****

**单因子测试指导书**

**专 业 信息管理与信息系统**

**作 者 董晓宇**

**日 期 2022.06.25**

目录

[1 单因子测试的逻辑与步骤（此部分不包含代码） 1](#_Toc67066684)

[2 单因子测试代码实操 ---- 基于tool\_kit包 2](#_Toc67066687)

[2.1 tool\_kit各个文件介绍 2](#_Toc67066688)

[2.2 单因子测试的步骤 2](#_Toc67066689)

[2.3 完成自己的回测操作 3](#_Toc67066690)

[3 常用函数---大家自己去网上搜用法 5](#_Toc67066694)

# 1 单因子测试的逻辑与步骤（此部分不包含代码）

量化选股的方式之一是因子选股。因子可以理解成一种计算公式，按照公式带入已有的股票数据从而获得每只股票的因子值，再通过因子值的大小进行选股。

因子成千上万，质量也参差不齐，只有好的因子才能够让我们选出有盈利能力的股票，因此因子效果的测试十分重要。我们使用分组测试的方法测试因子效果。分组测试的逻辑如下：

我们以月频因子为例。月频因子对应月频调仓，即每个月换一次股票，并且下一个月一直持有这些股票。月频因子通常会在每个月月末通过已有的数据得到每个股票的因子值，在每个月末我们按照因子值的大小从小到大将所有的股票排序，排序后将股票平均分成10组，下个月每组始终持有这些股票不动，由于我们是在历史数据上进行回测，因此我们可以分别计算这十个组的收益以及风险。

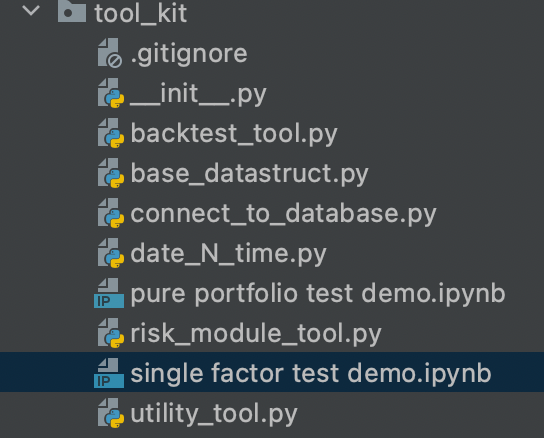
如果因子是好因子，那么第一组至第十组的效果将递增或递减。（逻辑：因子值与股票收益率呈正相关或负相关关系）。

请大家务必阅读《方正证券\_20180306\_“远山”量化选股系列（一）：规矩，方正单因子测试之评价体系》。这篇文章是我接触量化读的第一篇研报，详细的讲述了单因子选股的步骤与逻辑。大家只有在理解了逻辑才能读懂代码。

# 2 单因子测试代码实操 ---- 基于tool\_kit包

## 2.1 tool\_kit各个文件介绍

tool\_kit包是我在公司实习时使用的回测工具包。由于该工具包需要连接公司数据库，因此我们无法直接使用。但是该包中包含了单因子测试的所有功能函数，所以我们可以通过读csv的方式读入数据从而代替数据库读取数据操作，其他代码都可以直接copy工具包。



backtest\_tool.py文件是分组测试文件。

base\_datastruct.py大家使用不到，可以不用看。

connect\_to\_database.py是连接数据库文件。

date\_N\_time是日期转换文件

risk\_module\_tool.py用于数据预处理

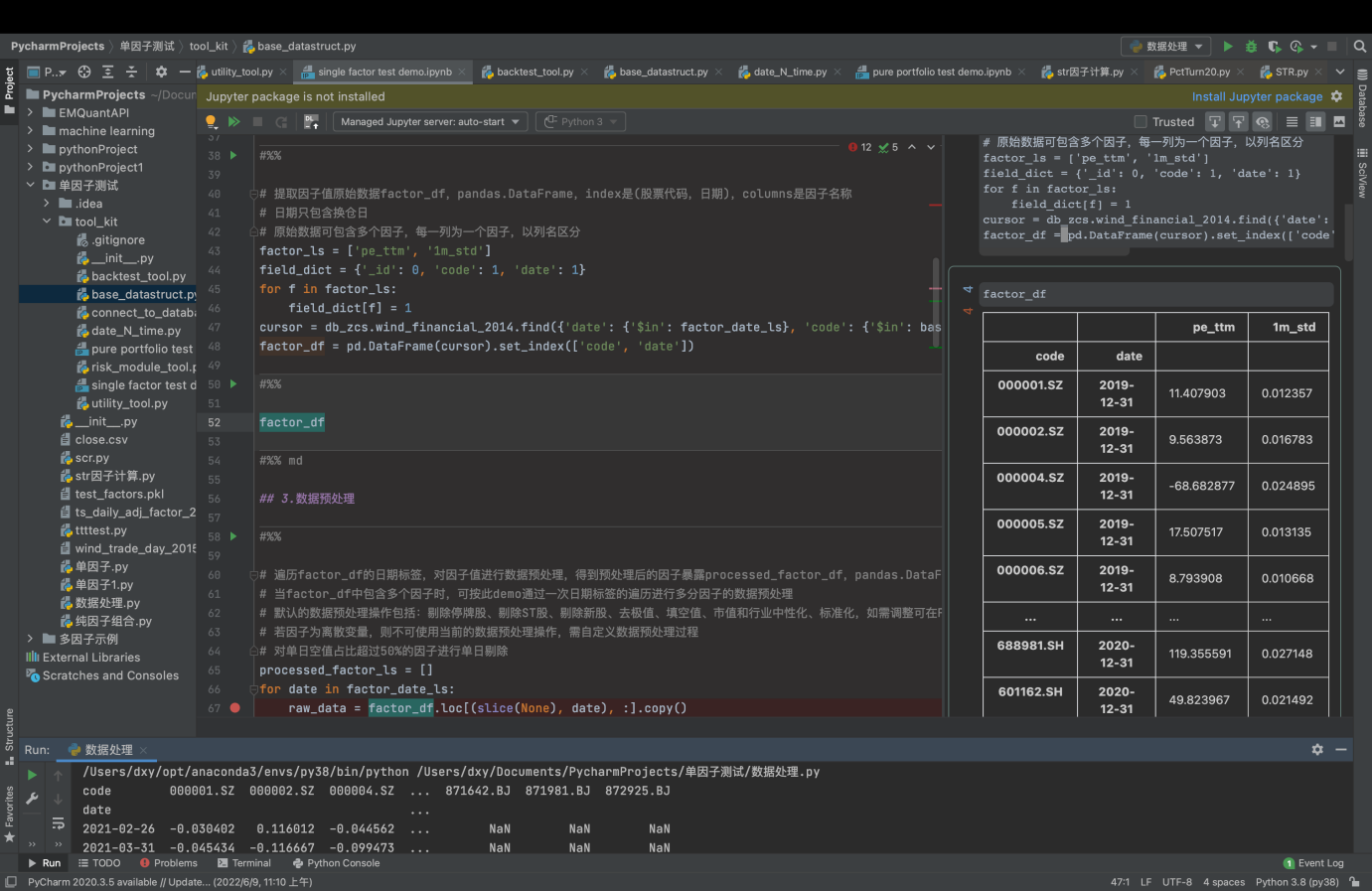
utility\_tool.py储存了各类函数，这些函数会在backtest\_too.py或risk\_module\_tool.py等等包中被引用。

pure portfolio test涉及纯因子组合测试，大家感兴趣的看

single factor test demo.ipynb 是使用tool\_kit包进行单因子测试的例子

## 2.2 单因子测试的步骤

大家详细阅读single\_factor\_test\_demo.ipynb这个单因子测试demo文件。这个文件左侧为代码右侧为运行结果。



（此处代码为从公司mongodb读取因子，实际操作时需要自己按照因子构建逻辑书写代码。factor\_df为读取后的因子值Dataframe，大家从##3.数据预处理开始看）

从3##开始的操作包括数据处理、分组回测。这便是得到因子值之后单因子测试的全部过程。首先大家一定要理解《方正证券\_20180306\_“远山”量化选股系列（一）：规矩，方正单因子测试之评价体系》，之后读single\_factor\_test\_demo.ipynb的代码，这个文件在数据处理、分组测试时分别调用了risk\_module\_tool.py与backtest\_tool.py，大家一定要仔细阅读理解这些文件中的代码！

以上过程将会很考验大家的代码功底，并且有着较大的难度，但是如果大家成功坚持下读懂了tool\_kit包，那么你的代码能力与对量化的理解能力都会有着质的飞跃！！

## 2.3 完成自己的回测操作

在读懂了tool\_kit包后，便可以自己写代码进行单因子测试了。tool\_kit包中只有连接数据库的操作无法做到，因此我们只需要把从数据库读取数据的过程变成从csv中读取数据，之后再需要做什么操作就从tool\_kit中复制粘贴便可（仅有小部分代码需要修改。）我在这里以分组回测为例（数据预处理同理）：见分组测试demo.py。这个文件写的比较粗糙，构造名称时比较随意，大家自己写的时候好好给变量取名字呀哈哈哈。

首先第一步是构造因子。  
df = df['close'].unstack(level=1).T  
df1 = df.shift(20)  
df = (df - df1)/df

这里由于时间原因我随便造了个简单的因子，即（当日的收盘价-20日前的收盘价）/20日前的收盘价。起名叫做str因子。这是一个月频调仓的因子。

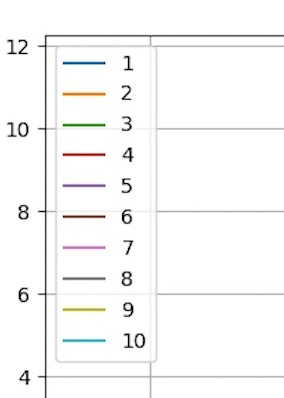
之后的分组测试代码全部从backtest\_tool中拷贝，大家可以参照我的代码看一下哪里进行了修改。需要注意的是日期问题一定要搞清楚！！！。除此之外画图函数的

ax1.plot(group\_value\_df['date'], group\_value\_df.iloc[:, :group].values)

这一行代码需要按照这个修改。且要删除这一行backtest\_tool画图函数的这一行代码

ax1.legend(labels=[str(i) for i in range(1, self.group+1)], loc=2)

### 这行代码的作用是显示图例（不删除会报错，但我也找不到报错原因哈哈哈，大家可以找一找原因）



# 3 常用函数---大家自己去网上搜用法

1.mongodb读取数据讲解

cursor = self.db.ts\_daily\_adj\_factor.find({'date': {'$gte': self.s\_date, '$lte': shift\_date(self.e\_date, self.freq, 'post')}}, {'\_id': 0, 'date': 1, 'code': 1, 'close': 1, 'adj\_factor': 1})

以上代码会经常见到，意思是从mongodb读取数据，其中{'date': {'$gte': self.s\_date, '$lte': shift\_date(self.e\_date, self.freq, 'post')}},为读取数据的起始终止日期， 'code': 1, 'close': 1, 'adj\_factor': 1})为读取哪些数据（此处为股票代码，收盘价，后复权因子）

2.shift函数

3.pd.concat()

4.stack,unstack函数（双重索引与矩阵的转化）

5.rank()

6.rolling函数

7.双重索引切片

其余的有空随时给大家更新

# 4 数据的获取与因子的获取

数据获取：

链接: https://pan.baidu.com/s/1H1YM3FEWuNI34NNCNMS47w 提取码: 0vww

--来自百度网盘超级会员v2的分享

这里是我从公司数据库导出的数据，tool\_kit包中读取mongodb的数据在这里都可以找到。zcs库表索引，factor\_tech数据字典这两个文件是对于csv中存取的内容。

因子获取：

因子可以通过研报读研报读到。研报可以从惠博找。我也给大家提供几篇供参考。