配对交易策略说明文档

配对交易作为一种广为流传的统计套利策略，提供了利用交易噪声与多空组合实现套利的可能性。本文将利用2019年的股票数据构建一个小资金量的配对交易策略，并在2020至今的时间范围内对策略进行回测与评估。

## 模型原理与概述

如果两个资产 ,其价格可如下构造平稳序列：

则资产具有协整性（cointegration),是均值回归的时间序列。配对交易的思路即为：当偏离均值时，采取做多/做空我们构建的多空组合；对于平稳分布而言， 必然会回归均值，在回归后反向做空/做多多空组合，即可实现一次配对交易获得套利。在这个模型中，如何检测的平稳性并求得均值与交易阈值，显然是实现交易的关键所在。

对于平稳分布，很自然的想到ARMA模型来描述。

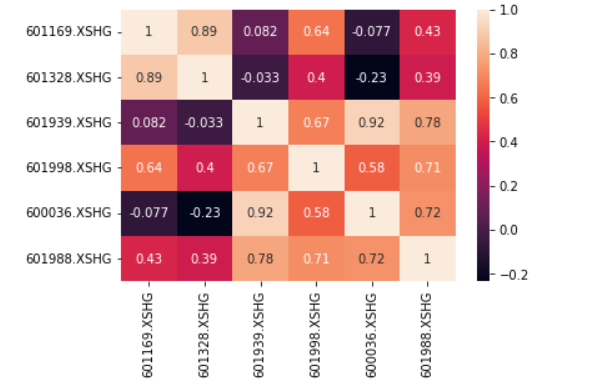
## 数据获取与评估

选取银行行业作为选股范围。

首先对得到的数据进行预处理,并计算其相关系数与，大体判断一下是否能够用于pair trading。

取19年数据为样本集，用以构建并优化策略，20年及以后的数据为回测集，用于评价策略。计算不同股票的相关系数。注意：并不是相关系数大就代表协整效应好，但相关系数小一定协整效应不好。

从图中可以看出，在选择的数据集中，'601169.XSHG','601328.XSHG'的相关系数最大，即为北京银行与交通银行，因此选择这两支股票作为后续计算的pair。



## 平稳性序列构建与检验

使用线性回归模型构建平稳序列，并且利用ADF检验其平稳性。具体结果参见代码“策略构建.ipynb”中1.1-1.2部分。

发现以年为单位的时间周期内，平稳性很差，p值约为0.25左右，无法认为序列平稳。因此重新按月进行平稳性检验。发现多数月分均可通过平稳性检验。换言之，这两只股票在月内的协整效应比较好。因此每月调整一次策略，即可满足变量平稳的要求

逐月结果如下所示：

year:2019 month:1

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.8874745242031925

(-3.1722543066162983, 0.021647937189217308, 8, 5271, {'1%': -3.4315912231936934, '5%': -2.862088492639781, '10%': -2.567061962230308}, -52912.99475400876)

p value:0.021647937189217308

T test:-3.1722543066162983

--------------------------seperate line----------------------------------------------------

year:2019 month:2

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.9069888453932988

(-3.0569180402725475, 0.029910125420907194, 10, 3589, {'1%': -3.4321733444512503, '5%': -2.862345651385977, '10%': -2.5671988611505343}, -33125.44468700458)

p value:0.029910125420907194

T test:-3.0569180402725475

--------------------------seperate line--------------------------------------------------

year:2019 month:3

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.9310405907792927

(-2.8430491271172285, 0.05238274698965932, 12, 5027, {'1%': -3.431651500360408, '5%': -2.862115123102503, '10%': -2.5670761386080314}, -50420.15801757234)

p value:0.05238274698965932

T test:-2.8430491271172285

--------------------------seperate line--------------------------------------------------------------------

year:2019 month:4

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.6414760569653567

(-3.315156196588582, 0.014214436591350438, 8, 5031, {'1%': -3.431650465047088, '5%': -2.862114665705449, '10%': -2.5670758951179296}, -51786.33138148823)

p value:0.014214436591350438

T test:-3.315156196588582

--------------------------seperate line---------------------------------------------------------

year:2019 month:5

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.8234887407531308

(-4.471487702447411, 0.0002213304228504798, 11, 4788, {'1%': -3.4317165015273323, '5%': -2.862143840025196, '10%': -2.567091425788434}, -41519.57573960478)

p value:0.0002213304228504798

T test:-4.471487702447411

--------------------------seperate line---------------------------------------------------------

year:2019 month:6

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.49984055071200995

(-2.0122536267422135, 0.2812262907798632, 11, 4548, {'1%': -3.431788653186758, '5%': -2.862175715236243, '10%': -2.5671083943786552}, -47902.93838871996)

p value:0.2812262907798632

T test:-2.0122536267422135

--------------------------seperate line-----------------------------------------------------------------

year:2019 month:7

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.7304383199206783

(-3.7910941813634174, 0.0029997588272598695, 26, 5493, {'1%': -3.4315410355947376, '5%': -2.8620663193384694, '10%': -2.567050158634283}, -58092.61699510171)

p value:0.0029997588272598695

T test:-3.7910941813634174

--------------------------seperate line------------------------------------------------------------------

year:2019 month:8

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.9381872307915633

(-3.408690048820666, 0.010666171698606305, 15, 5264, {'1%': -3.431592874562386, '5%': -2.8620892222219396, '10%': -2.56706235061258}, -52230.07425112926)

p value:0.010666171698606305

T test:-3.408690048820666

--------------------------seperate line--------------------------------------------

year:2019 month:9

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.8666515523288292

(-4.159361868912012, 0.0007714068823725739, 13, 4786, {'1%': -3.431717072875221, '5%': -2.862144092438703, '10%': -2.567091560158835}, -47114.52920166147)

p value:0.0007714068823725739

T test:-4.159361868912012

--------------------------seperate line-------------------------------------------------------------

year:2019 month:10

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.7966620282204718

(-2.2774026082508754, 0.1793596774195727, 20, 4299, {'1%': -3.4318720304552173, '5%': -2.8622125487431282, '10%': -2.5671280026535865}, -43532.6410816992)

p value:0.1793596774195727

T test:-2.2774026082508754

--------------------------seperate line----------------------------------------------------------------

year:2019 month:11

beta [0.38686121]

alpha [[0.7660498]]

R\_square 0.8036826798176335

(-3.9648467279735065, 0.001606192957005152, 16, 5023, {'1%': -3.431652537323488, '5%': -2.8621155812282546, '10%': -2.5670763824860745}, -52213.964568836454)

p value:0.001606192957005152

T test:-3.9648467279735065

--------------------------seperate line--------------------------------------------------------------------

year:2019 month:12

beta [0.38686121]

alpha [[0.7660498]]

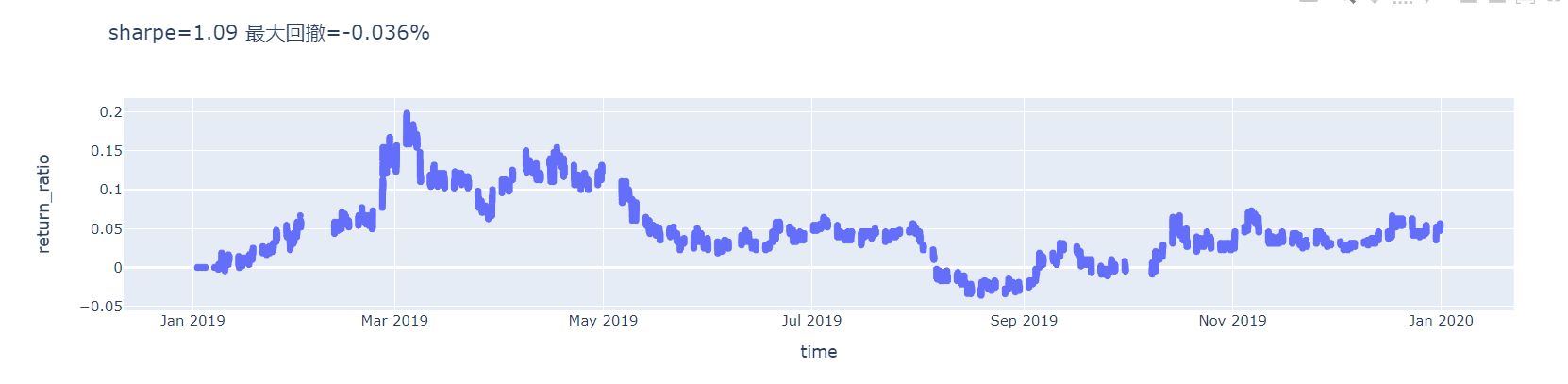
R\_square 0.8674750154617543

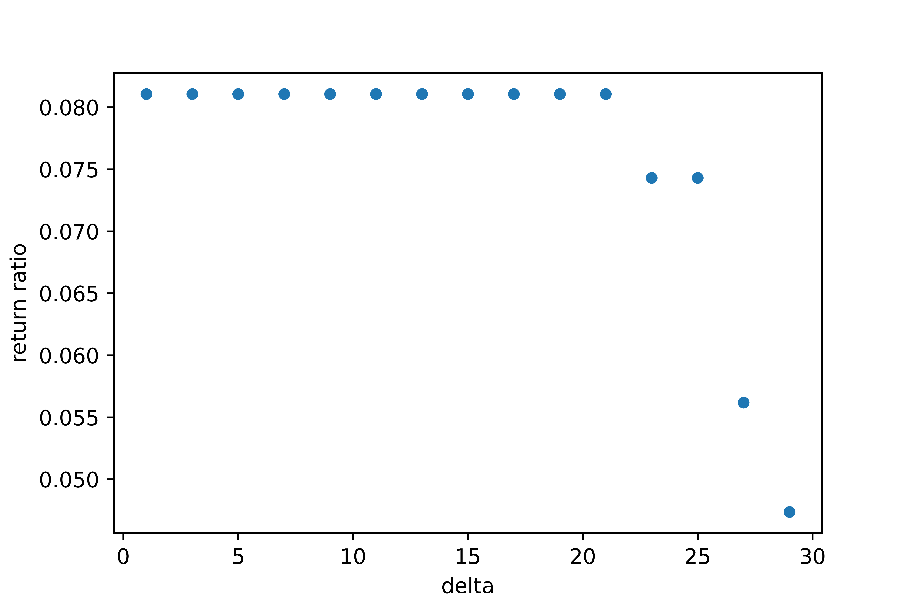
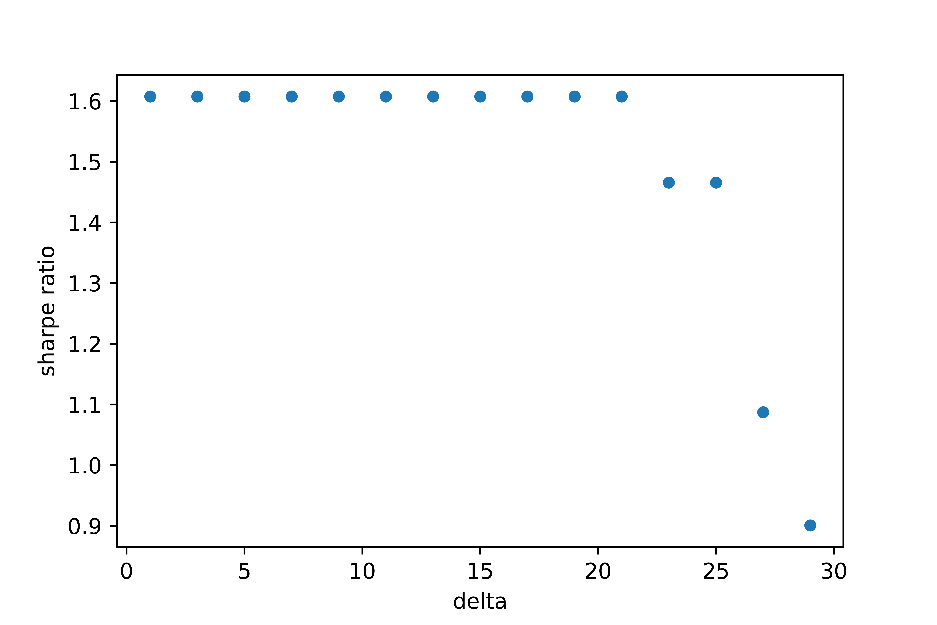
(-3.2660680565417577, 0.016465062774723267, 17, 5262, {'1%': -3.431593347189396, '5%': -2.862089431030616, '10%': -2.567062461768803}, -54532.13162873741)

p value:0.016465062774723267

T test:-3.2660680565417577

## 交易策略构建

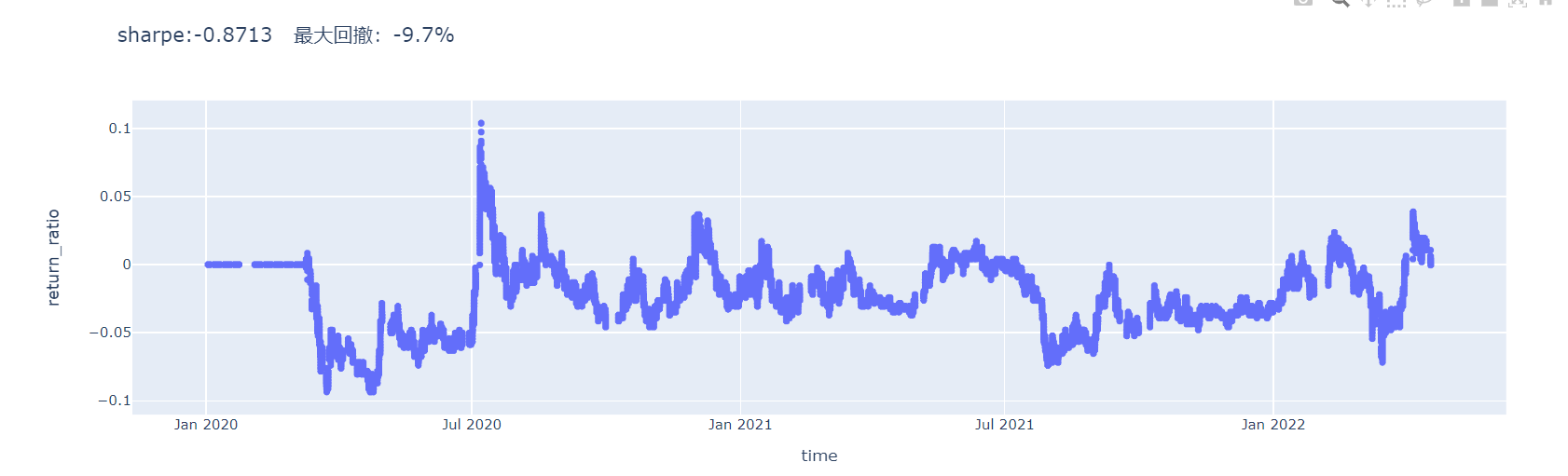
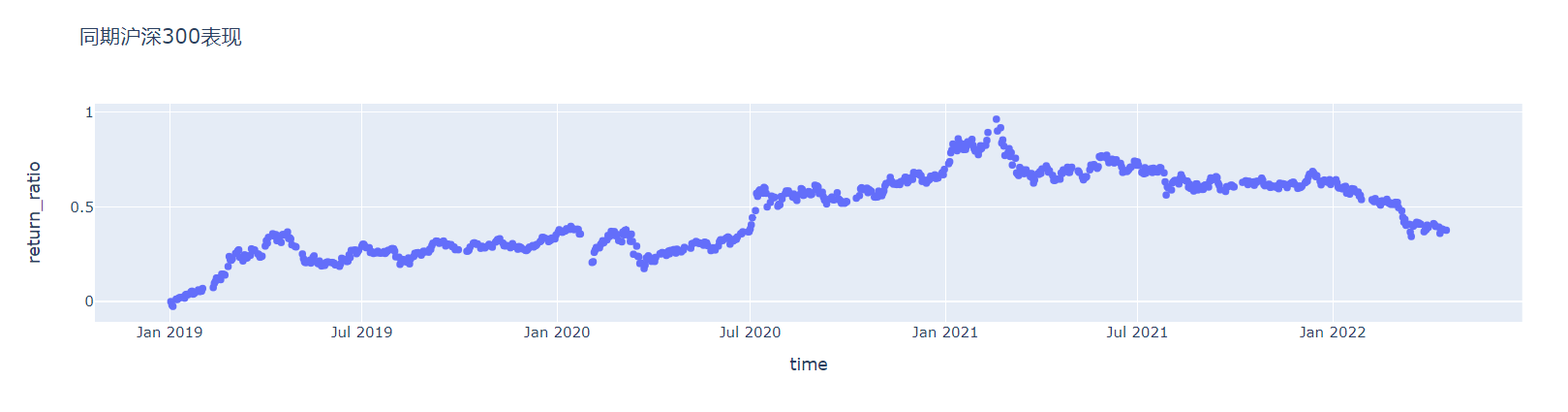
逐月进行策略调整，以上月均值作为平稳序列的均值。以作为策略的交易阈值，其中为上月的平稳序列方差的若干倍。对2019年数据进行回测，可以看到2019年年化收益率才0.3%左右，非常的低，将数据copy至excel，发现交易频次太小了，一年内交易次数不足10次。于是再次调低拟合时间，将月频改为每1000个数据点调整一次策略阈值。得到的结果如下图所示。

显然，改变策略调整频率极大的改善了收益率，也获得了不错的sharpe ratio。接下来是择优选择交易阈值。从0.1倍方差至30倍方差范围内逐一尝试交易阈值，并记录不同阈值下的收益率与sharpe ratio，结果如下图所示：

大约在23倍标准差为最优的收益率与sharpe ratio。选择23倍标准差即作为阈值。

## 策略回测

选取2020至今的数据作为回测范围，选取23倍标准差标准差作为阈值，每1000数据点进行策略调整，得到收益率曲线如下所示：

显然，该策略的回测表现并不好。

回测表现不好的原因：

1. 没有明显的止盈止损方案，在亏损大或者盈利较高的部分没有止盈止损方案。
2. 策略股票单一，一般仅有一只股票全仓或空仓的情况，这种情况使得收益受到单只股票价格波动很大。
3. 交易频率过低：分钟数据不足以支持足够多的交易频率，样本点过少也可能导致拟合结果差。

## 后续改进与想法

1.选股层面：可以衡量某一行业中所有股票的相关系数与$R^2$,选择二者均较大的多个组合作为Pair trading对象。（目前选股范围有点小，由于joinquant流量限制暂时无法实现）

2.周期层面：开始每年一周期修正策略，后来改为月频，但是好像还是没有找到合适的周期选择方法。一般来说策略制定的频率略大约交易频率的周期，所以可以假设这个策略大概是月内的频率

3.仓位方面：目前设定了仓位调仓阈值，但其实可以将’阈值‘改为’仓位梯度‘，当然仓位梯度的分布如何最优还需要解一个优化问题

4.成交问题：该模型并未考虑能否成交的问题，不过作为一个小资金量的低频策略，这一点是可以接受的。

5.阈值优化：目前对于ARMA问题的阈值优化有两种，一种是基于模拟，求一个梯度；另一种是求数学的解析解，ARMA问题总用数学最优化解，难度在于解微分方程（The Hamilton-Jacobi-Bellman (HJB) equation）。实话说，过往学术文献中的策略构建往往阈值选择比较随意。本文选择的方法表现并不好。

5.风控问题：应当引入合理的止盈止损机制来确保过大回撤的出现。