流程

1 训练集数据准备

2 开始训练

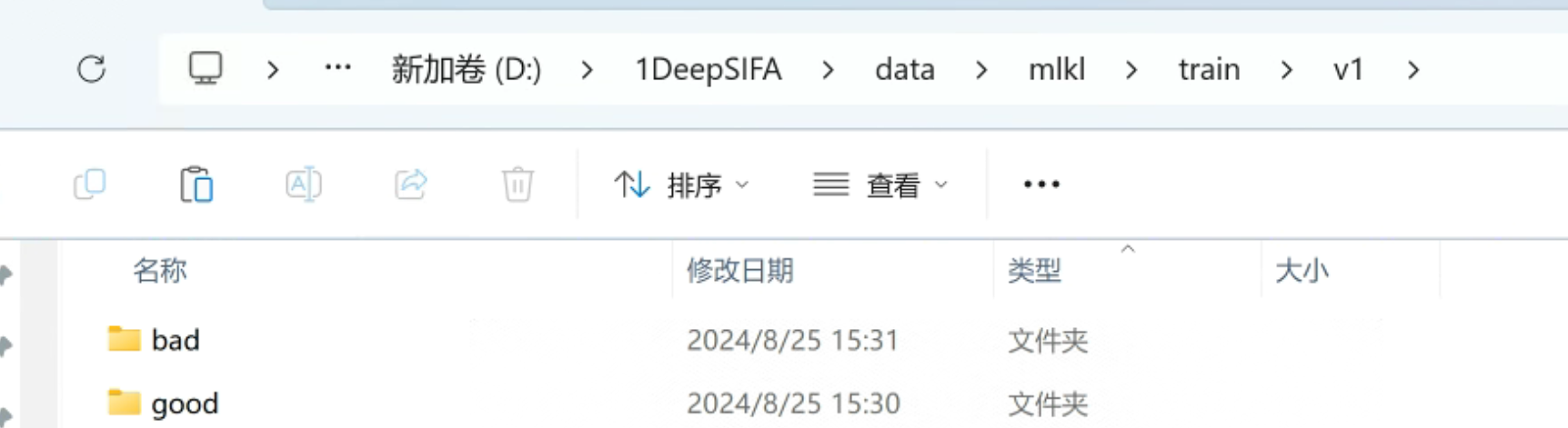
3 测试集数据准备

4 验证测试集

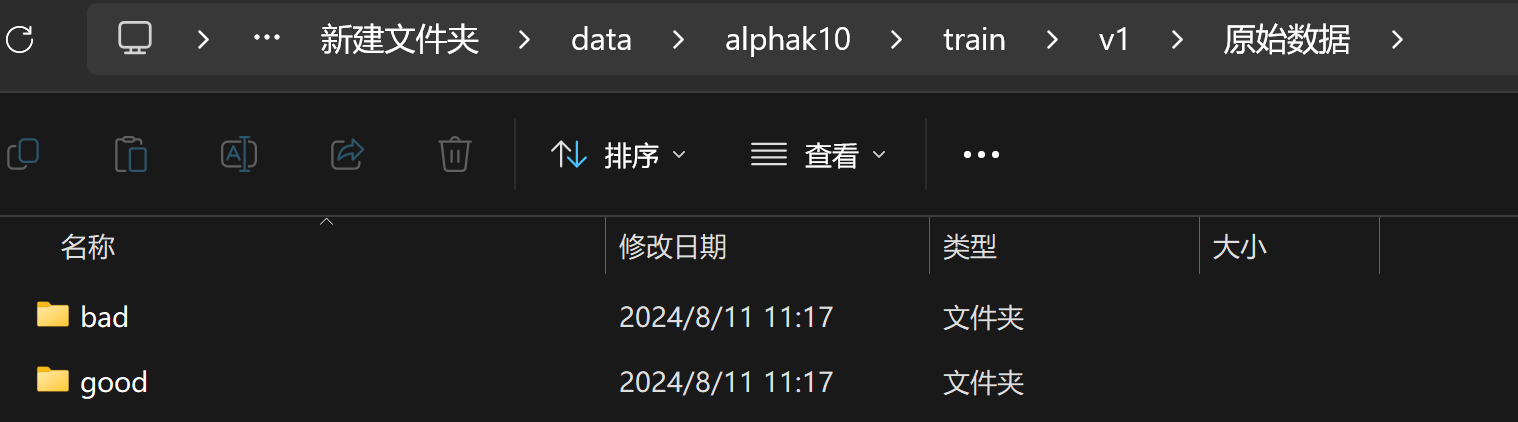
**1训练集数据准备**

1创建目录结构如下

mlkl

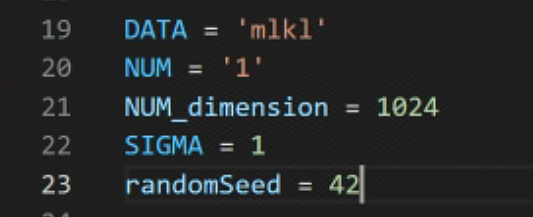


alphak10



2 把好曲线放入good文件夹，把坏曲线放入bad文件夹

3 执行脚本文件 生成训练集数据和csv.py



|  |  |  |
| --- | --- | --- |
| 超参数 |  |  |
| DATA | 目前支持2种数据  mlkl  alphak10 | 训练集是mlkl就写入mlkl  训练集是alphak10就写入alphak10 |
| NUM | 训练集第一版本就NUM=1，如果训练集有第二种版本就NUM=2，训练集还有第三种版本就NUM=3。 | 目前不需要修改 |
| NUM\_dimension | 是一条曲线有多少个数据点，原始数据有时候是500个点，有时候是1000个点，那就统一插值到1024个点。 | 目前不需要修改 |
| SIGMA | 高斯平滑的超参数设置，选为1是比较好的，可以平滑曲线去掉部分噪声。 | 目前不需要需改 |
| randomSeed | 5折交叉验证划分数据集的超参数，默认使用42 | 目前不需要需改 |

4训练集结构

执行 生成训练集数据和csv.py 脚本文件之后，在D:\1DeepSIFA\data\mlkl\train\v1目录文件下就会生成如下结构的文件，4个文件夹和一个csv文件。

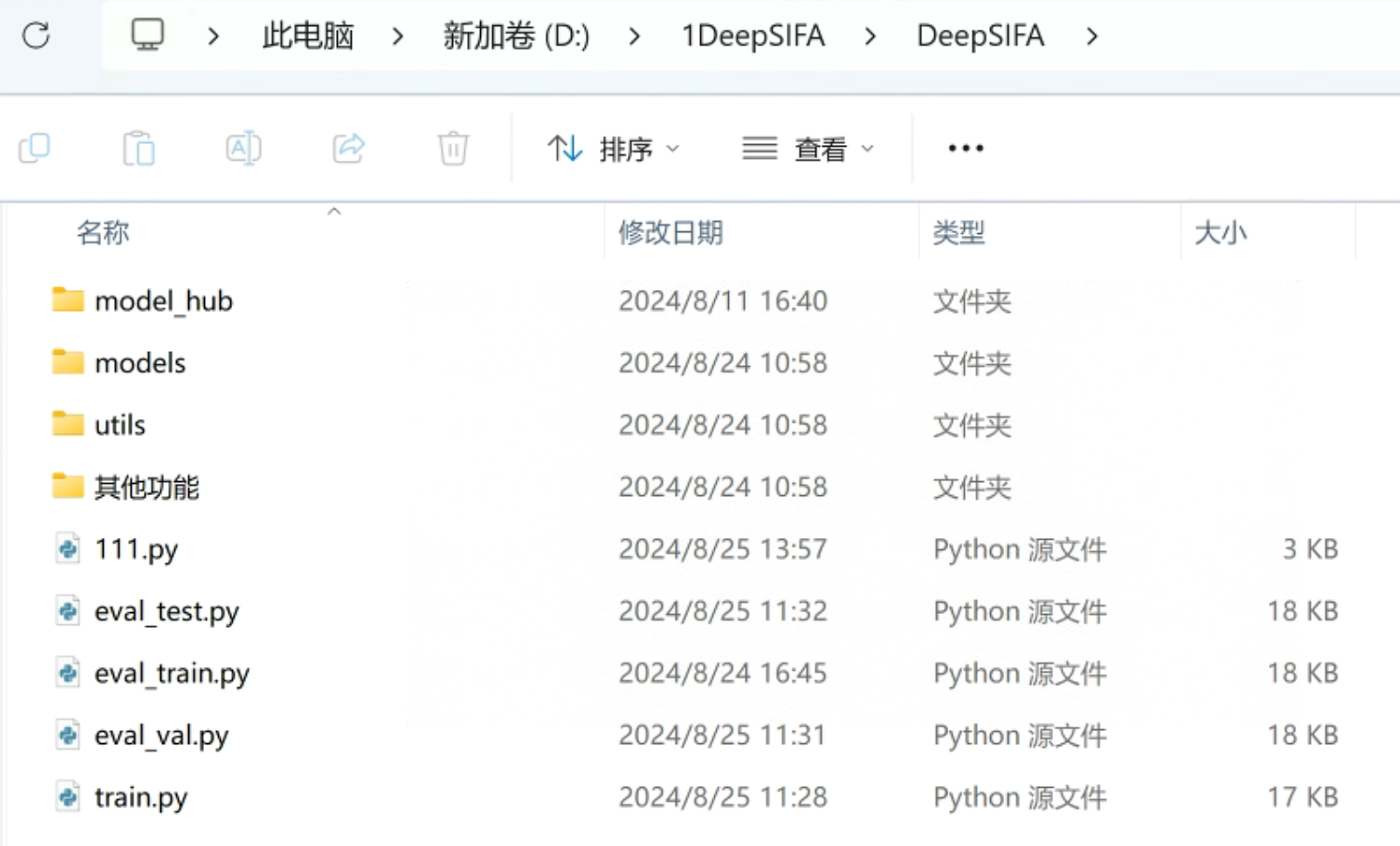


**2 开始训练**

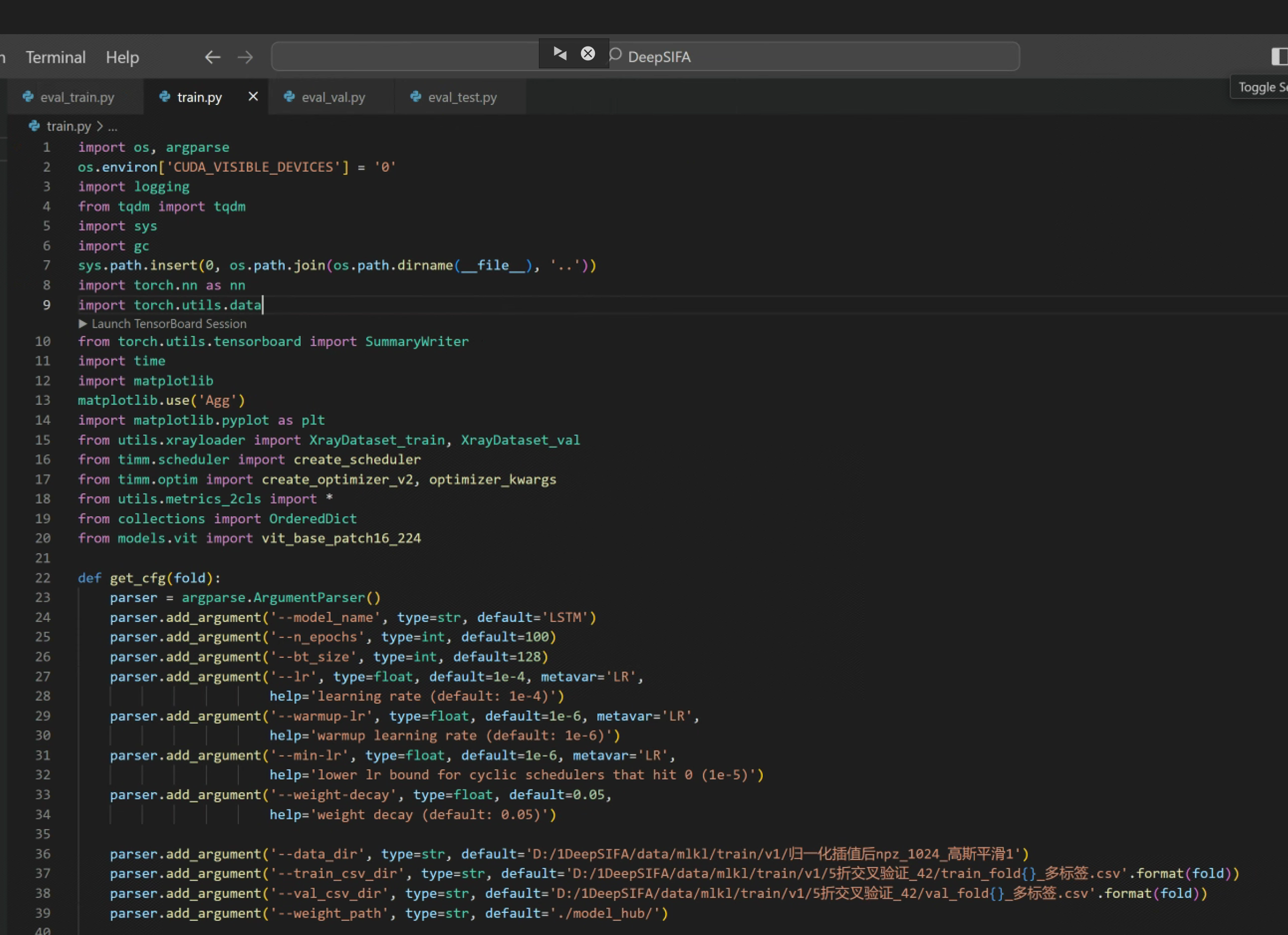
1 把DeepSIFA文件放到这个目录下



2 DeepSIFA包含文件如下



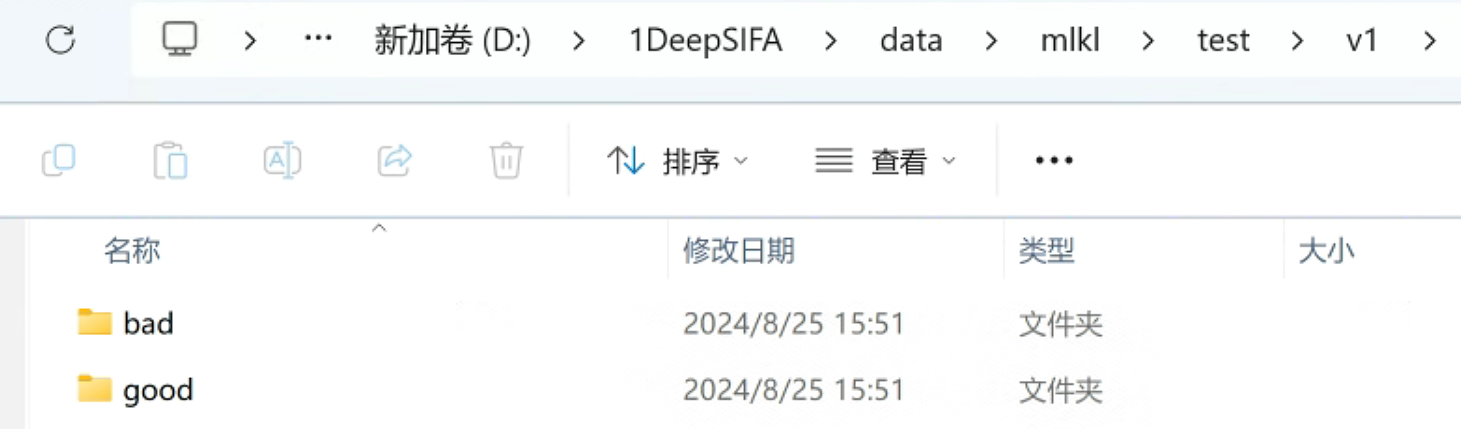
3 如果要训练的话，执行train.py文件就可以



|  |  |
| --- | --- |
| data\_dir设置为 | 'D:/1DeepSIFA/data/mlkl/train/v1/归一化插值后npz\_1024\_高斯平滑1' |
| train\_csv\_dir设置为 | 'D:/1DeepSIFA/data/mlkl/train/v1/5折交叉验证\_42/train\_fold{}\_多标签.csv'.format(fold) |
| val\_csv\_dir设置为 | 'D:/1DeepSIFA/data/mlkl/train/v1/5折交叉验证\_42/val\_fold{}\_多标签.csv'.format(fold) |

**3测试集数据准备**

1创建目录



2 把好曲线放入good文件夹，把坏曲线放入bad文件夹

3 执行脚本文件 生成测试集数据和csv.py



|  |  |  |
| --- | --- | --- |
| 超参数 |  |  |
| DATA | 目前支持2种数据  mlkl  alphak10 | 训练集是mlkl就写入mlkl  训练集是alphak10就写入alphak10 |
| NUM | 训练集第一版本就NUM=1，如果训练集有第二种版本就NUM=2，训练集还有第三种版本就NUM=3。 | 目前不需要修改 |
| NUM\_dimension | 是一条曲线有多少个数据点，原始数据有时候是500个点，有时候是1000个点，那就统一插值到1024个点。 | 目前不需要修改 |
| SIGMA | 高斯平滑的超参数设置，选为1是比较好的，可以平滑曲线去掉部分噪声。 | 目前不需要需改 |

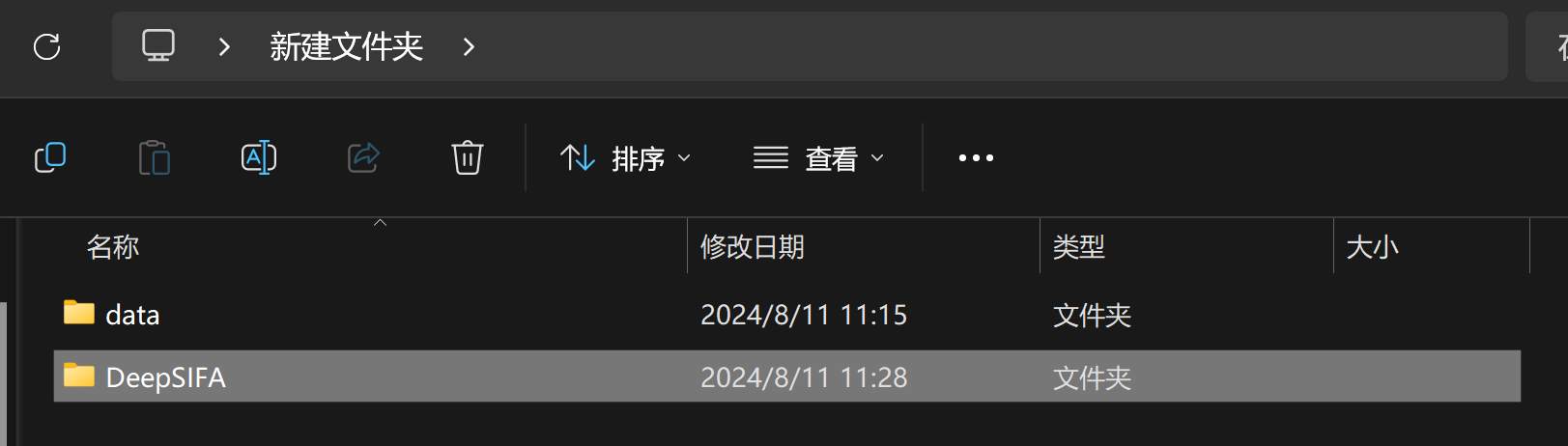
4测试集结构

执行 生成测试集数据和csv.py 脚本文件之后，在D:\1DeepSIFA\data\mlkl\test\v1目录文件下就会生成如下结构的文件，3个文件夹和一个csv文件。

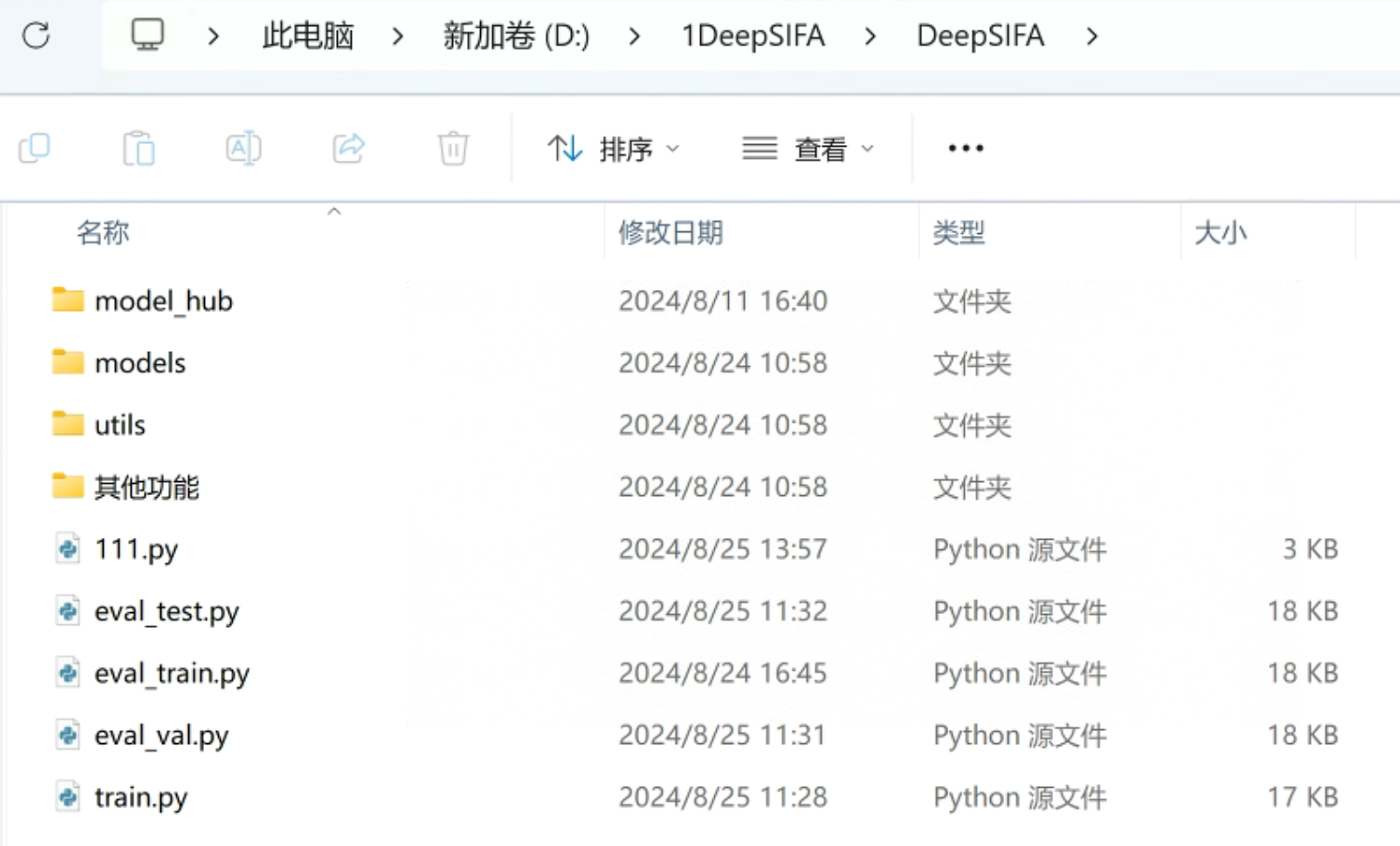


**4 验证测试集**

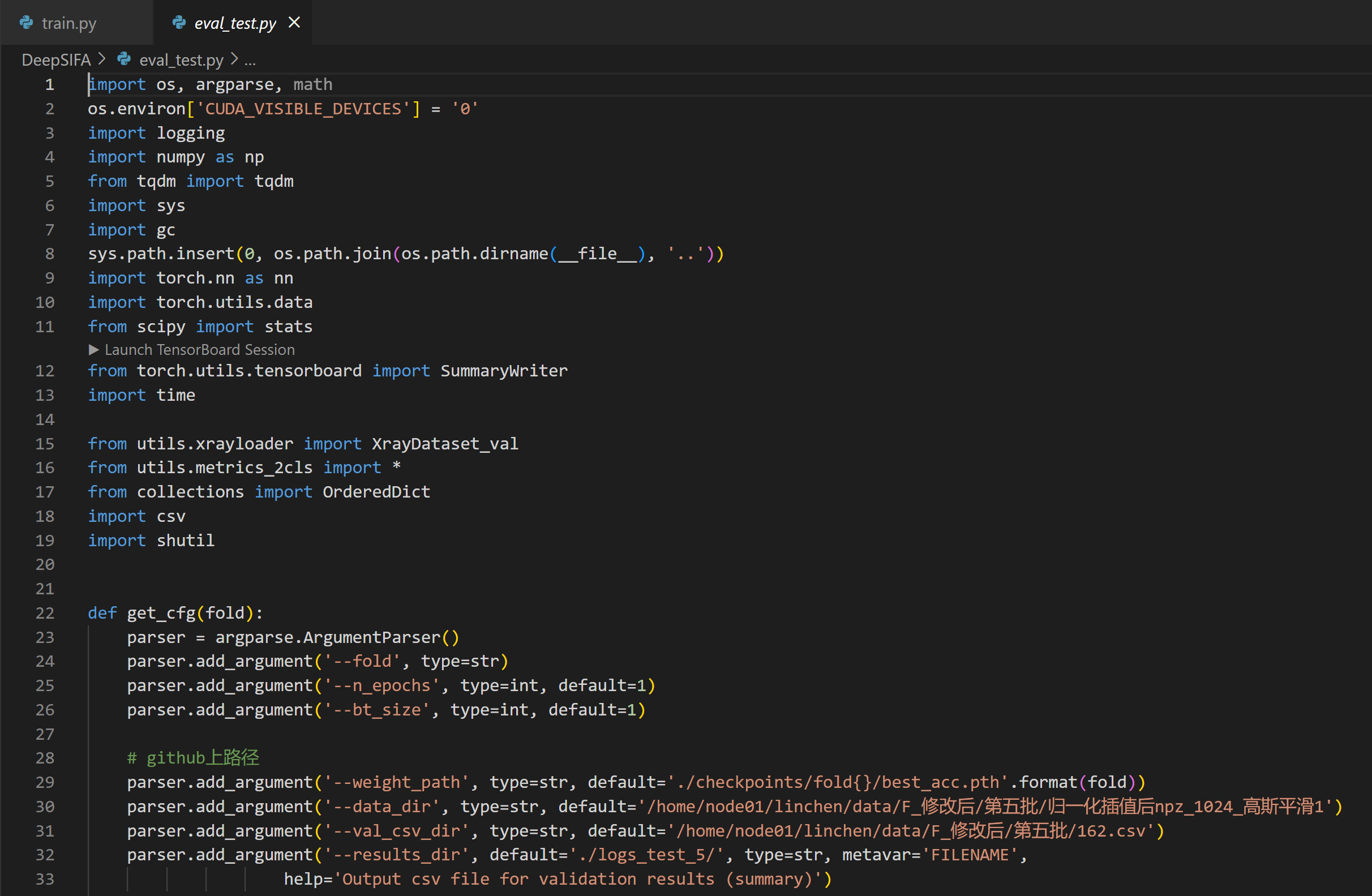
1 把DeepSIFA文件放到这个目录下



2 DeepSIFA包含文件如下



3 如果要验证测试集的话，执行eval\_test.py文件就可以



|  |  |
| --- | --- |
| data\_dir设置为 | 'D:/1DeepSIFA/data/mlkl/test/v1/归一化插值后npz\_1024\_高斯平滑1' |
| val\_csv\_dir设置为 | 'D:/1DeepSIFA/data/mlkl/test/v1/337.csv'.format(fold) |