**1.框架结构**

**# big**

meta\_strategy\_framwork 共分为数据，信号，回测三部分，三部分均可以分开运行

**# 参数**

params.py 参数设定

**# 数据获取**

rqdata.py

copydata.py

可以单独使用，在copydata中的main函数中给定股票列表即可

**# 信号计算**

index\_24.py 原始指标计算

sig\_data.py 数据处理

sig\_fra.py 原始信号计算

sig\_meta\_strategy.py 策略信号计算

diff.py 分数阶差分

Cal\_fra.py 信号组合

get\_new\_expre.py 得到不同参数的表达式

quick\_sig.py 快速计算信号，等同于sig\_fra.py

get\_new\_expre.py 根据给定的信号公式得到新的参数公式列表

**# 回测**

strategy,py 策略回测

**# 使用方法**

在params中设定好参数，运行main.py脚本即可

**# 注意事项**

1.如果出现需要计算的指标没有在库里，需要检查sig\_data.py中的cal\_index\_data函数中是否计算了此指标

2.此框架可以生成做空信号，可以在sig\_meta\_strategy.py中调整，但是使用的回测框架无法对股票做空

2.策略回测步骤

在params中更改股票池，其中get\_code\_list函数是得到沪深300的股票池，再更改\_Expression即可，expression为两维的列表，第一个元素为买入信号，第二个元素为卖出信号，通过+\*来构建基础信号之间的逻辑关系

\_Expression具体构建方法如下：

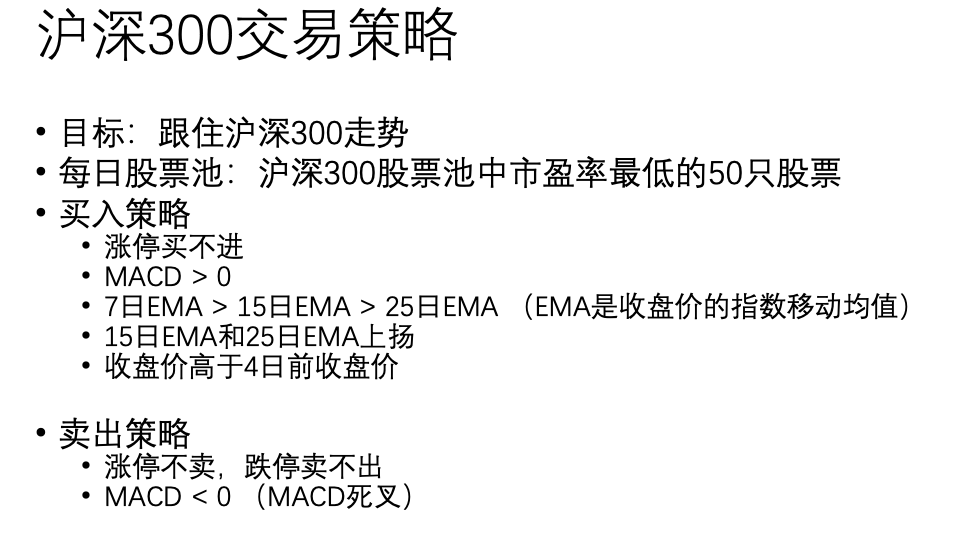
# 阈值型信号 指标名+#+阈值+#+方向(1为大于,0为小于)+&thre,如果为大盘信号，则在最后加上&HS

# 交叉型信号 指标1名+#+指标2名+#+方向(1为金叉,0为死叉)+&cross,如果为大盘信号，则在最后加上&HS

# 趋势型信号 指标1名+#+趋势延续天数+#+方向(1为上涨,0为下跌)+&trend,如果为大盘信号，则在最后加上&HS

# 比较型信号 指标1名+#+指标2名+#+方向(1为指标1大于指标2,0为指标1小于指标2)+&diff,如果为大盘信号，则在最后加上&HS

举例说明：



可以实现为['close\_EMA\_7#close\_EMA\_15#1&diff\*close\_EMA\_15#close\_EMA\_25#1&diff\*close\_EMA\_15#2#1&trend\*close\_EMA\_25#2#1&trend\*MACD#0#1&thre\*close#close\_shift\_4#1&diff\*K#40#1&thre&HS','MACD#0#0&thre+K#40#0&thre&HS']

3.结果储存，在main函数中有两个函数，一个是single，一个是optimal，分别是单个策略回测和优化策略回测，分别将结果储存在/result/single和/result/mul中，singel储存为singel.xlsx中，mul储存为expre和unit两个excel文件，每次运行single，就会在/result/single/single.xlsx中增加一行param中expression的结果