时序数据处理算子开发

1 作业简述

基于提供的算子开发模板，实现时序数据处理算法并进行测试验证。

2 时序数据处理算法

这里提供3组待实现的时序数据处理算法：

1. 数据画像算法，共22个，详细说明见作业文件"数据画像"。
2. 数据质量算法、数据匹配算法、频域相关算法，共16个，详细说明见作业文件"数据质量"、"数据匹配"、"频域相关"。
3. 数据修复算法、异常检测算法、字符串处理算法、序列发现算法，共15个，详细说明见作业文件"数据修复"、"异常检测"、"字符串处理"、"序列发现"。

每个小组根据选题情况，**实现3组中的1组算法即可。**

3 算子开发说明

算法基类模板为FlokAlgorithmLocal.py。算子开发的示例文件为SelectTimeseries.py，实现的是在输入的时序数据表中选择指定的时间序列。算法逻辑实现编写于算法类的run()方法内。在SelectTimeseriesUT.py中对算法进行测试验证。

具体开发流程如下：

1. 在SelectTimeseries.py中，引入算子基类FlokAlgorithmLocal及需要用到的第三方库。

```
from FlokAlgorithmLocal import FlokAlgorithmLocal, FlokDataFrame
```

2. 定义算法类并添加run方法。

```
class SelectTimeseries(FlokAlgorithmLocal):
    def run(self, inputDataSets, params):
        input_data = inputDataSets.get(0)
        timeseries = params.get("timeseries", None)
        if timeseries:
            timeseries_list = timeseries.split(',')
            output_data = input_data[timeseries_list]
        else:
            output_data = input_data
        result = FlokDataFrame()
        result.addDF(output_data)
        return result
```

   - 接收数据及参数：

     输入数据和算子参数通过函数参数的形式传递，在run方法中，inputDataSets为接收到的数据，params为封装完的参数。

     ```
     def run(self, inputDataSets, params):
     ```

   - 获取数据和参数：

通过get方法获取输入数据，get(0)获取第一个输入端口的数据，get(1)获取第二个输入端口的数据……（在本次作业中输入均为1张表，仅需使用get(0)即可），参数通过params中对应key的值来获取。

```
input_data = inputDataSets.get(0)
timeseries = params.get("timeseries", None)
```

- 编写算法逻辑：

  在将需要的数据和参数导入之后，编写算法逻辑，实现所需功能即可。

- 输出运算结果：

  在算法类中，需要将运算结果返回，以便进行之后的计算分析。通过调用FlokDataFrame来实现结果返回，具体用法如下。

```
result = FlokDataFrame()
result.addDF(output_data)
return result
```

  result为自己定义的返回值，是FlokDataFrame类的实例对象，调用addDF方法来输出结果。output_data为运算结果，数据类型为pandas.DataFrame。

3. 单元测试

在算法类定义完成之后，在SelectTimeseriesUT.py中编写并进行单元测试。所有算子的单元测试应集中到UTCollection.py中，用于一次性测试验证所有算子。

4. 运行示例

运行SelectTimeseries.py，输入的时序数据文件是test_in.csv，其中数据如下所示：

Time	root.test.d1.s1	root.test.d2.s2	root.test.d3.s3
2022-01-01 00:00:00	1.1	2.2	3.3
2022-01-01 00:00:01	2.1	3.2	4.3
2022-01-01 00:00:02	3.1	4.2	5.3
2022-01-01 00:00:03	4.1	5.2	6.3
2022-01-01 00:00:04	5.1	6.2	7.3

执行成功后会生成输出文件test_out_1.csv和test_out_2.csv，test_out_2.csv中数据如下所示：

Time	root.test.d2.s2
2022-01-01 00:00:00	2.2
2022-01-01 00:00:01	3.2
2022-01-01 00:00:02	4.2
2022-01-01 00:00:03	5.2
2022-01-01 00:00:04	6.2

4 评分说明

- 基础要求（100分，这里是除答辩展示外其他部分按百分制划分的占比）
    - 实现所选算法组的所有时序数据处理算法。对实现的每一个算法，提供完善的单元测试和正确性验证，应充分考虑和覆盖所有情况。（80分）
    - 代码风格应保持良好，并提供关键、必要的注释说明。（5分）
    - 撰写项目文档，文档应包括每个算子的核心算法逻辑说明，以及每个算子的测试样例简介。如涉及到第三方代码或文档的使用或引用，需在项目文档中注明。小组分工也请附在项目文档中。（15分）

5 作业提交格式

小组提交的作业文件夹需包含下面3个目录：

- src：代码目录，目录下包括所有实现的算子及测试文件。
- data：数据文件目录，目录下包括所有算子测试涉及到的数据文件。
- doc：项目文档目录，目录下包括pdf格式的项目文档。

其他目录可根据需要自行添加。提交时请将作业文件夹压缩后提交压缩包到网络学堂。

6 其他说明

- 部分算法的说明中可能涉及到文件内的其他算法，或其他文件内的算法，必要时可查阅对应算法。
- ”算子“和”算法“概念类似，均可理解为实现的功能函数，这里不加以区分，可视为同一概念。