# 配置说明

20241101 for windows

将shell文件改为python文件方便直接使用

## 前言

requirements的版本太老了，wheel编译会失败，换用最新的版本就行。python==3.10.15

### 主目录~设置

运行代码

import os;print(f'{os.path.expanduser("~")=}')

打印出来的就是主目录

现在我们有几个文件夹在主目录~下并列，后者是git clone我fork后的版本得到的

├── dataset  
  
├── MATP-with-HEAT  
  
│   ├── INTERPRET\_challenge\_regular\_generalizability\_track  
  
│   ├── ...

**以下都以dataset/Multi-v1为示例，其他同理**

## 预处理

**INTERPRET\_challenge\_regular\_generalizability\_track** 目标：重新按frame\_id排序得到sorted文件夹

1. case\_id处理说明

    本次处理中case\_id是不规范的列，综合考虑如下因素：

    - INTERPRET\_challenge\_regular\_generalizability\_track代码中不允许包含case\_id列

    - MATP-with-HEAT并没有提及case\_id列

    - MATP-with-HEAT数据载入中的所有samples，是指所有scenario\_names(如DR\_CHN\_Merging\_ZS2)文件夹下的所有csv，详见IT\_ALL\_MTP\_dataset函数定义

    因此决定，将每个csv按照case\_id拆分成若干个csv

1. 重构文件夹

    在myutils/data\_rebuilt\_folder.sh开头填写要运行的数据集地址DATASET\_DIR，然后运行

    ```cmd

    sh data\_rebuilt\_folder.sh

    ```

    输出符合input样式的recorded\_trackfiles，不动这个位置

1. case\_id处理

    在myutils/config.py填写你要处理的某个数据集DIR地址，然后运行

    ```cmd

    python -u “myutils/split\_case.py”

    ```

    如果要运行单个scenario或者全部scenario，改动split\_case.py变量target\_case\_name

    输出拆分case后的文件的文件在~/recorded\_trackfiles

1. sorted

    ```cmd

    python -u “myutils/segment\_data.py”

    ```

    采用default指令的默认设置为block\_len=40，gap\_len=20，argv\_len=0

    输出sorted在~/recorded\_trackfiles对应scenario下