CICC
中金公司

基于Barra模型的业绩归因分析

崔诗颖

中金财富研究

2017年12月18日

基于Barra模型的业绩归因分析

目录

CICC 中金公司

一、流程框架

口 模型目标

- 已知某一具体产品的净值与持仓数据时，对其风格因子与行业因子做归因分析，直观反应出因子的收益贡献与风险贡献。

- 模型的实际效果为：初步实现了每日自动从wind导入行情与宏观数据、自动更新因子库、自动针对每个基金产品算其收益与风险归因，并在可视化界面中看其累计情况。

一、流程框架

□ 基于多因子模型（MFM）

$$\tilde{r_p} = \sum_{j=1}^{N} w_j \left(\sum_{k=1}^{K} x_{jk} \tilde{f_k} + \tilde{u_j} \right)$$

$\tilde{r_p}$：股票总收益

x_{jk}：因子载荷

f_k：单位头寸时因子收益率

w_j：股票j在组合中占的比重

$$\sigma_p = \sqrt{w^T (X F X^T + \Delta) w}$$

F：因子收益率协方差矩阵

一、流程框架

模型重要参数估计：f_k F

- 其中 f_k由已知个股/组合收益率序列及因子载荷，用回归方法得到。
- F为f_k的协方差矩阵，可用指数加权改进。

归因结果：

- $\sum_{j=1}^{N} w_j \, x_{jk} \widetilde{f_k}$为第k个因子的收益贡献

在求f_k时有两种处理方式：

- 时间序列回归法
- 横截面回归法

经检验，无法用时间序列回归，不能通过单位根检验，且时序存在Arch效应

一、流程框架

□ 横截面回归法

- 对于投资组合$\tilde{r_p}$，有n支股票，股票权重分配为w；k个因子，因子载荷为x_{jk}
- 对每天的n支股票，其股票收益率为Y，k个x_{jk}为X，回归出的f_k为当日每只股票对因子K的收益率（斜率），注意x_{jk}为n天前
- 当日收益贡献为 $\sum_{j=1}^{n} x_{jk} * f_k$
- 可以辅助基金经理量化选股，规避风险
- 清晰看到每只股票的因子收益情况
- 不用rolling，直接每天取当日算，且股票有3000支，回归数据集大

一、流程框架

a) 从wind提供的量化接口中导入沪深所有股票的所需指标，方便后续更新因子库使用（所有数据采用空值向前取）

【收盘价、开盘价、最高价、最低价、市值、成交量、收益率、换手率、一致预测净利润2年复合增长率、净资产收益率TTM、市盈率PE、Beta50、同比增长率、自由流通股本、ATR波幅、负债合计、长期负债占比、资产总计、实收资本（股本）、所属行业wind代码】存入到stock.pickle（三维表）模块中

【SZ50,HS300,ZZ500,ZZ800,ZZ1000】的收益率
存入到index.csv中

wind_get_data.py

```
# -*- coding: utf-8 -*-
"""
Created on Mon Dec 11 11:44:29 2017

@author: Rebecca Cui
"""

import ...
#----------------------------根据需要修改下面的参数----------------------------
market_cols = ['close','open','high','low','mkt_cap_ard','volume','pct_chg',
               'turn','west_netprofit_CAGR','roe_ttm2','prt_topnstocktonav',
               'beta_100w','yoyeps_basic','free_float_shares','ATR',
               'tot_non_cur_liab','tltoebitda','tot_assets','tangibleasset',
               'industry_gicscode']
index_codes = ['000016.SH','000905.SH','000906.SH','000852.SH','000300.SH']
beginDate = '2015-06-01'
path = r'C:\\DELL\\internship\\CICC\\Barra\\raw data' # 结果储存路径
#------------------------------------------------------------------------------
today = datetime.date.today().strftime('%Y-%m-%d')
endDate = (datetime.date.today()-datetime.timedelta(days=1)).strftime('%Y-%m-%d')
w.start()
def GetMarketInfo(code,cols):...
def Align(code,df):...
def Concat(codes,cols):...
# 通过wset取当前沪深300成份股
stockSector = w.wset("sectorconstituent","date="+today+";sectorid=1000000090000000
dates,codes,names = stockSector.Data
```

CICC 中金公司

一、流程框架

b) 建立全股票池的因子库，目前共纳入因子31个，因子文件存入Database中，在这一步不做标准化、去极值等任何预处理，同时更新中信行业因子

```
9
10      #property copyright "Shiying Cui"
11      #property version   "5.0"
12      #property private!!!
13      ##########################1 收盘 2 开盘 3 最高 4 最低
14      ######9 一致预测净利润2年复合增长率  10  净资产收益
15      #######13  同比增长率   14  自由流通股本  15 ATR波幅
16
17      class style_fac(object):
18          def __init__(self,data,index):
19              self.data = data
20              self.index = index
21          ##存各指数价格，上证指数为0，沪深300为1，中证5
22
23      ###1
24          def bigsize(self):...
36      ###2
37          def medsize(self):...
48      ###3
49          def high_low(self):...
63      ###4
64          def retv(self):...
75      ###5
76          def turn(self):...
87      ###6
88          def wgt_rt(self):...
102     ###7
103         def halpha(self):...
```

```
129     ###8
130         def vol(self):...
141     ###9
142         def CAGR(self):...
145     ###10
146         def ROE(self):...
149     ###11
150         def topstock(self):...
153     ###12
154         def increase(self):...
157     ###13
158         def EY(self):...
169     ###14
170         def MLEV(self):...
181     ###15
182         def DTOA(self):...
193     ###16
194         def BLEV(self):...
206     ###17
207         def B300(self):...
235     ###18
236         def B500(self):...
264     ###19
265         def B800(self):...
293     ###20
294         def B1000(self):...
322     ###21
323         def R300(self):...
```

```
352     ###22
353         def R500(self):...
382     ###23
383         def R800(self):...
412     ###24
413         def R1000(self):...
442     ###25
443         def ASI(self):...
474     ###26
475         def DDI(self):...
502     ###27
503         def Hurst(self):...
520     ###28
521     ##########P1,P2默认值为3###############
522         def KDJ(self,P1,P2):...
544     ###29
545         def MFI(self):...
577     ###30
578         def Ulcer(self):...
600     ###31
601         def BR(self):...
624     ###industry
625         def industry(self):...
645     ###将各股收益率存入表中
646         def netrt(self):
647             temp = self.data.iloc[:, :, 6]
648             temp.to_csv(r"C:\\Dell\\internship\
649
```

CICC 中金公司

一、流程框架

c) 处理每日基金底层持仓数据

修正错误：
观察到基金底层持仓表中股票组合（1102）占净值百分比，因此需要从基金中提取股票组合的权重及股票组合的累计净值。

此前对于基金收益率直接拿基金累计净值计算得来【错】

表中：
1102代表股票投资
SH 代码为 0101
估值增值为0199 0701
SZ 代码为 3101，4101
估值增值为 3199，4199，3301

估值增值和成本股等市价为0，都未计入持仓中。

证券投资基金估值表

恒生电子___中金浦泰艾方量化金选1号私募投资基金___专用

估值日期：2017-03-07　　单位净值：1.005

科目代码	科目名称	市值占净值%	估值增值
102113	期货清算备付金	3.8369	
10211301	永安期货	3.8369	
1031	**存出保证金**	**0.6002**	
103106	券商保证金-中金公司	0.1719	
103113	期货交易存出保证金	0.4283	
10311322	永安期货	0.4283	
1102	**股票投资**	**22.8285**	**320,642.47**
110201	上交所A股	11.0385	108,701.89
11020101	上交所A股成本	11.0385	108,701.89
11020199	上交所A股估值增值	0.1162	
110231	深交所A股	10.0409	177,665.63
11023101	深交所A股成本	10.0409	177,665.63
11023199	深交所A股估值增值	0.1900	
110241	深交所A股_创业板	1.7491	34,274.95
11024101	深交所A股成本_创业板	1.7491	34,274.95
11024199	深交所A股估值增值_创业板	0.0367	
1105	**基金投资**	**64.3161**	**4,810.89**
110503	ETF基金	64.3161	4,810.89
11050301	ETF基金成本	64.3161	4,810.89
11050301511990	华宝添益	64.3161	4,810.89
11050399	ETF基金估值增值	0.0051	
1202	**买入返售金额资产**	**8.5555**	

CICC 中金公司

一、流程框架

c) 处理每日基金底层持仓数据

算法流程：

对每日底层持仓数据表，提取其基金名称，并通过字符串切割提取代码将其转换为wind代码

该基金所有持仓数量放在一张csv中，未出现的股票直接赋值为0

除此外，根据1102股票组合占净值百分比，算出纯股票组合的产品净值，在此基础上算出收益率，作为接下来回归的y

```
10    path = r'C:\DELL\internship\CICC\Barra\test'
11    file_list = os.listdir(path) # 指定文件夹中的所有文件
12    # 删除文件夹中无用的表
13
14    def Replace(s):
15        dic = {'0101':'.SH','01990':'.SH',
16               '3101':'.SZ','4101':'.SZ','31990':'.SZ','41990'
17        prefix = s[-6:]
18        suffix = s[4:8]
19        ss = prefix + dic[suffix]
20        return ss
21
22
23    def GetEquity(df):...
36
37    def GetValue(df):...
43
44    #ser = GetNet(df)
45    rt_equity = pd.DataFrame()
46    rt_cols = []
47    rt_vals = []
48    pt_equity = pd.DataFrame()
49    pt_cols = []
50    pt_vals = []
51
52    for file in file_list:...
71
72    rt_cal = pd.DataFrame(rt_vals,index=rt_cols,columns=['equity'
```

CICC 中金公司

一、流程框架

d) 收益归因分析

对所有参与因子做标准化与去极值处理（标准化保证均值为0）

对基本面因子取窗口为8天，技术面因子取窗口为2天

做不带截距项的加权最小二乘回归，加权方法参考于广发研报 （市值的加权）

不做参数优化，不设置in-out sample

输出结果为3个，分别是各因子收益贡献绝对值、各因子收益占比，各因子

```
1   import numpy as np
2   import pandas as pd
3   import datetime as dt
4   from datetime import timedelta
5   import statsmodels.api as sm
6   import scipy.stats.mstats as ssm
7
8   class regress(object):
9       def __init__(self,today,data,hold,tot_ret):...
16
17      def date_win(self, win1, win2):...
33
34      def pre(self,low_limit,up_limit):...
166
167     def industry(self):...
177
178     def factor_ret(self):...
332
```

参数：
Today为当前希望分析的日期；
Data为更新的行情库
Hold为处理好的增量持仓数据表
Tot_ret为处理好的股票部分收益率
Win1取2天，win2取8天
Low_limit和up_limit 取0.01，主要是刨去inf值，标准化后会变成 +、- inf

一、流程框架

e) 风险归因分析

对于风险协方差：

$$F_{kd}^{(d)} = cov(f_k, f_l) = \sum_{s=t-h}^{t} \tau^{t-s}(f_{ks} - f_{kmean})(f_{ls} - f_{lmean}) / \sum_{s=t-h}^{t} \tau^{t-s}$$

边际风险贡献FMCAR：

$$FMCAR = \frac{F * \mathrm{x}_{pA}}{\sigma_p}$$

参数：
ito为半衰期，意义为第t-ito/2天的权重为第t天的一半；
win为窗口，即时间序列长度

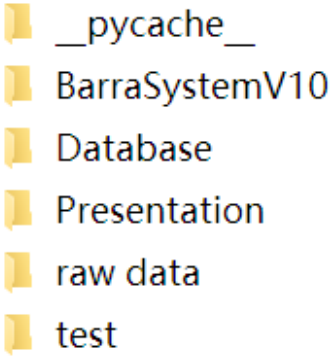
Win取40，ito取5

总风险贡献即为风险暴露乘边际风险贡献

输出结果为风险贡献和风险贡献占比

不做参数优化，不设置in-out sample

二、使用方法

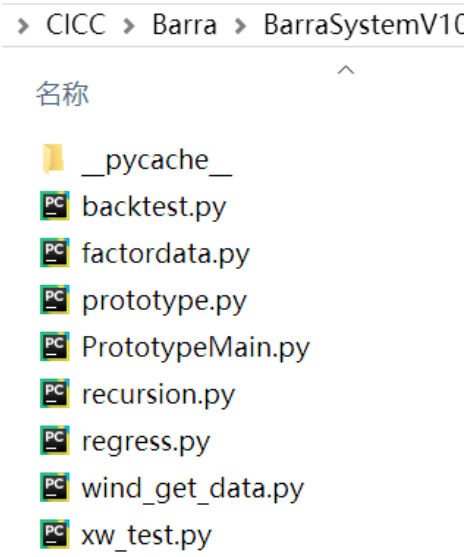


BarraSystemV10: python project 打包所有程序

Database: 风格与行业因子数据库

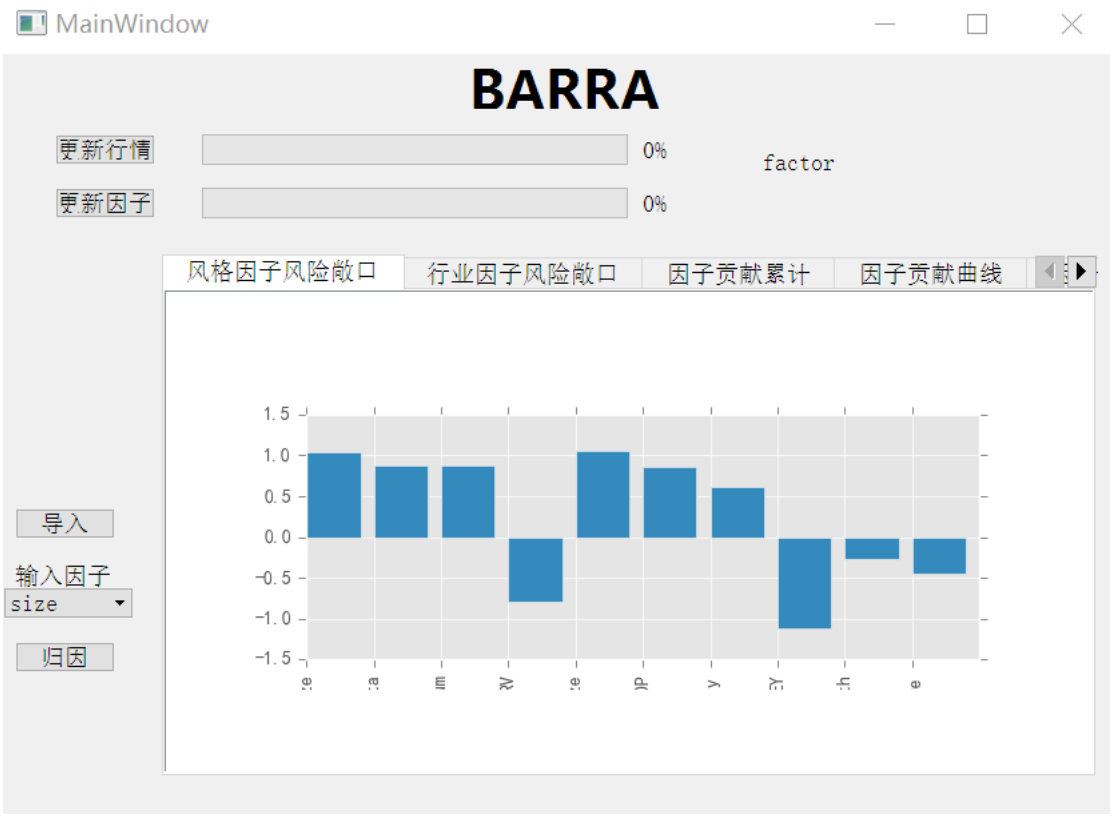
Raw data: 存放股票与股指行情数据、基金股票部分累计净值、收益率、基金持仓数据

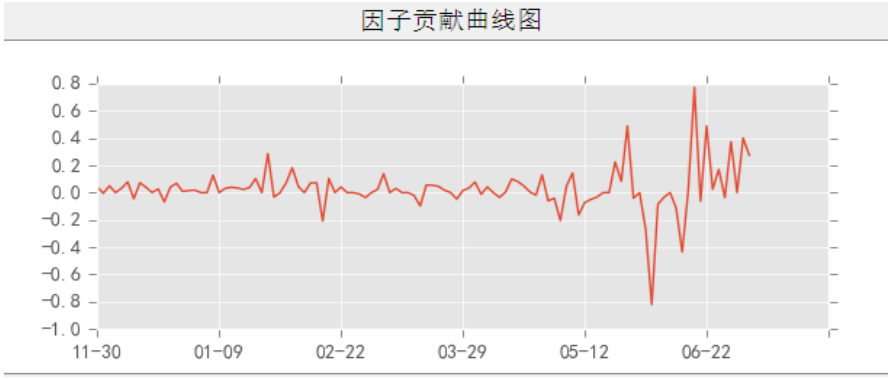
Test: 存放所有的基金底部持仓数据（日更）

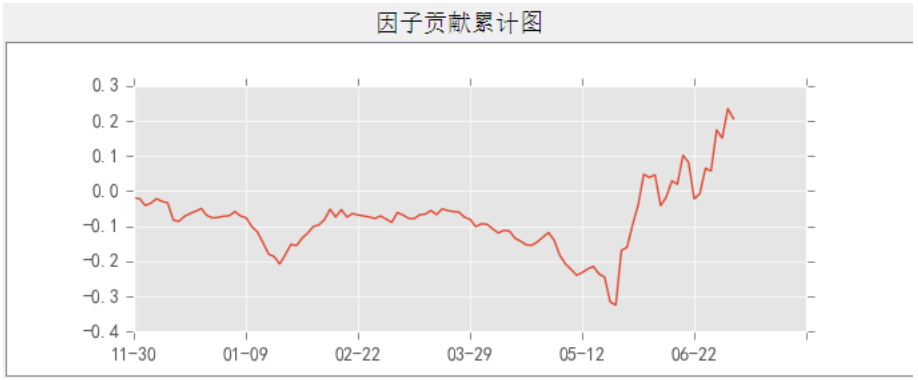


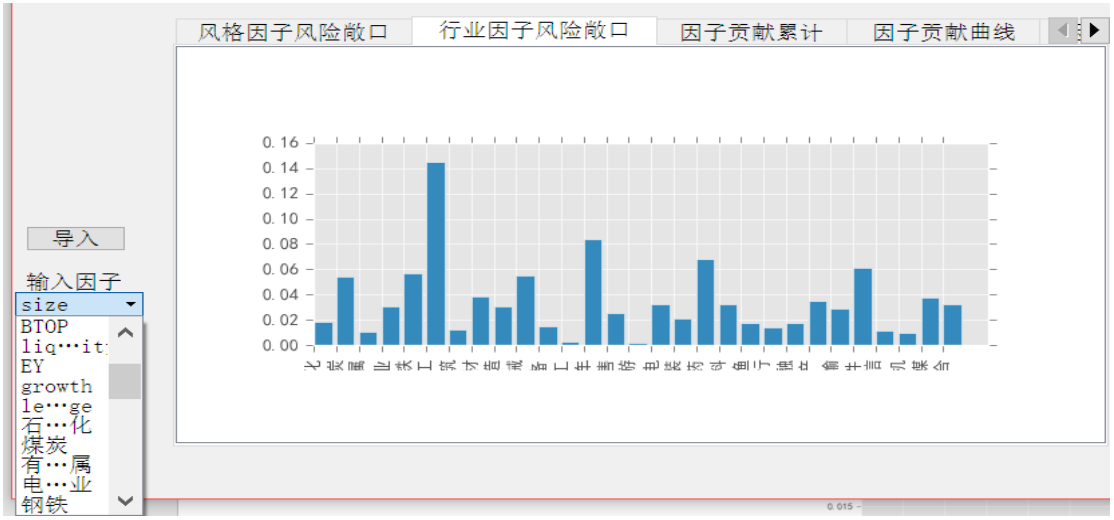
Backtest: 因子回测文件（定义具体函数类）
Factordata: 更新每日因子库的子类
Prototype: py GUI 框架参数与函数定义（父类）
PrototypeMain: 执行函数【每日运行该程序即可弹出图形界面，方便操作】
Recursion: 对因子进行回测，输出十几个指标，可每月运行一次，找出该月表现较好的因子进行归因
Regress: 两个类： 收益归因与风险归因
Wind_get_data: 从wind量化接口日更所需数据： stock和stock_index两张表
Xw_test: 处理每日基金持仓底表，更新各基金的hold(持仓表) 和 net_value(股票部分净值表)

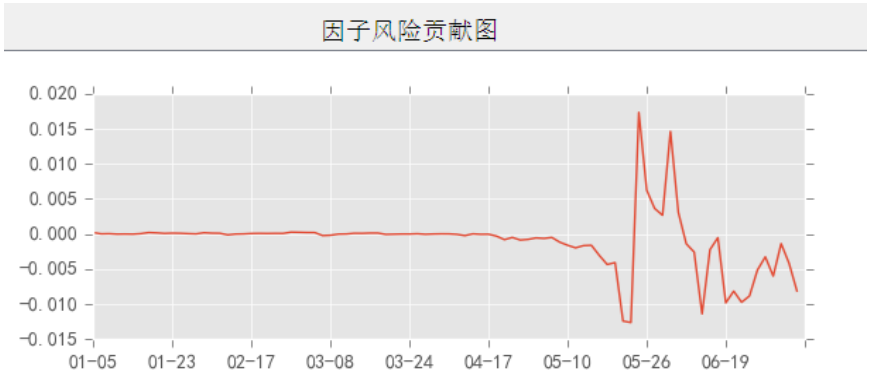
二、使用方法


MainWindow
BARRA
更新行情
0%
factor
更新因子
0%
风格因子风险敞口
行业因子风险敞口
因子贡献累计
因子贡献曲线
导入
输入因子
size
归因

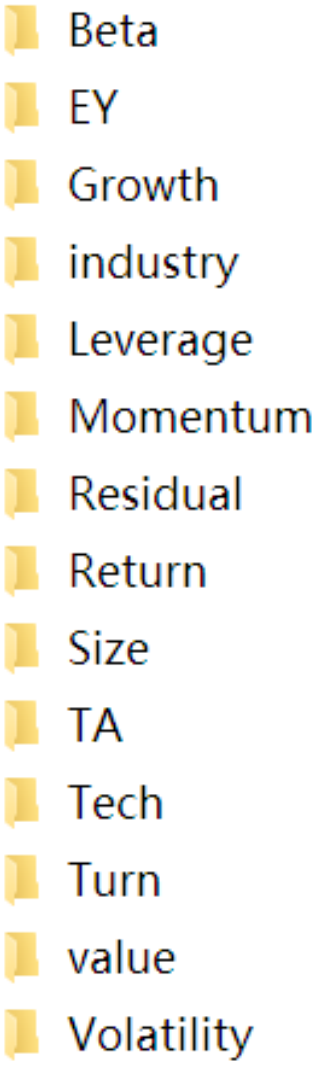

因子贡献曲线图
0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.8
-1.0
11-30
01-09
02-22
03-29
05-12
06-22


因子贡献累计图
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
-0.4
11-30
01-09
02-22
03-29
05-12
06-22


风格因子风险敞口
行业因子风险敞口
因子贡献累计
因子贡献曲线
导入
输入因子
size
BTOP
liq…it
EY
growth
le…ge
石…化
煤炭
有…属
电…业
钢铁


因子风险贡献图
0.020
0.015
0.010
0.005
0.000
-0.005
-0.010
-0.015
01-05
01-23
02-17
03-08
03-24
04-17
05-10
05-26
06-19

三、因子筛选



因子目标

1）避免大量公式复杂的价量因子，选择直观与逻辑上便于理解的

----------此处为与多因子选股不同之处

2）因子间要有低相关性：

----------构造了一个衡量相关性强弱的构造指标（针对月度选择因子）

3）因子回测文件 ----recursion.py 执行文件
----backtest.py 函数脚本

4）因子筛选：
月度调仓，人工手动分析，观察各因子IC和IC decay

CICC 中金公司

三、因子筛选

大类因子	小类因子	因子计算方式
Size	LNCAP	个股总市值对数：反应大盘股。
Beta	BETA(50,300,500,800,1000)	$r_i = \alpha + \beta r_m + e_i$：利用个股收益率序列和沪深300指数收益率序列进行一元线性回归，收益率序列长度取252交易日。个股收益率序列和沪深300指数收益率序列均以半衰指数加权，半衰期为63日。 [表现最好的为800]
RV	R(50,300,500,800,1000)	上述回归中的残差序列，由于选定了800，因此残差也选800。
Residual Volatility	HAlpha	个股500天收益与上证综指回归的截距项。
	Wgt_rt	个股1个月以每日换手率作为权重对每日收益率求算数平均。
Non-linear Size	NLSIZE	$SIZE$的立方：反应中盘股。

三、因子筛选

大类因子	小类因子	因子计算方式
TA	vol	过去一个月成交量的标准差。
Liquidity	STOM	$\mathrm{STOM} = \ln(\sum_{t=1}^{21} \frac{V_t}{S_t})$，其中$V_t$表示当日成交量，$S_t$表示流通股本。
	STOQ	$\mathrm{STOQ} = \ln(\frac{1}{\mathrm{T}} \sum_{\tau}^{T} \exp(STOM_{\tau}))$：其中T = 3，月度。
	STOA	$\mathrm{STOA} = \ln(\frac{1}{\mathrm{T}} \sum_{\tau}^{T} \exp(STOM_{\tau}))$：其中T = 12，月度。
Volatility	hml	过去1个月每日最高价-最低价序列的标准差。
	retv	过去1个月收益率的标准差。
Growth	CAGR	一致预测净利润2年复合增长率 。
	inrease	企业盈利同比增长率。
	ROE	净资产收益率TTM 。

三、因子筛选

大类因子	小类因子	因子计算方式
Leverage	MLEV	$MLEV = (ME + LD) / ME$：其中ME表示企业当前总市值，LD表示企业长期负债。
	DTOA	$DTOA = TD / TA$：其中TD表示总负债，TA表示总资产。
	BLEV	$BLEV = (BE + LD) / BE$：其中BE表示企业账面权益，LD表示企业长期负债。
Turn	turn	过去一个月换手率的标准差。
Value	EP	市盈率ttm的倒数
Tech	ASI	累计振动升降指标。
	BR	多空双方力量强弱指标：1月中h-c的和除以c-l的和。
	DDI	方向标准离差指数。
	Hurst	Hurst条件概率指标。
	KDJ	KDJ随机指标。
	MFI	资金流量指标。
	Ulcer	收盘价相对于最高价变动率指标。

三、因子筛选

判断因子相关性

$$C_{AB} = \frac{\text{mean}(Corr_i^{AB} | i = 1, \cdots, N)}{std(Corr_i^{AB} | i = 1, \cdots, N)}$$

该指标逻辑如下：
$Corr_i^{AB}$为A与B两个因子在第 i 天对全股票池横截面的相关系数

按照更新归因模型的频率来看（每月更一次），N可以取20(即20个交易日）

若想知道两个重要因子间是否长期相关，可以将N取久一些，例如取600.

【注：该指标为自己构造，可能不准确，慎用】

四、CTA引入

目标：判断市场处于震荡还是趋势

涉及领域：小波去噪、信号处理

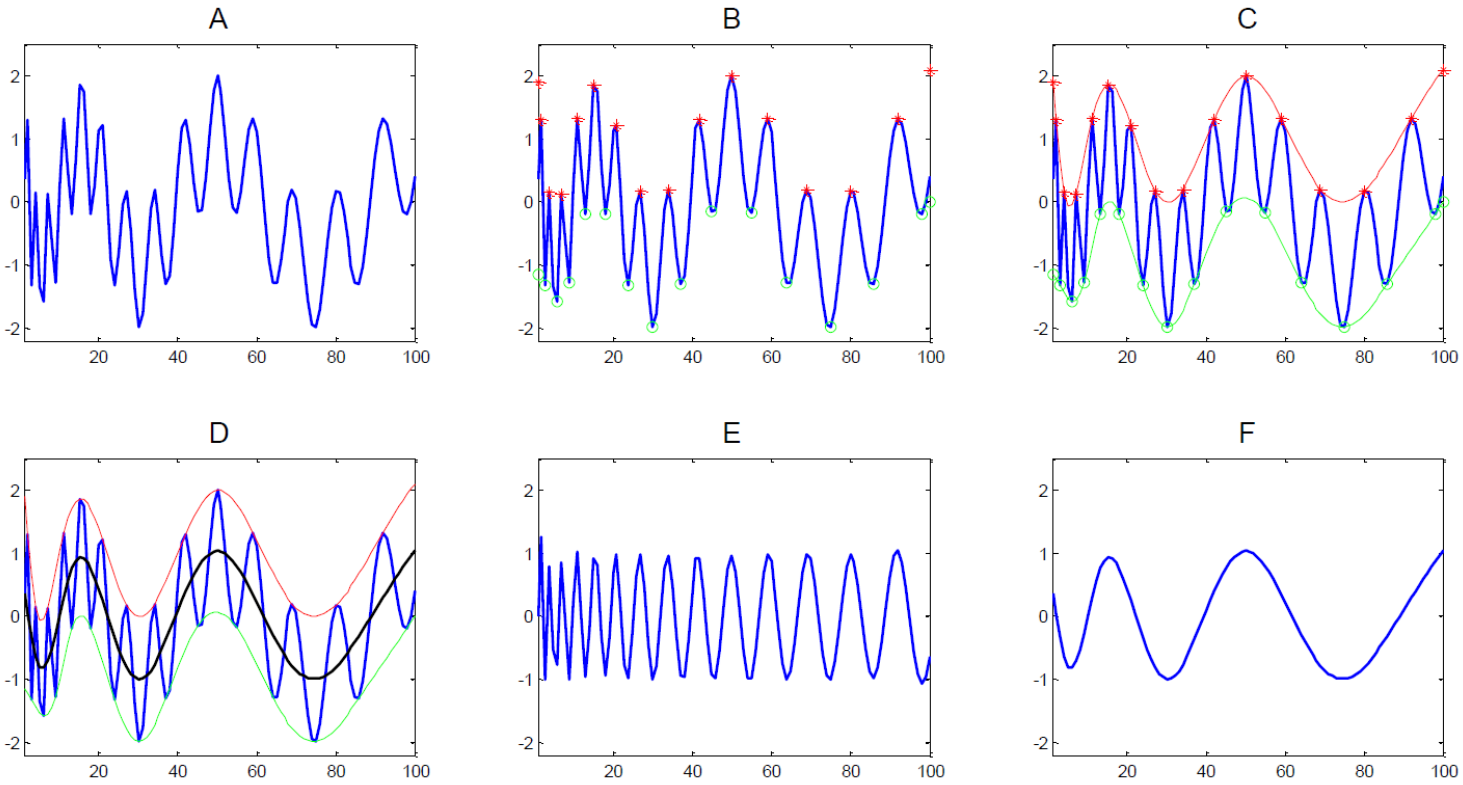
两步法：

1. **运用历史数据优化出好的时点**
2. **运用机器学习找出好的时间间隔进行预测**

四、CTA引入

EMD经验模态分解
复杂信号分解为 波动项+趋势项：

$$s(t) = \sum_{i=1,2,\cdots,n} IMF_I(t) + r_n(t)$$



四、CTA引入

R-Breaker实证分析

右图为主函数

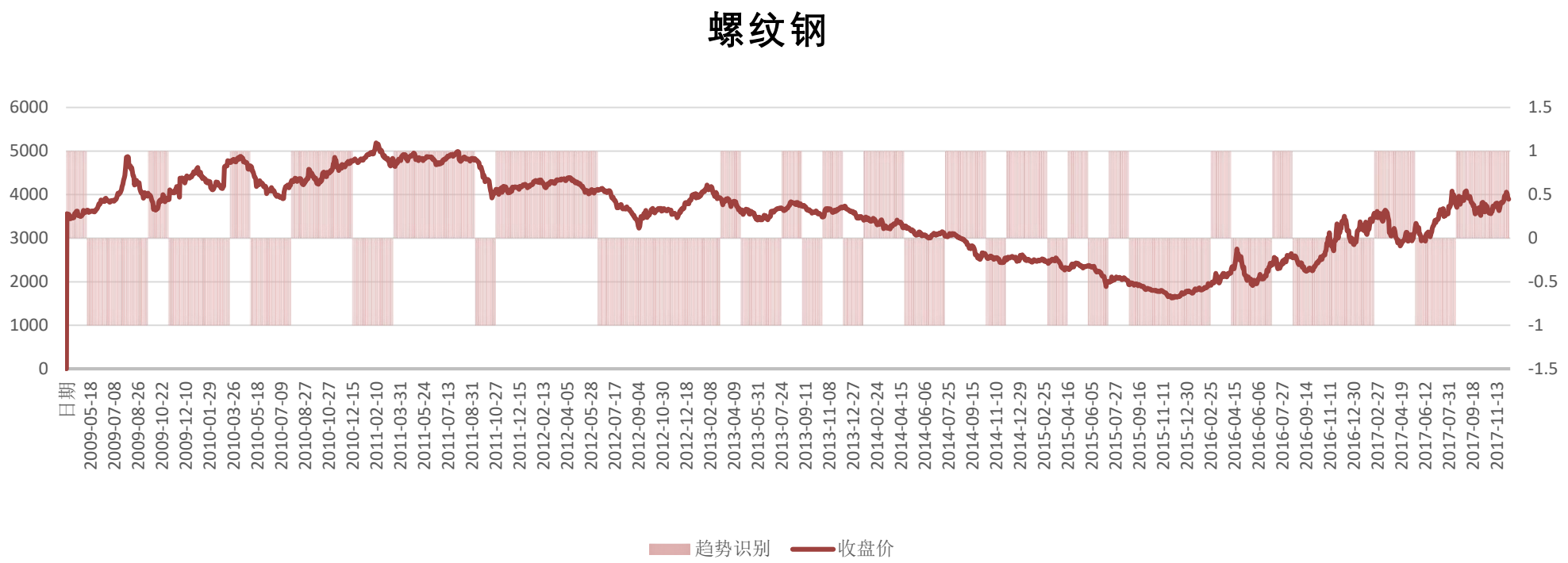
按比例止盈止损
初步定为止盈10%
止损5%

```
def run(self):
    self.counter = self.counter + 1
    if self.counter >= self.data_freq / self.Run_freq:
        self.cal_thr()
        self.counter = 0
    if self.ObserSellThr is None:
        return

    trade_flag = 0

    last_price = self.PriceList[-1]
    if marketPosition.marketPosition[self.symbol] == 0:
        if last_price < self.BreakSellThr:
            trade_flag = -1
            #order.insert_openShortPosition(symbol=self.symbol,volume=self.trade_shares, price=None, is_market=True)
            #空仓情况下，盘中价格跌破突破卖出价，采取趋势策略，即在该点做空；
        if last_price > self.BreakBuyThr:
            trade_flag = 1
            #order.insert_openLongPosition(symbol=self.symbol,volume=self.trade_shares, price=None, is_market=True)
            #空仓情况下，盘中价格超过突破买入价，采取趋势策略，即在该点做多


    if last_price > self.ObserSellThr:
        self.isCrossObserSellThr = True
    if last_price < self.ObserBuyThr:
        self.isCrossObserBuyThr = True
```

CICC 中金公司

四、CTA引入

结果示例：螺纹钢（目前做了21个商品品种和南华指数）



基于Barra模型的业绩归因分析

财富研究部丨 2017年11月3日

CICC 中金公司