, 23, 23],
    '班级': ['1班', '1班', '2班', '2班', '2班']
})

# 删除重复行, 重置索引
新表格 = df.drop_duplicates(ignore_index=True)
print("删除重复行后: ")
print(新表格)
```

模块三 异常值检测与处理

任务 7 3σ 原则检测异常值

```
import pandas as pd
import numpy as np

# 含异常成绩: 10、200
df = pd.DataFrame({'数学成绩': [60, 75, 88, 92, 10, 200, 85]})

#  $3\sigma$ 找异常值
def 检测异常值(ser):
    mean = ser.mean()
    std = ser.std()
    条件 = (ser < mean - 3*std) | (ser > mean + 3*std)
    return ser[条件]
```

```
print("数学成绩异常值：")
print(检测异常值(df['数学成绩']))
```

任务 8 箱形图可视化检测异常值

```
import pandas as pd
import matplotlib.pyplot as plt

# 解决中文乱码
plt.rcParams['font.sans-serif'] = ['SimHei']
plt.rcParams['axes.unicode_minus'] = False

df = pd.DataFrame({'数学成绩': [60, 75, 88, 92, 10, 200, 85]})
# 绘制箱形图，圆圈为异常值
df.boxplot(column='数学成绩')
plt.title('数学成绩异常值箱形图')
plt.show()
```

任务 9 替换异常值

```
import pandas as pd

df = pd.DataFrame({'数学成绩': [60, 75, 88, 92, 10, 200, 85]})

# 批量替换：10→60 200→90
新表格 = df.replace([10, 200], [60, 90])
print("异常值替换后：")
print(新表格)
```

模块四 数据类型转换

任务 10 astype 强制类型转换

```
import pandas as pd

# 文字型数字
df = pd.DataFrame({'分数': ['80', '90', '75', '66']})
print("转换前类型：", df['分数'].dtype)

# 转为整型
df['分数'] = df['分数'].astype('int')
print("转换后类型：", df['分数'].dtype)
print(df)
```

任务 11 to_numeric 智能类型转换

```
import pandas as pd

# 包含非法字符
ser = pd.Series(['80', '90', 'abc', '75'])

# 非法字符转为NaN, 不报错
新数据 = pd.to_numeric(ser, errors='coerce')
print("智能转换结果: ")
print(新数据)
```

模块五 表格合并

任务 12 纵向堆叠（上下拼接）

```
import pandas as pd

表1 = pd.DataFrame({'姓名': ['张三', '李四'], '语文': [80, 90]})
表2 = pd.DataFrame({'姓名': ['王五', '赵六'], '语文': [70, 85]})

# 上下合并, 重置索引
合并表 = pd.concat([表1, 表2], ignore_index=True)
print("纵向拼接结果: ")
print(合并表)
```

任务 13 主键合并（左右关联）

```
import pandas as pd

语文表 = pd.DataFrame({'姓名': ['张三', '李四', '王五'], '语文': [80, 90, 70]})
数学表 = pd.DataFrame({'姓名': ['张三', '李四', '赵六'], '数学': [60, 75, 88]})

# 按姓名字段左连接
合并表 = pd.merge(语文表, 数学表, on='姓名', how='left')
print("主键关联合并: ")
print(合并表)
```

任务 14 索引合并 join

```
import pandas as pd

表1 = pd.DataFrame({'语文':[80,90,70]})
表2 = pd.DataFrame({'数学':[60,75,88]})

# 按行索引直接合并
合并表 = 表1.join(表2)
print("索引合并结果: ")
print(合并表)
```

任务 15 重叠数据填充 combine_first

```
import pandas as pd
import numpy as np

表1 = pd.DataFrame({'语文':[np.nan, 90, 80], '数学':[60, np.nan, 88]})
表2 = pd.DataFrame({'语文':[85, 95, 75], '数学':[65, 70, 90]})

# 用表2填补表1缺失值
新表 = 表1.combine_first(表2)
print("重叠数据补全: ")
print(新表)
```

模块六 数据重塑、透视、分箱与哑变量

任务 16 行列重塑 stack/unstack

```
import pandas as pd

df = pd.DataFrame({
    '语文':[80,90],
    '数学':[60,75]
}, index=['张三','李四'])

# 宽表转长表
长表 = df.stack()
print("宽表转长表: ")
print(长表)
```

任务 17 透视表 pivot

```
import pandas as pd

df = pd.DataFrame({
    '日期': ['周一', '周一', '周二', '周二'],
    '商品': ['苹果', '香蕉', '苹果', '香蕉'],
    '销量': [10, 20, 15, 25]
})

# 生成透视表
透视表 = df.pivot(index='日期', columns='商品', values='销量')
print("商品销量透视表: ")
print(透视表)
```

任务 18 面元划分 数据分箱

```
import pandas as pd

年龄 = pd.Series([18, 22, 35, 45, 55, 60])
# 划分区间和标签
bins = [0, 18, 30, 50, 100]
labels = ['未成年', '青年', '中年', '老年']

# 自动分箱
分组结果 = pd.cut(年龄, bins=bins, labels=labels)
print("年龄分组结果: ")
print(分组结果)
```

任务 19 哑变量处理 + 分类统计

```
import pandas as pd

# 电影类型数据
df = pd.DataFrame({
    '电影名称': ['电影1', '电影2', '电影3'],
    '类型': ['喜剧/动作', '动作/爱情', '喜剧/爱情']
})

# 拆分类型+哑变量+统计
类型拆分 = df['类型'].str.split(' / ', expand=True).stack()
哑变量表 = pd.get_dummies(类型拆分, dtype=int)
类型统计 = 哑变量表.sum()

print("各电影类型数量统计: ")
print(类型统计)
```

任务 20 遍历 DataFrame 行数据

```
import pandas as pd

df = pd.DataFrame({'A': [1, 2, 3], 'B': [4, 5, 6]})

# 逐行遍历
for 索引, 行数据 in df.iterrows():
    print(f"行索引: {索引}")
    print(f"行内容: \n{行数据}\n")
```