精勤求学　敦笃励志

果毅力行　忠恕任事

遗传规划挖因子

金融工程71

周一飞、刘晓、姜泓任、马毓婕

精勤求学 敦笃励志

果毅力行 忠恕任事

遗传规划简介
Introduction to Genetic Programming

因子挖掘流程
Factor mining process

测试结果分析
Test results and factor analysis

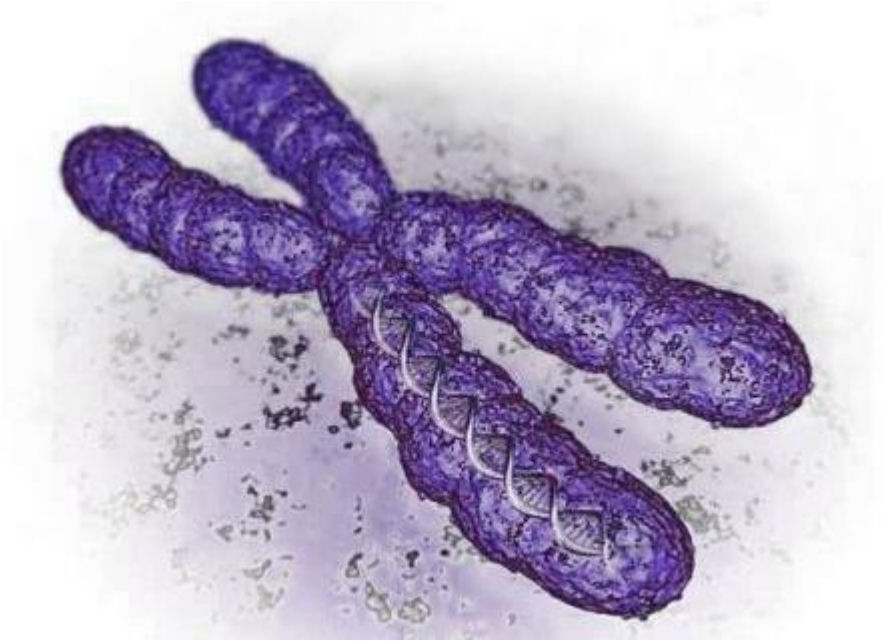
总结与思考
Conclusion thinking

1

遗传规划简介

Introduction to Genetic Programming

1.遗传规划简介



物竞天择，适者生存

——《天演论》

- **遗传规划(genetic programming)**：

其是演化算法(evolutionary algorithm)的分支，是一种启发式的优化算法。遗传规划从随机生成的公式群体开始，通过模拟自然界中遗传进化的过程，一般包括基因编码，种群初始化，交叉变异算子，经营保留机制等基本操作来逐渐生成契合特定目标的公式群体。作为一种监督学习方法，遗传规划可以根据特定目标，发现某些隐藏的、难以通过人脑构建出的数学公式。

- **因子研究思路的转变：**

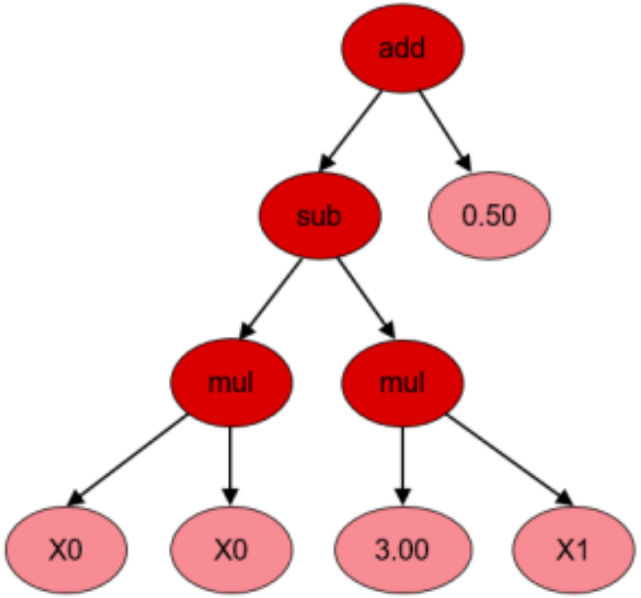
“先有逻辑、后有公式” —— “先有公式、后有逻辑”

在海量的数据中探索，并进一步根据一定目标筛选有效的因子，在试图取解释因子的内涵。

1.遗传规划简介

个体公式表示

- **原始公式** $y = X_0^2 - 3 * X_1 + 0.5$
- **前缀表达式** $y = \left(+(-* X_0 X_0)(* 3 X_1)\right) 0.5\right)$
- **树**



适应度

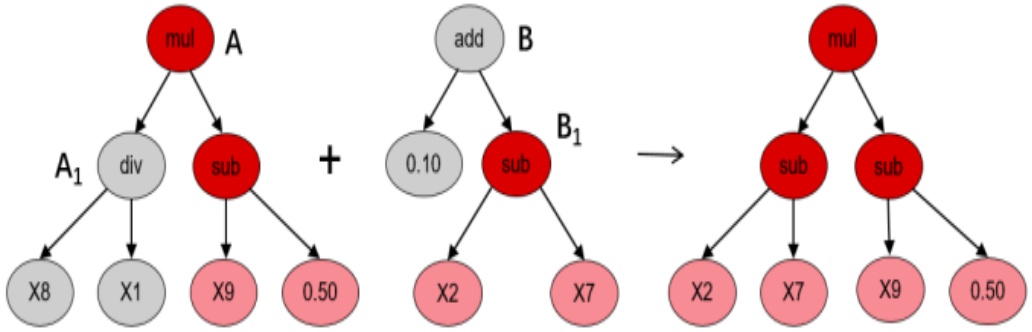
选股因子在回测区间内：

- **平均Rank IC**
- **因子收益率**

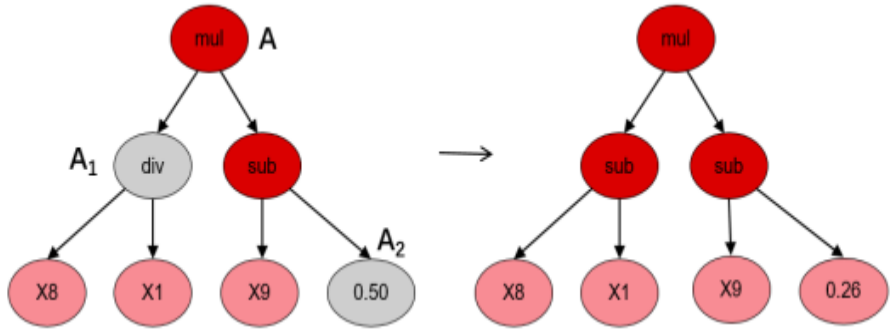
1.遗传规划简介

四种公式进化方法

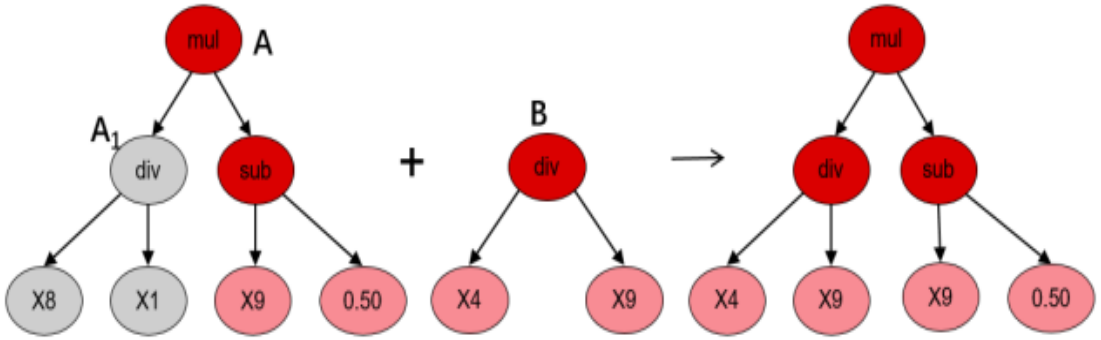
● 交叉



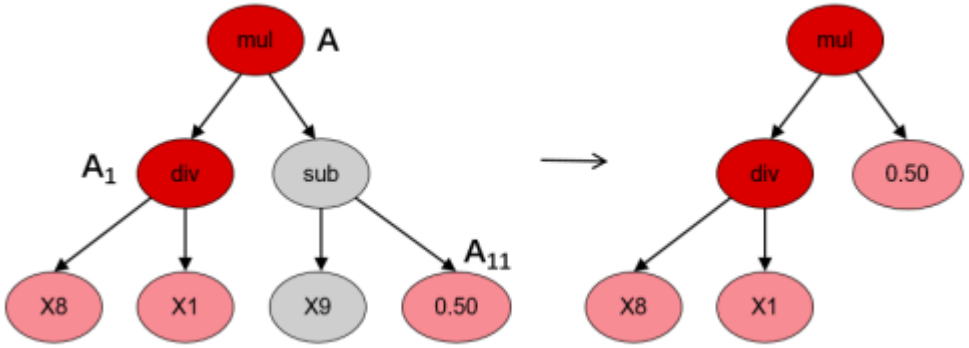
● 点变异



●子树变异

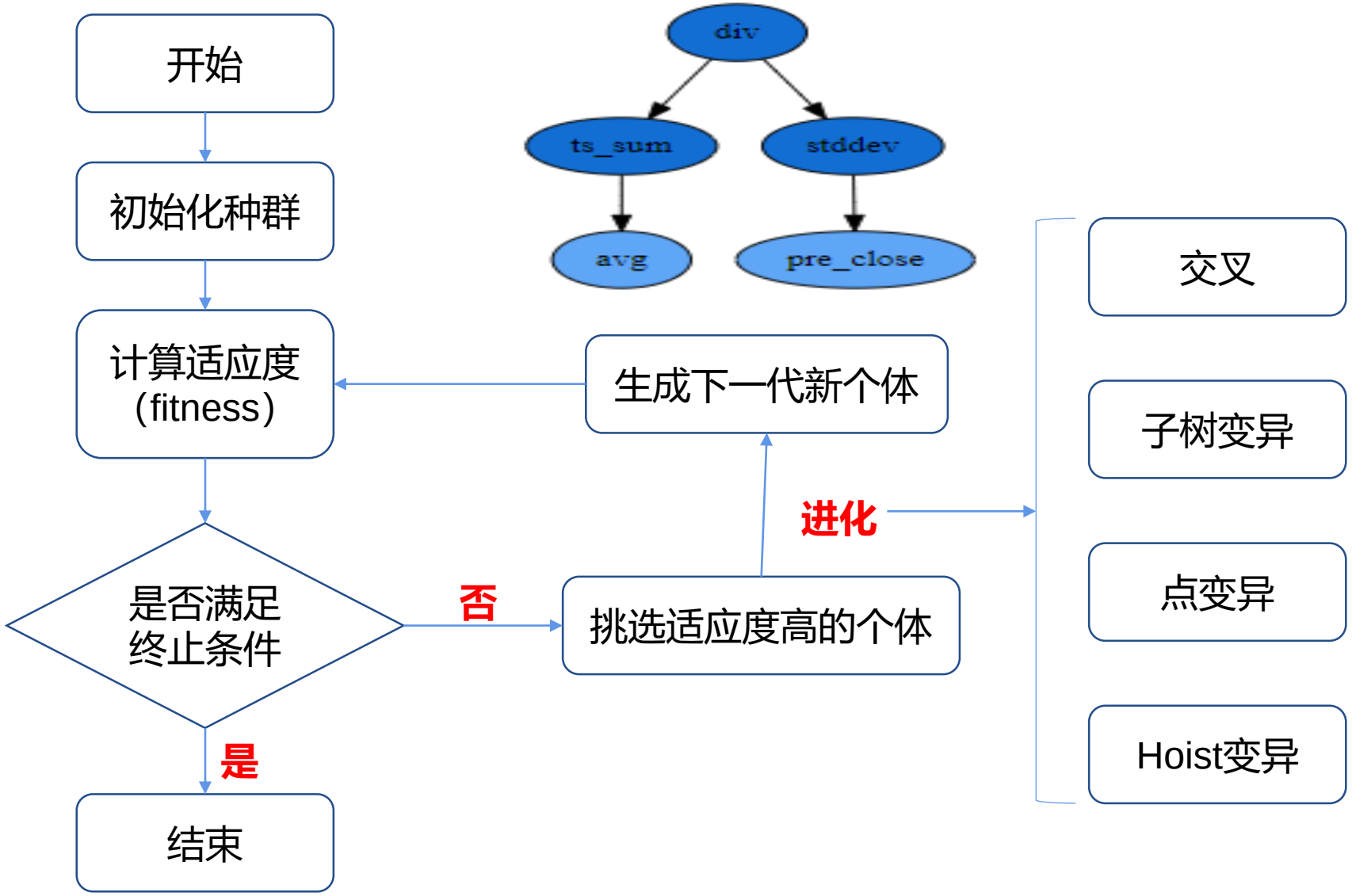


● Hoist变异



1.遗传规划简介

遗传规划流程图



2

因子挖掘流程

Factor mining process

2.因子挖掘流程

（1）数据获取

- 训练集数据范围: 2017.6.1_2018.6.1(包含牛熊)， 2017.1.1_2018.1.1(只有牛市)
- 股票池:沪深300成分股
- 数据字段： ['open', 'high', 'low', 'avg', 'pre_close', 'close','volume']
- 标签: 未来3天的收益率， 未来20天的收益率

2.因子挖掘流程

（2）使用遗传规划进行因子挖掘

- 使用 gplearn 库
- 自定义公式函数群　10+16
- 定义适应度 —— spearman
- 定义模型参数，增加惩罚因子，约束公式深度
- 输出4个模型，每个模型输出10个公式

```
generations = 3
function_set = init_function + user_function
#metric = my_metric
init_depth=(1,3) #最初生成树的深度(min_depth，max_depth)
population_size = 100
random_state=0
tournament_size=20
est_gp = SymbolicTransformer(
                             feature_names=fields,
                             function_set=function_set,
                             generations=generations,
                             metric='spearman',    #'spearman'秩相关系数
                             parsimony_coefficient=0.004,#惩罚 节俭系数(越大，约束越强，默认0.001)
                             init_depth=init_depth, # 公式树的初始化深度
                             population_size=population_size,
                             tournament_size= tournament_size,
                             random_state=random_state,
                             p_crossover = 0.4,
                             p_subtree_mutation = 0.01,
                             p_hoist_mutation = 0,
                             p_point_mutation = 0.01,
                             p_point_replace = 0.4,
                            )

est_gp.fit(X_train, y_train)
```

自定义函数	rank(X)	返回值为向量，其中第 i 个元素为X_i在向量 X 中的分位数。
自定义函数	delay(X, d)	返回值为向量，d 天以前的 X 值。
自定义函数	correlation(X, Y, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列和Y_i值构成的时序数列的相关系数。
自定义函数	covariance(X, Y, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列和Y_i值构成的时序数列的协方差。
自定义函数	scale(X, a)	返回值为向量 a*X/sum(abs(x))，a 的缺省值为 1，一般 a 应为正数。
自定义函数	delta(X, d)	返回值为向量 X - delay(X, d)。
自定义函数	signedpower(X, a)	返回值为向量 sign(X).*(abs(X).^a)，其中.*和.^两个运算符代表向量中对应元素相乘、元素乘方。
自定义函数	decay_linear(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列的加权平均值，权数为 d, d－1, …, 1(权数之和应为 1，需进行归一化处理)，其中离现在越近的日子权数越大。
自定义函数	indneutralize(X, indclass)	返回值为向量，对 X 进行行业中性化处理，indclass 取为中信一级行业。
自定义函数	ts_min(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列中最小值。
自定义函数	ts_max(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列中最大值。
自定义函数	ts_argmin(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列中最小值出现的位置。
自定义函数	ts_argmax(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列中最大值出现的位置。
自定义函数	ts_rank(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列中本截面日X_i值所处分位数。
自定义函数	ts_sum(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列之和
自定义函数	ts_product(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列的连乘乘积。
自定义函数	ts_stddev(X, d)	返回值为向量，其中第 i 个元素为过去 d 天X_i值构成的时序数列的标准差。

自定义公式函数群　10+16

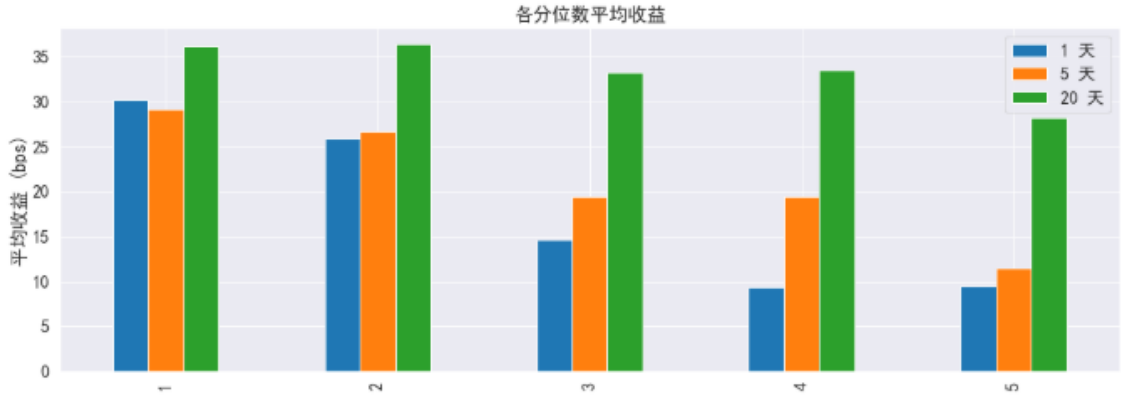
参数名称	定义和参数设置说明	参数设置
generations	公式进化的世代数量。世代数量越多，消耗算力越多，公式的进化次数越多。	3
population_size	每一代公式群体中的公式数量。公式数量越大，消耗算力越多，公式之间组合的空间越大。	1000
function_set	用于构建和进化公式时使用的函数集，可自定义更多函数。	使用图表 8 中的函数集
init_depth	公式树的初始化深度，init_depth 是一个二元组(min_depth, max_depth)，树的初始深度将处在[min_depth, max_depth]区间内。设置树深度最小 1 层，最大 4 层。最大深度越深，可能得出越复杂的因子，但是因子的意义更难解释。	(1,4)
tournament_size	每一代的所有公式中，tournament_size 个公式会被随机选中，其中适应度最高的公式能进行变异或繁殖生成下一代公式。tournament_size 越小，随机选择范围越小，选择的结果越不确定	20
metric	适应度指标，可自定义更多指标。	自定义的 RankIC 指标
p_crossover	父代进行交叉变异进化的概率。交叉变异是最有效的进化方式，可以设置为较大概率。	0.4
p_subtree_mutation	父代进行子树变异进化的概率。子树变异的结果不稳定，概率不宜过大。	0.01
p_hoist_mutation	父代进行 Hoist 变异进化的概率。本文的测试中公式树层次都较低，所以没有使用 Hoist 变异。	0
p_point_mutation	父代进行点变异进化的概率。点变异的结果不稳定，概率不宜过大。	0.01
p_point_replace	即点变异中父代每个节点进行变异进化的概率。点变异的概率已经很小，可设置为较大概率保证点变异的执行。	0.4

定义模型参数

2.因子挖掘流程

（3）单因子测试

- 测试集数据范围：2020.04-2020.07
- 股票池:沪深300成分股
- 数据字段：['open', 'high', 'low', 'avg', 'pre_close', 'close', 'volume']
- 分层收益分析
- IC分析



3

测试结果分析

Test results and factor analysis

3.测试结果分析

模型1：牛熊两市+未来3日收益率

	fitness	expression	depth	length
alpha_1	0.0314829	log(stddev(sma(low)))	3	4
alpha_3	0.0263626	stddev(stddev(sma(close)))	3	4
alpha_6	0.0238864	log(stddev(pre_close))	2	3
alpha_2	0.0214272	log(stddev(stddev(sma(close))))	4	5
alpha_4	0.0211335	stddev(stddev(stddev(close)))	3	4
alpha_9	0.0164814	sma(low)	1	2
alpha_5	0.0111292	stddev(sma(ts_max(stddev(pre_close))))	4	5
alpha_8	0.00903541	stddev(ts_max(stddev(pre_close)))	3	4
alpha_7	0.00294866	log(stddev(ts_max(stddev(pre_close))))	4	5
alpha_10	-0.0387033	log(sin(ts_min(close)))	3	4

模型2：牛熊两市+未来20日收益率

	depth	expression	fitness	length
alpha_3	1	stddev(low, 20)	0.0785097	3
alpha_4	1	stddev(close, 20)	0.0770966	3
alpha_5	1	stddev(open, 20)	0.0770124	3
alpha_7	1	stddev(pre_close, 20)	0.0764108	3
alpha_9	1	stddev(high, 20)	0.0737471	3
alpha_10	1	stddev(high, 20)	0.0737471	3
alpha_1	2	stddev(stddev(high, 20), 20)	0.066714	5
alpha_6	2	cube(stddev(avg, 20))	0.0666987	4
alpha_8	2	sqrt(stddev(pre_close, 20))	0.0664108	4
alpha_2	2	stddev(stddev(high, 20), 15)	0.0649213	5

模型3：牛市+未来3日收益率

	depth	expression	fitness	length
alpha_2	1	ts_rank(pre_close)	0.0545167	2
alpha_1	2	ts_rank(ts_rank(pre_close))	0.0543059	3
alpha_3	2	max(ts_rank(low), ts_rank(pre_close))	0.0469682	5
alpha_4	3	ts_rank(max(ts_rank(low), ts_rank(pre_close)))	0.0444154	6
alpha_5	3	mul(ts_rank(scale(close)), 0.671)	0.0434614	5
alpha_6	2	max(ts_rank(low), ts_rank(low))	0.0422986	5
alpha_7	2	ts_rank(ts_rank(low))	0.0379246	3
alpha_8	1	ts_rank(volume)	0.0166262	2
alpha_9	2	delay(add(-0.807, open))	0.00500159	4
alpha_10	2	sin(ts_min(close))	0.00112382	3

模型4：牛市+未来20日收益率

	depth	expression	fitness	length
alpha_1	2	ts_max(max(volume, 0.225))	0.0485949	4
alpha_2	2	ts_sum(scale(volume))	0.0467758	3
alpha_3	1	ts_sum(volume)	0.0506937	2
alpha_4	2	ts_sum(ts_sum(volume))	0.045971	3
alpha_5	1	scale(volume)	0.0471454	2
alpha_6	1	delay(volume)	0.0459509	2
alpha_7	2	delay(add(-0.807, open))	0.00644231	4
alpha_8	1	square(high)	0.0141421	2
alpha_9	1	ts_max(0.225)	-0.00461432	2
alpha_10	2	sin(ts_min(close))	-0.00953748	3

3.测试结果分析

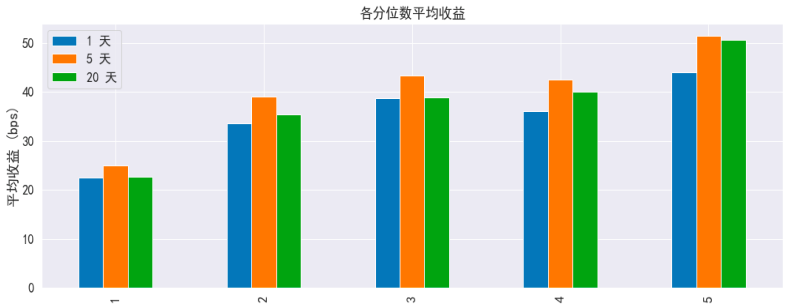
模型1：牛熊两市+未来3日收益率

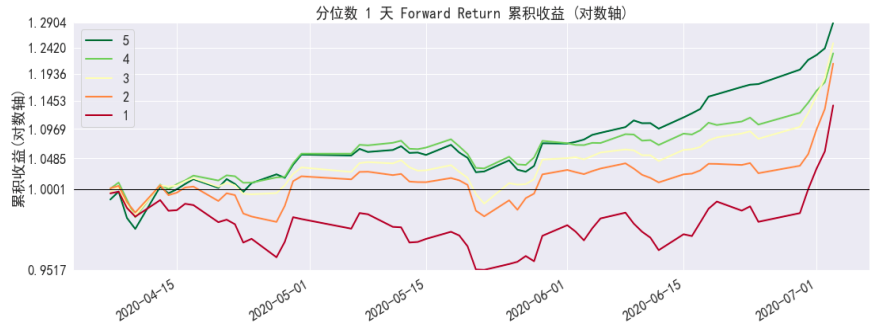
alpha6　log($stddev(pre\_close)$)

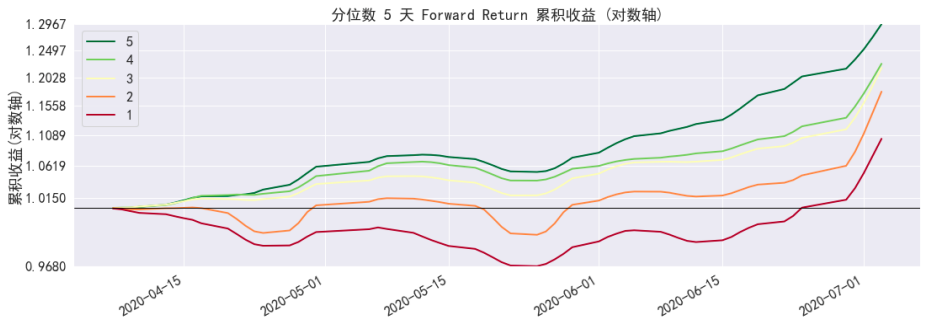
#第 i 个元素为过去 d 天Xi值构成的时序数列之标准差

收益分析

	period_1	period_5	period_20
Ann. alpha	0.265	0.454	0.389
beta	0.212	0.080	0.130
Mean Period Wise Return Top Quantile (bps)	44.019	51.361	50.595
Mean Period Wise Return Bottom Quantile (bps)	22.539	24.944	22.713
Mean Period Wise Spread (bps)	21.480	26.847	27.969







	period_1	period_5	period_20
IC Mean	0.028	0.105	0.217
IC Std.	0.190	0.175	0.128
IR	0.147	0.600	1.705
t-stat(IC)	1.130	4.610	13.099
p-value(IC)	0.263	0.000	0.000
IC Skew	-0.255	-0.960	0.352
IC Kurtosis	-0.482	0.087	-0.882

	period_1	period_20	period_5
Quantile 1 Mean Turnover	0.042	0.138	0.121
Quantile 2 Mean Turnover	0.089	0.282	0.262
Quantile 3 Mean Turnover	0.107	0.374	0.323
Quantile 4 Mean Turnover	0.113	0.398	0.329
Quantile 5 Mean Turnover	0.053	0.176	0.151

	period_1	period_5	period_20
Mean Factor Rank Autocorrelation	0.996	0.97	0.961

3.测试结果分析

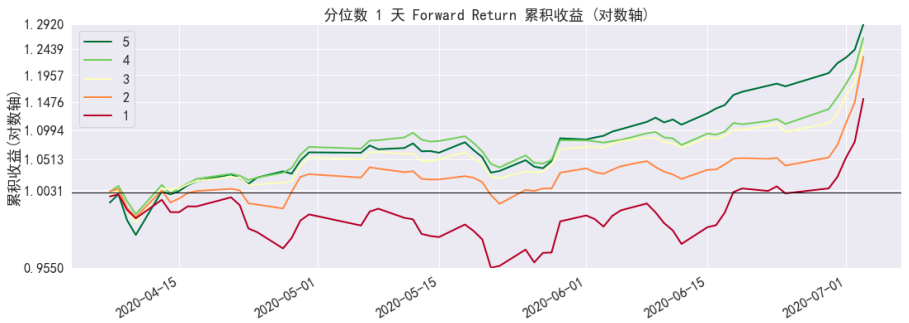
模型2：牛熊两市+未来20日收益率

alpha8 stddev(ts_max(stddev(pre_close)))

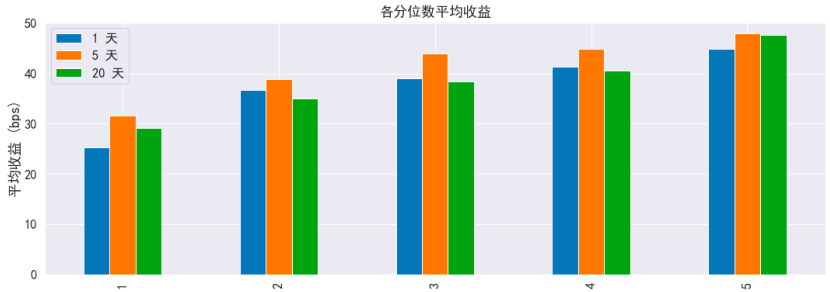
Ts_max其中第 i 个元素为过去 d 天X_i值构成的时序数列中最大值

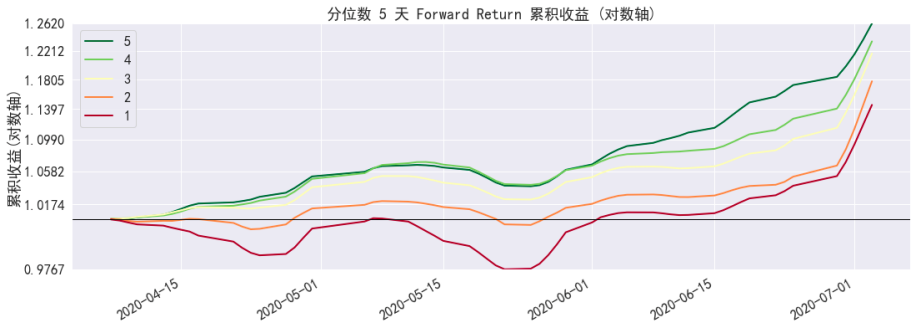
收益分析

	period_1	period_5	period_20
Ann. alpha	0.370	0.434	0.435
beta	0.813	0.692	0.816
Mean Period Wise Return Top Quantile (bps)	44.942	47.889	47.579
Mean Period Wise Return Bottom Quantile (bps)	25.353	31.672	29.193
Mean Period Wise Spread (bps)	19.588	16.630	18.752



	period_1	period_5	period_20
IC Mean	0.026	0.074	0.165
IC Std.	0.152	0.153	0.143
IR	0.173	0.483	1.158
t-stat(IC)	1.320	3.677	8.816
p-value(IC)	0.192	0.001	0.000
IC Skew	-0.295	-0.743	0.583
IC Kurtosis	-0.539	-0.306	-0.617





换手率分析

	period_1	period_20	period_5
Quantile 1 Mean Turnover	0.229	0.494	0.497
Quantile 2 Mean Turnover	0.251	0.589	0.550
Quantile 3 Mean Turnover	0.243	0.597	0.553
Quantile 4 Mean Turnover	0.218	0.585	0.537
Quantile 5 Mean Turnover	0.094	0.354	0.354

	period_1	period_5	period_20
Mean Factor Rank Autocorrelation	0.907	0.673	0.696

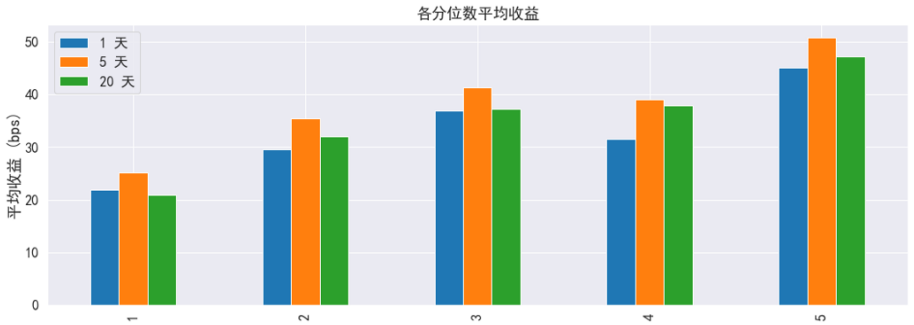
3.测试结果分析

模型4：牛市+未来20日收益率

alpha7　*Delay(add(-0.807,open))*

	period_1	period_5	period_20
Ann. alpha	0.440	0.608	0.480
beta	0.842	0.749	0.820
Mean Period Wise Return Top Quantile (bps)	45.035	50.723	47.219
Mean Period Wise Return Bottom Quantile (bps)	21.827	25.220	20.881
Mean Period Wise Spread (bps)	23.208	26.286	26.687

<Figure size 432x288 with 0 Axes>



	period_1	period_5	period_20
IC Mean	0.038	0.117	0.222
IC Std.	0.174	0.170	0.130
IR	0.218	0.687	1.708
t-stat(IC)	1.784	5.623	13.983
p-value(IC)	0.079	0.000	0.000
IC Skew	-0.590	-1.061	0.130
IC Kurtosis	0.096	0.737	-1.201

<Figure size 432x288 with 0 Axes>

换手率分析

	period_1	period_20	period_5
Quantile 1 Mean Turnover	0.004	0.014	0.009
Quantile 2 Mean Turnover	0.012	0.048	0.029
Quantile 3 Mean Turnover	0.022	0.093	0.048
Quantile 4 Mean Turnover	0.020	0.080	0.042
Quantile 5 Mean Turnover	0.007	0.023	0.014

	period_1	period_5	period_20
Mean Factor Rank Autocorrelation	1.0	1.0	0.999

<Figure size 432x288 with 0 Axes>

4

总结思考

Conclusion thinking

4.总结思考

（1）项目学习总结

- 1.了解了遗传规划的基本原理及操作
- 2.学习了Python中gplearn库中对遗传规划的使用，各种参数的定义
- 3.实验：选取了沪深300成分股，在牛市和牛熊两市时间段下的数据，并依据3日，20日收益率作为标签进行因子挖掘。
- 4.单因子检验：回顾单因子分析流程，探索因子公式的表现

（2）小组分工

- 周一飞，姜泓任：因子挖掘代码及优化
- 刘晓：代码优化及单因子测试，PPT展示
- 马毓婕：代码优化及word报告

4.总结思考

(3) 遇到的困难和解决对策:

- 聚宽上无法使用gplearn —— 跨平台使用本地jupyter与聚宽平台
- 时间线太长，噪声太多 —— 取特定的牛熊时间段训练
- 函数太少缺少灵活性 —— 改写了部分函数使得参数可变
- 因子长度太长 —— 加惩罚因子参数，约束公式深度，防止过拟合
- 无法批量生成因子 —— 研究gplearn参数文档，找到对应方法
- 因子公式重复 —— 调整参数，换时间区间

仍需提升:

- 适应度因子预处理:改写函数可以使用,但因子横截面回归与原数据格式不符,需要大幅改动源代码
- 数据量及速度：当数据量变大时，没有快速并行算法，运算速度较慢

精勤求学　敦笃励志

果毅力行　忠恕任事

感谢聆听

期待您的指导!