最新代码使用流程：

1. 下载日线数据（下载并对日线数据进行复权处理，同时保留下复权因子文件）

**Get\_dealed\_daily\_data\_hldiff.py**

**所需文件：**

top10.csv

stock\_info\_0410.csv

**生成文件：**

日线数据 final\_save\_path+'daily\_stocks.csv'

和复权因子数据 final\_save\_path+'fq\_factor.csv'

1. 下载最近的分时数据（未经复权处理的分时数据）

**get\_minute\_data.py**

**生成文件：**

final\_save\_path = path + start\_date1 + '\_'+ end\_date1 + '.csv'

eg: E:/Stock/Data/Fenshi20200111/20200110\_20200111.csv

1. 拼接分时数据（将最新的分时数据和之前保存的分时数据拼接，并且进行复权操作）

**format\_fenshi.py**

**所需文件：**

上一个时段保存的分时原始数据(未经过复权处理的)：

Eg: D:/Stock/Data/190101\_191220/fenshi\_011220.csv

（之前保存的最新的分时文件在 压缩文件fs20200110中）

最新下载的最近时段的分时数据

Eg: E:/Stock/Data/Fenshi20200109/20191221\_20200109.csv

与日线数据对应的最新的复权因子数据，即1中生成的复权因子数据

Eg: D:/Stock/Data/20200110/fq\_factor.csv

**生成文件：**

到最新日期的未经过复权的历史分时数据

Eg: final\_save\_path + 'fenshi\_20200110.csv'

可以用于测试的经过复权处理的分时数据

Eg: final\_save\_path +'fenshi\_stocks.csv'

1. 进行测试

**search\_best\_para\_11\_19\_hl\_diff.py （此版本 增加前22个交易日的涨幅信息作为筛选）**

**代码： choose\_matrix1 = choose\_matrix1[choose\_matrix1[:,22] <= hl\_limit]**

**所需文件：**

分时数据 eg: fenshi\_path = 'D:/Stock/Data/20200110/ParaFindData/fenshi\_stocks.csv'

日线数据：eg: daily\_path = 'D:/Stock/Data/20200110/ParaFindData/daily\_stocks.csv'

(这两个数据在 文件夹中ParaFindData压缩文件内)

另外针对于 之前的测试文件 search\_para\_0818.py

数据文件在 para\_data\_0818压缩文件夹中