# World\_Quant\_Alphas

该Repository下每个文件下是对World Quant 101 alphas中的一个的复制，其中WQ\_Alphas\_Analysis.py是计算各个因子的代码，以及对因子进行分组分析的代码，而类似#041文件下下的alpha#41.py是第41号因子策略化的代码，回测系统使用的是量子金服云宽客策略平台。

下面是从论文中摘出来的101个因子的计算公式

\* Alpha#1:

(rank(Ts\_ArgMax(SignedPower(((returns < 0) ? stddev(returns, 20) : close), 2.), 5)) - 0.5)

\* Alpha#2:

(-1 \* correlation(rank(delta(log(volume), 2)), rank(((close - open) / open)), 6))

\* Alpha#3:

(-1 \* correlation(rank(open), rank(volume), 10))

\* Alpha#4:

(-1 \* Ts\_Rank(rank(low), 9))

\* Alpha#5:

(rank((open - (sum(vwap, 10) / 10))) \* (-1 \* abs(rank((close - vwap)))))

\* Alpha#6:

(-1 \* correlation(open, volume, 10))

\* Alpha#7:

((adv20 < volume) ? ((-1 \* ts\_rank(abs(delta(close, 7)), 60)) \* sign(delta(close, 7))) : (-1\* 1))

\* Alpha#8:

(-1 \* rank(((sum(open, 5) \* sum(returns, 5)) - delay((sum(open, 5) \* sum(returns, 5)), 10))))

\* Alpha#9:

((0 < ts\_min(delta(close, 1), 5)) ? delta(close, 1) : ((ts\_max(delta(close, 1), 5) < 0) ? delta(close, 1) : (-1 \* delta(close, 1))))

\* Alpha#10:

rank(((0 < ts\_min(delta(close, 1), 4)) ? delta(close, 1) : ((ts\_max(delta(close, 1), 4) < 0) ? delta(close, 1) : (-1 \* delta(close, 1)))))

\* Alpha#11:

((rank(ts\_max((vwap - close), 3)) + rank(ts\_min((vwap - close), 3))) \* rank(delta(volume, 3)))

\* Alpha#12:

(sign(delta(volume, 1)) \* (-1 \* delta(close, 1)))

\* Alpha#13:

(-1 \* rank(covariance(rank(close), rank(volume), 5)))

\* Alpha#14:

((-1 \* rank(delta(returns, 3))) \* correlation(open, volume, 10))

\* Alpha#15:

(-1 \* sum(rank(correlation(rank(high), rank(volume), 3)), 3))

\* Alpha#16:

(-1 \* rank(covariance(rank(high), rank(volume), 5)))

\* Alpha#17:

(((-1 \* rank(ts\_rank(close, 10))) \* rank(delta(delta(close, 1), 1))) \* rank(ts\_rank((volume / adv20), 5)))

\* Alpha#18:

(-1 \* rank(((stddev(abs((close - open)), 5) + (close - open)) + correlation(close, open, 10))))

\* Alpha#19:

((-1 \* sign(((close - delay(close, 7)) + delta(close, 7)))) \* (1 + rank((1 + sum(returns, 250)))))

\* Alpha#20:

(((-1 \* rank((open - delay(high, 1)))) \* rank((open - delay(close, 1)))) \* rank((open - delay(low, 1))))

\* Alpha#21:

((((sum(close, 8) / 8) + stddev(close, 8)) < (sum(close, 2) / 2)) ? (-1 \* 1) : (((sum(close, 2) / 2) < ((sum(close, 8) / 8) - stddev(close, 8))) ? 1 : (((1 < (volume / adv20)) || ((volume / adv20) == 1)) ? 1 : (-1 \* 1))))

\* Alpha#22:

(-1 \* (delta(correlation(high, volume, 5), 5) \* rank(stddev(close, 20))))

\* Alpha#23:

(((sum(high, 20) / 20) < high) ? (-1 \* delta(high, 2)) : 0)

\* Alpha#24:

((((delta((sum(close, 100) / 100), 100) / delay(close, 100)) < 0.05) || ((delta((sum(close, 100) / 100), 100) / delay(close, 100)) == 0.05)) ? (-1 \* (close - ts\_min(close, 100))) : (-1 \* delta(close, 3)))

\* Alpha#25:

rank(((((-1 \* returns) \* adv20) \* vwap) \* (high - close)))

\* Alpha#26:

(-1 \* ts\_max(correlation(ts\_rank(volume, 5), ts\_rank(high, 5), 5), 3))

\* Alpha#27:

((0.5 < rank((sum(correlation(rank(volume), rank(vwap), 6), 2) / 2.0))) ? (-1 \* 1) : 1)

\* Alpha#28:

scale(((correlation(adv20, low, 5) + ((high + low) / 2)) - close))

\* Alpha#29:

(min(product(rank(rank(scale(log(sum(ts\_min(rank(rank((-1 \* rank(delta((close - 1), 5))))), 2), 1))))), 1), 5) + ts\_rank(delay((-1 \* returns), 6), 5))

\* Alpha#30:

(((1.0 - rank(((sign((close - delay(close, 1))) + sign((delay(close, 1) - delay(close, 2)))) + sign((delay(close, 2) - delay(close, 3)))))) \* sum(volume, 5)) / sum(volume, 20))

\* Alpha#31:

((rank(rank(rank(decay\_linear((-1 \* rank(rank(delta(close, 10)))), 10)))) + rank((-1 \* delta(close, 3)))) + sign(scale(correlation(adv20, low, 12))))

\* Alpha#32:

(scale(((sum(close, 7) / 7) - close)) + (20 \* scale(correlation(vwap, delay(close, 5), 230))))

\* Alpha#33:

rank((-1 \* ((1 - (open / close))^1)))

\* Alpha#34:

rank(((1 - rank((stddev(returns, 2) / stddev(returns, 5)))) + (1 - rank(delta(close, 1)))))

\* Alpha#35:

((Ts\_Rank(volume, 32) \* (1 - Ts\_Rank(((close + high) - low), 16))) \* (1 - Ts\_Rank(returns, 32)))

\* Alpha#36:

(((((2.21 \* rank(correlation((close - open), delay(volume, 1), 15))) + (0.7 \* rank((open - close)))) + (0.73 \* rank(Ts\_Rank(delay((-1 \* returns), 6), 5)))) + rank(abs(correlation(vwap, adv20, 6)))) + (0.6 \* rank((((sum(close, 200) / 200) - open) \* (close - open)))))

\* Alpha#37:

(rank(correlation(delay((open - close), 1), close, 200)) + rank((open - close)))

\* Alpha#38:

((-1 \* rank(Ts\_Rank(close, 10))) \* rank((close / open)))

\* Alpha#39:

((-1 \* rank((delta(close, 7) \* (1 - rank(decay\_linear((volume / adv20), 9)))))) \* (1 + rank(sum(returns, 250))))

\* Alpha#40:

((-1 \* rank(stddev(high, 10))) \* correlation(high, volume, 10))

\* Alpha#41:

(((high \* low)^0.5) - vwap)

\* Alpha#42:

(rank((vwap - close)) / rank((vwap + close)))

\* Alpha#43:

(ts\_rank((volume / adv20), 20) \* ts\_rank((-1 \* delta(close, 7)), 8))

\* Alpha#44:

(-1 \* correlation(high, rank(volume), 5))

\* Alpha#45:

(-1 \* ((rank((sum(delay(close, 5), 20) / 20)) \* correlation(close, volume, 2)) \* rank(correlation(sum(close, 5), sum(close, 20), 2))))

\* Alpha#46:

((0.25 < (((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10))) ? (-1 \* 1) : (((((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10)) < 0) ? 1 : ((-1 \* 1) \* (close - delay(close, 1)))))

\* Alpha#47:

((((rank((1 / close)) \* volume) / adv20) \* ((high \* rank((high - close))) / (sum(high, 5) / 5))) - rank((vwap - delay(vwap, 5))))

\* Alpha#48:

(indneutralize(((correlation(delta(close, 1), delta(delay(close, 1), 1), 250) \* delta(close, 1)) / close), IndClass.subindustry) / sum(((delta(close, 1) / delay(close, 1))^2), 250))

\* Alpha#49:

(((((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10)) < (-1 \* 0.1)) ? 1 : ((-1 \* 1) \* (close - delay(close, 1))))

\* Alpha#50:

(-1 \* ts\_max(rank(correlation(rank(volume), rank(vwap), 5)), 5))

\* Alpha#51:

(((((delay(close, 20) - delay(close, 10)) / 10) - ((delay(close, 10) - close) / 10)) < (-1 \* 0.05)) ? 1 : ((-1 \* 1) \* (close - delay(close, 1))))

\* Alpha#52:

((((-1 \* ts\_min(low, 5)) + delay(ts\_min(low, 5), 5)) \* rank(((sum(returns, 240) - sum(returns, 20)) / 220))) \* ts\_rank(volume, 5))

\* Alpha#53:

(-1 \* delta((((close - low) - (high - close)) / (close - low)), 9))

\* Alpha#54:

((-1 \* ((low - close) \* (open^5))) / ((low - high) \* (close^5)))

\* Alpha#55:

(-1 \* correlation(rank(((close - ts\_min(low, 12)) / (ts\_max(high, 12) - ts\_min(low, 12)))), rank(volume), 6))

\* Alpha#56:

(0 - (1 \* (rank((sum(returns, 10) / sum(sum(returns, 2), 3))) \* rank((returns \* cap)))))

\* Alpha#57:

(0 - (1 \* ((close - vwap) / decay\_linear(rank(ts\_argmax(close, 30)), 2))))

\* Alpha#58:

(-1 \* Ts\_Rank(decay\_linear(correlation(IndNeutralize(vwap, IndClass.sector), volume, 3.92795), 7.89291), 5.50322))

\* Alpha#59:

(-1 \* Ts\_Rank(decay\_linear(correlation(IndNeutralize(((vwap \* 0.728317) + (vwap \* (1 - 0.728317))), IndClass.industry), volume, 4.25197), 16.2289), 8.19648))

\* Alpha#60:

(0 - (1 \* ((2 \* scale(rank(((((close - low) - (high - close)) / (high - low)) \* volume)))) - scale(rank(ts\_argmax(close, 10))))))

\* Alpha#61:

(rank((vwap - ts\_min(vwap, 16.1219))) < rank(correlation(vwap, adv180, 17.9282)))

\* Alpha#62:

((rank(correlation(vwap, sum(adv20, 22.4101), 9.91009)) < rank(((rank(open) + rank(open)) < (rank(((high + low) / 2)) + rank(high))))) \* -1)

\* Alpha#63:

((rank(decay\_linear(delta(IndNeutralize(close, IndClass.industry), 2.25164), 8.22237)) - rank(decay\_linear(correlation(((vwap \* 0.318108) + (open \* (1 - 0.318108))), sum(adv180, 37.2467), 13.557), 12.2883))) \* -1)

\* Alpha#64:

((rank(correlation(sum(((open \* 0.178404) + (low \* (1 - 0.178404))), 12.7054), sum(adv120, 12.7054), 16.6208)) < rank(delta(((((high + low) / 2) \* 0.178404) + (vwap \* (1 - 0.178404))), 3.69741))) \* -1)

\* Alpha#65:

((rank(correlation(((open \* 0.00817205) + (vwap \* (1 - 0.00817205))), sum(adv60, 8.6911), 6.40374)) < rank((open - ts\_min(open, 13.635)))) \* -1)

\* Alpha#66:

((rank(decay\_linear(delta(vwap, 3.51013), 7.23052)) + Ts\_Rank(decay\_linear(((((low \* 0.96633) + (low \* (1 - 0.96633))) - vwap) / (open - ((high + low) / 2))), 11.4157), 6.72611)) \* -1)

\* Alpha#67:

((rank((high - ts\_min(high, 2.14593)))^rank(correlation(IndNeutralize(vwap, IndClass.sector), IndNeutralize(adv20, IndClass.subindustry), 6.02936))) \* -1)

\* Alpha#68:

((Ts\_Rank(correlation(rank(high), rank(adv15), 8.91644), 13.9333) < rank(delta(((close \* 0.518371) + (low \* (1 - 0.518371))), 1.06157))) \* -1)

\* Alpha#69:

((rank(ts\_max(delta(IndNeutralize(vwap, IndClass.industry), 2.72412), 4.79344))^Ts\_Rank(correlation(((close \* 0.490655) + (vwap \* (1 - 0.490655))), adv20, 4.92416), 9.0615)) \* -1)

\* Alpha#70:

((rank(delta(vwap, 1.29456))^Ts\_Rank(correlation(IndNeutralize(close, IndClass.industry), adv50, 17.8256), 17.9171)) \* -1)

\* Alpha#71:

max(Ts\_Rank(decay\_linear(correlation(Ts\_Rank(close, 3.43976), Ts\_Rank(adv180, 12.0647), 18.0175), 4.20501), 15.6948), Ts\_Rank(decay\_linear((rank(((low + open) - (vwap + vwap)))^2), 16.4662), 4.4388))

\* Alpha#72:

(rank(decay\_linear(correlation(((high + low) / 2), adv40, 8.93345), 10.1519)) / rank(decay\_linear(correlation(Ts\_Rank(vwap, 3.72469), Ts\_Rank(volume, 18.5188), 6.86671), 2.95011)))

\* Alpha#73:

(max(rank(decay\_linear(delta(vwap, 4.72775), 2.91864)), Ts\_Rank(decay\_linear(((delta(((open \* 0.147155) + (low \* (1 - 0.147155))), 2.03608) / ((open \* 0.147155) + (low \* (1 - 0.147155)))) \* -1), 3.33829), 16.7411)) \* -1)

\* Alpha#74:

((rank(correlation(close, sum(adv30, 37.4843), 15.1365)) < rank(correlation(rank(((high \* 0.0261661) + (vwap \* (1 - 0.0261661)))), rank(volume), 11.4791))) \* -1)

\* Alpha#75:

(rank(correlation(vwap, volume, 4.24304)) < rank(correlation(rank(low), rank(adv50), 12.4413)))

\* Alpha#76:

(max(rank(decay\_linear(delta(vwap, 1.24383), 11.8259)), Ts\_Rank(decay\_linear(Ts\_Rank(correlation(IndNeutralize(low, IndClass.sector), adv81, 8.14941), 19.569), 17.1543), 19.383)) \* -1)

\* Alpha#77:

min(rank(decay\_linear(((((high + low) / 2) + high) - (vwap + high)), 20.0451)), rank(decay\_linear(correlation(((high + low) / 2), adv40, 3.1614), 5.64125)))

\* Alpha#78:

(rank(correlation(sum(((low \* 0.352233) + (vwap \* (1 - 0.352233))), 19.7428), sum(adv40, 19.7428), 6.83313))^rank(correlation(rank(vwap), rank(volume), 5.77492)))

\* Alpha#79:

(rank(delta(IndNeutralize(((close \* 0.60733) + (open \* (1 - 0.60733))), IndClass.sector), 1.23438)) < rank(correlation(Ts\_Rank(vwap, 3.60973), Ts\_Rank(adv150, 9.18637), 14.6644)))

\* Alpha#80:

((rank(Sign(delta(IndNeutralize(((open \* 0.868128) + (high \* (1 - 0.868128))), IndClass.industry), 4.04545)))^Ts\_Rank(correlation(high, adv10, 5.11456), 5.53756)) \* -1)

\* Alpha#81:

((rank(Log(product(rank((rank(correlation(vwap, sum(adv10, 49.6054), 8.47743))^4)), 14.9655))) < rank(correlation(rank(vwap), rank(volume), 5.07914))) \* -1)

\* Alpha#82:

(min(rank(decay\_linear(delta(open, 1.46063), 14.8717)), Ts\_Rank(decay\_linear(correlation(IndNeutralize(volume, IndClass.sector), ((open \* 0.634196) + (open \* (1 - 0.634196))), 17.4842), 6.92131), 13.4283)) \* -1)

\* Alpha#83:

((rank(delay(((high - low) / (sum(close, 5) / 5)), 2)) \* rank(rank(volume))) / (((high - low) / (sum(close, 5) / 5)) / (vwap - close)))

\* Alpha#84:

SignedPower(Ts\_Rank((vwap - ts\_max(vwap, 15.3217)), 20.7127), delta(close, 4.96796))

\* Alpha#85:

(rank(correlation(((high \* 0.876703) + (close \* (1 - 0.876703))), adv30, 9.61331))^rank(correlation(Ts\_Rank(((high + low) / 2), 3.70596), Ts\_Rank(volume, 10.1595), 7.11408)))

\* Alpha#86:

((Ts\_Rank(correlation(close, sum(adv20, 14.7444), 6.00049), 20.4195) < rank(((open + close) - (vwap + open)))) \* -1)

\* Alpha#87:

(max(rank(decay\_linear(delta(((close \* 0.369701) + (vwap \* (1 - 0.369701))), 1.91233), 2.65461)), Ts\_Rank(decay\_linear(abs(correlation(IndNeutralize(adv81, IndClass.industry), close, 13.4132)), 4.89768), 14.4535)) \* -1)

\* Alpha#88:

min(rank(decay\_linear(((rank(open) + rank(low)) - (rank(high) + rank(close))), 8.06882)), Ts\_Rank(decay\_linear(correlation(Ts\_Rank(close, 8.44728), Ts\_Rank(adv60, 20.6966), 8.01266), 6.65053), 2.61957))

\* Alpha#89:

(Ts\_Rank(decay\_linear(correlation(((low \* 0.967285) + (low \* (1 - 0.967285))), adv10, 6.94279), 5.51607), 3.79744) - Ts\_Rank(decay\_linear(delta(IndNeutralize(vwap, IndClass.industry), 3.48158), 10.1466), 15.3012))

\* Alpha#90:

((rank((close - ts\_max(close, 4.66719)))^Ts\_Rank(correlation(IndNeutralize(adv40, IndClass.subindustry), low, 5.38375), 3.21856)) \* -1)

\* Alpha#91:

((Ts\_Rank(decay\_linear(decay\_linear(correlation(IndNeutralize(close, IndClass.industry), volume, 9.74928), 16.398), 3.83219), 4.8667) - rank(decay\_linear(correlation(vwap, adv30, 4.01303), 2.6809))) \* -1)

\* Alpha#92:

min(Ts\_Rank(decay\_linear(((((high + low) / 2) + close) < (low + open)), 14.7221), 18.8683), Ts\_Rank(decay\_linear(correlation(rank(low), rank(adv30), 7.58555), 6.94024), 6.80584))

\* Alpha#93:

(Ts\_Rank(decay\_linear(correlation(IndNeutralize(vwap, IndClass.industry), adv81, 17.4193), 19.848), 7.54455) / rank(decay\_linear(delta(((close \* 0.524434) + (vwap \* (1 - 0.524434))), 2.77377), 16.2664)))

\* Alpha#94:

((rank((vwap - ts\_min(vwap, 11.5783)))^Ts\_Rank(correlation(Ts\_Rank(vwap, 19.6462), Ts\_Rank(adv60, 4.02992), 18.0926), 2.70756)) \* -1)

\* Alpha#95:

(rank((open - ts\_min(open, 12.4105))) < Ts\_Rank((rank(correlation(sum(((high + low) / 2), 19.1351), sum(adv40, 19.1351), 12.8742))^5), 11.7584))

\* Alpha#96:

(max(Ts\_Rank(decay\_linear(correlation(rank(vwap), rank(volume), 3.83878), 4.16783), 8.38151), Ts\_Rank(decay\_linear(Ts\_ArgMax(correlation(Ts\_Rank(close, 7.45404), Ts\_Rank(adv60, 4.13242), 3.65459), 12.6556), 14.0365), 13.4143)) \* -1)

\* Alpha#97:

((rank(decay\_linear(delta(IndNeutralize(((low \* 0.721001) + (vwap \* (1 - 0.721001))), IndClass.industry), 3.3705), 20.4523)) - Ts\_Rank(decay\_linear(Ts\_Rank(correlation(Ts\_Rank(low, 7.87871), Ts\_Rank(adv60, 17.255), 4.97547), 18.5925), 15.7152), 6.71659)) \* -1)

\* Alpha#98:

(rank(decay\_linear(correlation(vwap, sum(adv5, 26.4719), 4.58418), 7.18088)) - rank(decay\_linear(Ts\_Rank(Ts\_ArgMin(correlation(rank(open), rank(adv15), 20.8187), 8.62571), 6.95668), 8.07206)))

\* Alpha#99:

((rank(correlation(sum(((high + low) / 2), 19.8975), sum(adv60, 19.8975), 8.8136)) < rank(correlation(low, volume, 6.28259))) \* -1)

\* Alpha#100:

(0 - (1 \* (((1.5 \* scale(indneutralize(indneutralize(rank(((((close - low) - (high - close)) / (high - low)) \* volume)), IndClass.subindustry), IndClass.subindustry))) - scale(indneutralize((correlation(close, rank(adv20), 5) - rank(ts\_argmin(close, 30))), IndClass.subindustry))) \* (volume / adv20))))

\* Alpha#101:

((close - open) / ((high - low) + .001))

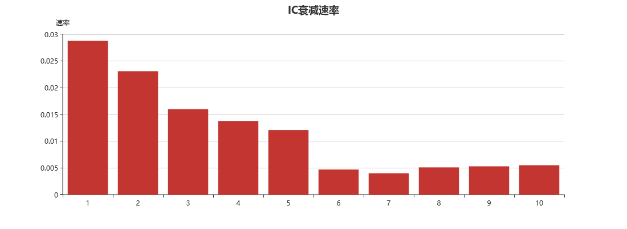
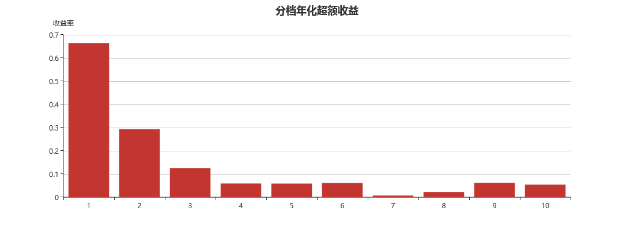
World Quant 101 Alphas 分析

|  |
| --- |
| #002 |

表示交易量较两日前的变化，正表示交易量增加，负表示交易量减少，可看做当日股价涨跌幅，正表示涨，负表示跌。忽略前面的负号时，该因子就是这两个指标的6日秩相关系数：为正，表示前6日股价上涨幅度排名靠前时，交易量增加量排名也靠前；为负，表示前6日股价上涨幅度排名靠前时，交易量增加量排名却靠后。取负号后则相反。该因子经过检验，发现是个反转因子，股价上涨幅度排名与交易量变化排名负相关的股票，在后期收益更高。

10组1日（20130101-20161231，中证500）

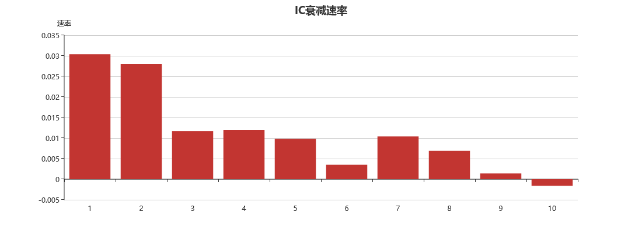
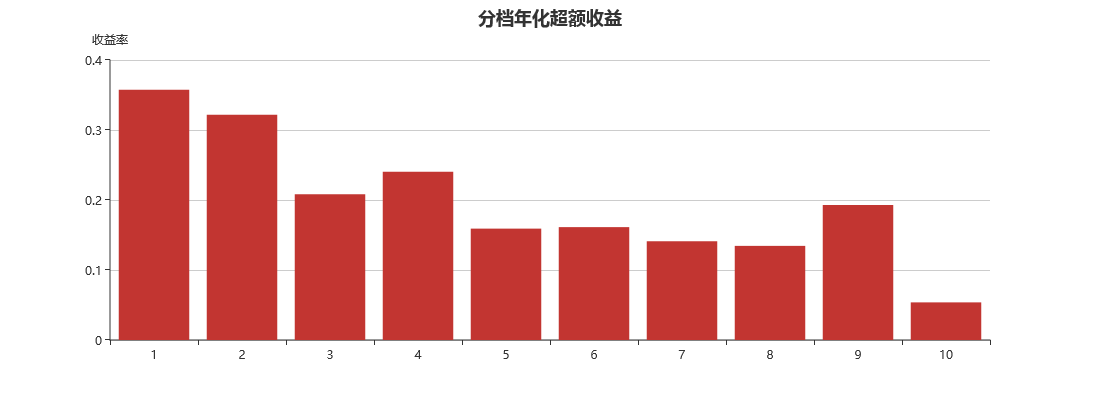




可以看出，1、2、3档有明显的单调递减趋势，而四档至以后的超额Alpha区分度不大，且单调性不强。该因子整体IC为0.0286，但各组内的IC却都低于0.01，说明各组内因子有效性不大。该因子的IC衰减较慢，当期因子与5期后的收益率的IC比率有0.01，下面将周期调为5天。

10组5日（20130101-20161231，中证500）





周期改为5天后，各组胜率均有所降低，第一组收益率降低，各组之间超额收益差距减小。IC衰减依然较慢。

当周期调整为10天甚至更长，各组之间收益率则无差别。

20日周期，100只股，回测结果如下：



与中证500无甚差别！

调仓周期改为10日：

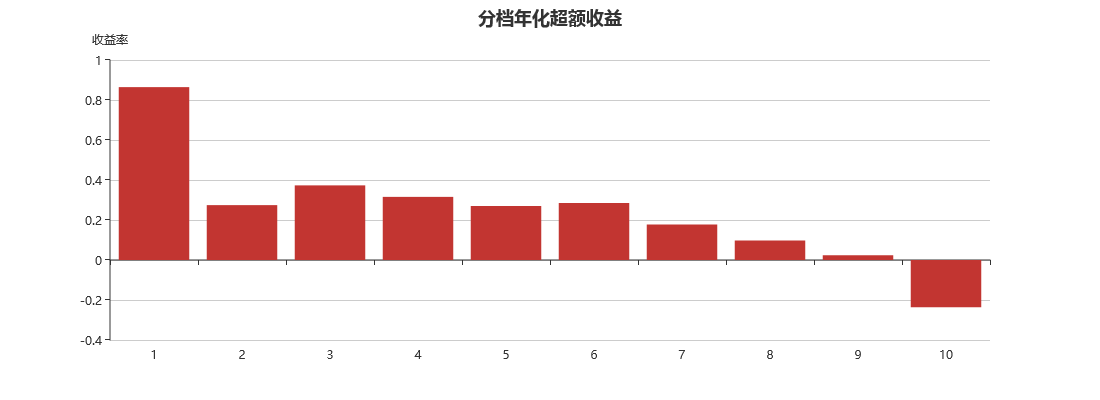


调仓周期太短，收益降低了。

|  |
| --- |
| #013 |

收盘价与交易量的秩相关系数的排序。相关性越负，因子越大，收益越高。涨停股该因子值较大：涨停时，收盘价上升为涨停价是，交易量一般都会降低，从而使得相关性为负。所以应该考虑剔除涨跌停股票。

下图是该因子分十组的收益表现。不管是不剔除涨跌停和st股票，还是仅剔除涨跌停，还是涨跌停和st股票都剔除，各组收益率表现模式都与下图类似。在周期选择上，1日、5日、10日区别不大，周期短一点收益稍微高些；1月为周期，前三组收益基本持平。

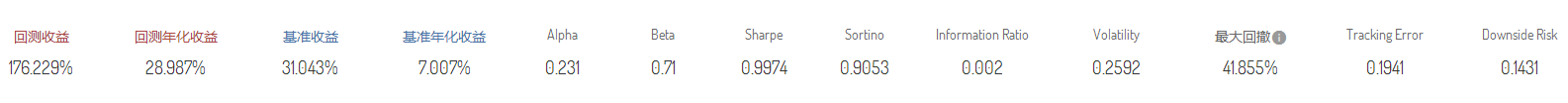


该因子单调性不甚好，第一组收益远大于其他组。该因子可以用来选出因子最大的一些股票来持有。

使用云宽客平台，选择因子最大的200只股票，调仓周期为20天，平均分配资金，无止损，回测结果如下：



使用ricequant平台，选择因子最大的100只股票，调仓周期为10天，平均分配资金，10%个股止损，回测结果如下：

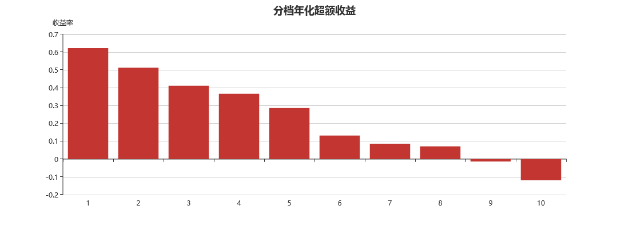
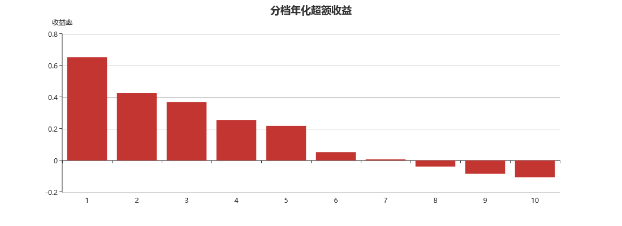


Alpha确实显著存在，但夏普比率不高，回撤太大。15年那一波是硬伤，逃不掉。

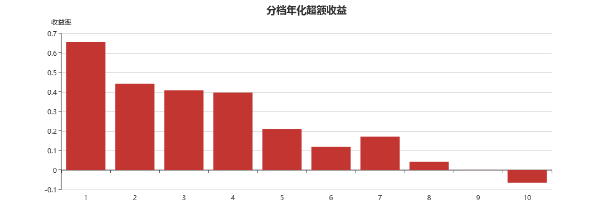
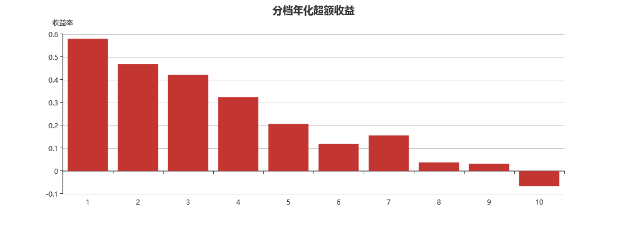
|  |
| --- |
| #015 |

该因子与#13号因子类似，都是价与量的秩相关系数的排序，相关系数前3日排序和越小，因子越大。

该因子1日和5日周期各组收益情况如下，单调性都非常好。（左：1日，右：5日）



仅剔除涨跌停股票，和剔除涨跌停以及st股票后，各组收益情况如下：（左：剔除涨跌停，右：剔除涨跌停和st）



单调性稍微有点紊乱，大致趋势还是有的。

使用云宽客平台，选择因子最大的100只股票，调仓周期为20天，平均分配资金，无止损，回测结果如下：



下面是因子最大的100只股，调仓周期10天的回测结果。

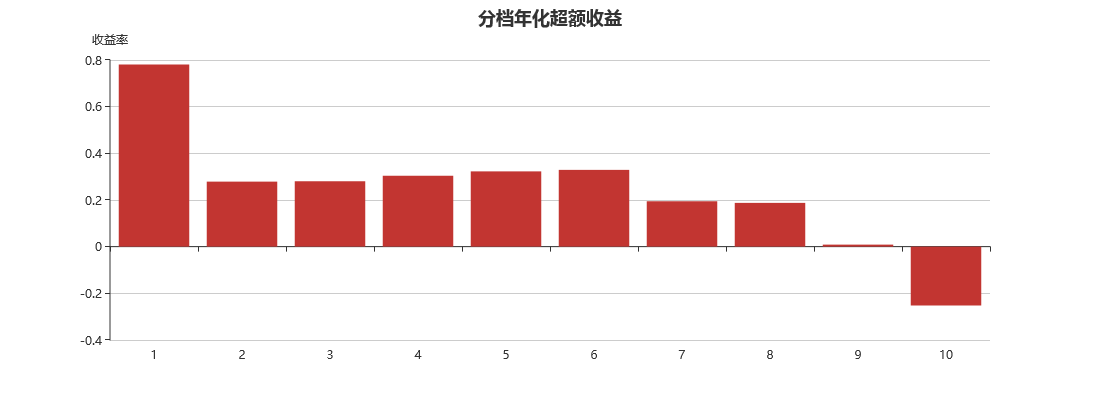


总的来说，该策略收益不太高。不过由于该因子单调性较好，或许可以做多空策略。

|  |
| --- |
| #016 |

这个因子不说了，与#13号因子极其相似。

5日周期，分组收益如下：第一组收益明显高于其他组。



将涨跌停股票和st股票剔除后，各组收益模式依然如此。10日及更长的周期效果不好。

选因子最大的100只股，调仓周期为5日，回测结果如下：



总的来说，效果不太好，交易费用吃掉了很多收益。试试10日为调仓周期：



20日一调仓：



|  |
| --- |
| #041 |

是最当日最高价与最低价的几何平均，是当日的交易量加权平均价格。因子分组检验发现，该因子与后期收益率呈负相关关系。表明当价格越高于最高价与最低价的平均时（因子负的越多），后期越有可能涨。

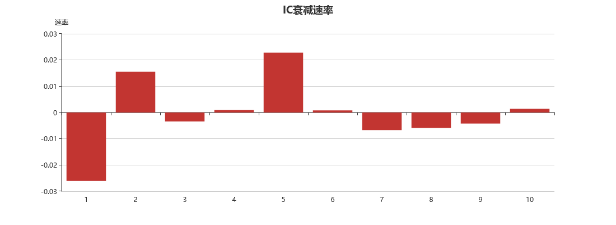
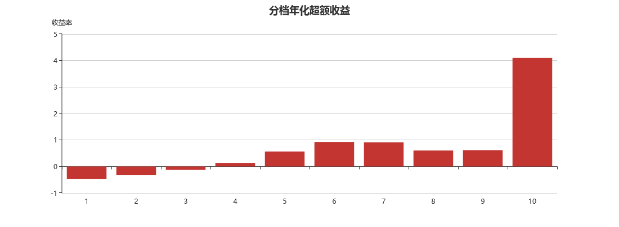
从数据分析中可以看出，该因子最小的都是些涨停的股票，开盘后，股价一路上升，然后封停。这样的股票当天的最高价是涨停价，最低价时开盘价或是比开盘价低，从而最大是开盘价和最高价的中间，当价格涨到接近涨停线时不断震荡，会使得vwap较大，这样的走势计算出的因子是最小的。还有另外一种情况，前期高位震荡，收盘急速拉低。盘中深V也会有较低的因子。

另外，因子为0的股票也很特殊。因子为0，因为这几乎意味着high=low=vwap，全天就一个价格，不是开盘涨停就是开盘跌停。当然，也有可能时high与low的均值与vwap恰好相等，但这几率较小。

该因子是经过国外股市数据验证过，用A股数据计算因子时，应该对涨停跌停股票做一些处理。

10组1日（20130101-20161231，中证500）

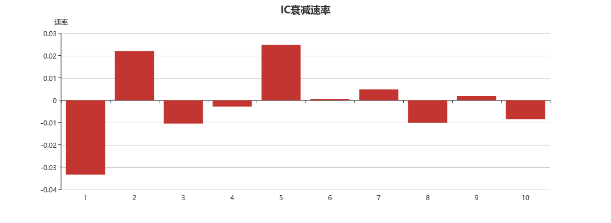
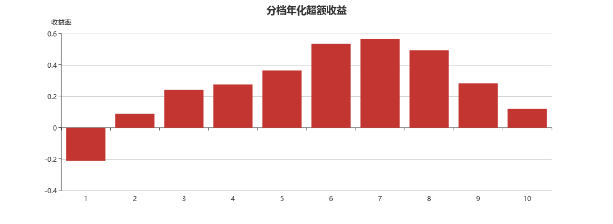




各组有一定的单调性，但第10组与其他组有巨大的区别。IC方面，整体IC为-0.026，第10组的IC 为-0.046，说明在第10组内不同因子的股票其后期收益率还有很大差别。IC衰减上看，该因子仅与后一日收益率呈负相关关系，与后面第二日收益率则是呈反转的正向关系，说明该因子所选股票只能持有一天。但为了以防万一，还是看看5日周期的效果。

10组5日（20130101-20161231，中证500）





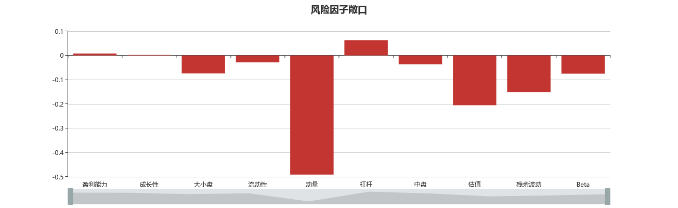
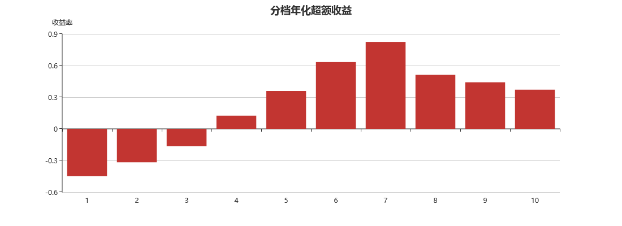
这就比较奇怪了，以5日为周期之后，单调性不复存在，各组收益率呈现出一种倒U型关系，7组收益率最高，如果第7组的因子值平均在0左右，可以把该因子取个绝对值试试，看看是否价格越接近于最高价与最低价的平均的股票，后期收益率越高，但这个要怎么解释？前4组IC为正，后几组IC为负，这难道意味着前4组动量，后6组反转？第10组的IC依然很高，可以就第10组仔细研究下。

当周期调整为10天甚至更长，各组之间收益率则无差别。

接下来我把涨跌停的股票的因子设为空，再来分析因子效果。

10组1日（20130101-20161231，中证500）





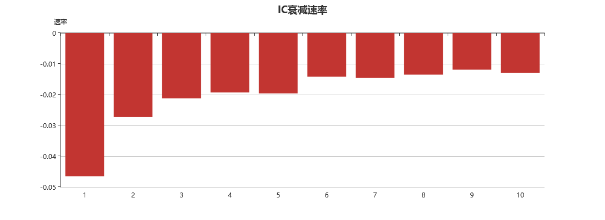
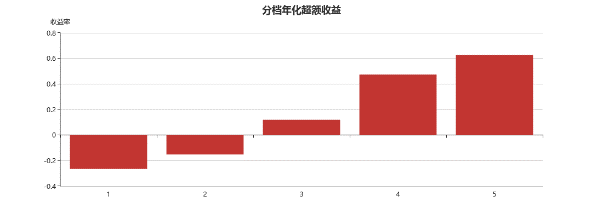
可以看出，即使是一天一调整，第10组的收益率也不再是最高的了。5日周期结果差不多，第10组的收益率进而降为负的了。

可是为什么第7组的收益率会是最高的？我计算了所有股票该因子的平均值，-0.0148，第7组的因子平均值为-0.0252，第6组的因子平均值为-0.0120。大胆猜想，股票因子接近市场平均值时，后期收益率更高。将每日因子都减去当日所有股票该因子的平均值，再取绝对值，然后看因子最小的股票收益率是否最高。

分10组，第8和9和10组收益率差别不大，影响单调性，所以分5组。

5组1日（20130101-20161231，中证500）





单调性比较明显，IC依次降低。但整体IC只有不到0.05，不是很理想，但相比之前，已经提升不少。

选因子最小的100只股，20日一调仓，结果如下：



收益率不错，但回撤和波动率太高，导致sharpe比率和calmar比率不高。看看10日调仓周期的结果：

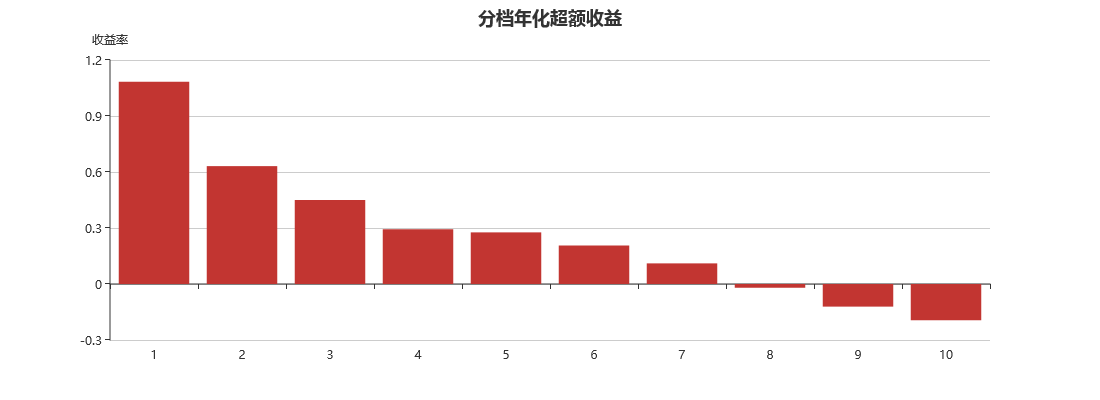


收益降低了，看来还是20日比较靠谱。

|  |
| --- |
| #044 |

依然是价与量的相关性因子。

5日周期各组收益情况如下：



10日周期单调性依然明显，各组收益差距略有收窄。更长周期效果不好。

10日调仓，选择因子最大的100只股，回测收益情况如下：



20日为周期，回测情况如下：



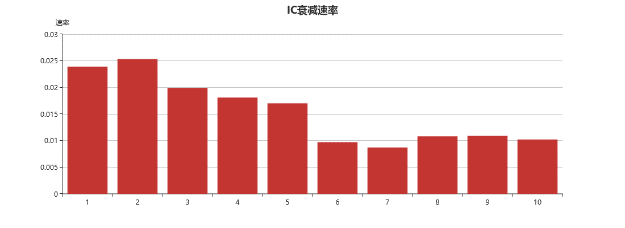
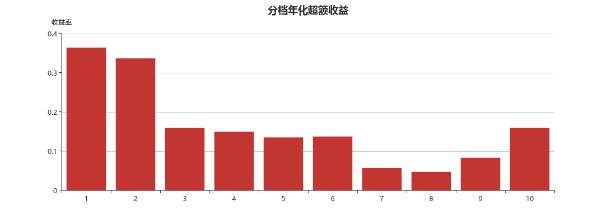
无甚变化。

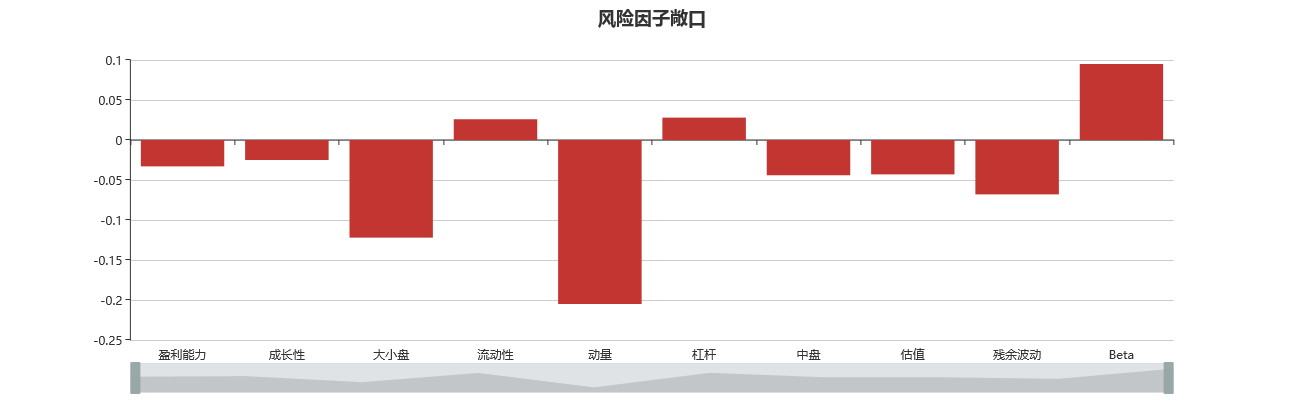
|  |
| --- |
| #055 |

是当日收盘价处于前12日最高最低价之间的什么位置，越靠近最高价，值越大，反之则越小。每日计算该值的排名，以及当日交易量的排名，然后计算前6日这两个指标的相关性。若相关性为负，表示收盘价靠近前12日最高价时，成交量却在减少，而收盘价靠近12日最低价时，成交量却在增加。分组检验发现，这两个排名具有负相关性的股票，后期收益较高。由于该因子是该相关系数乘以-1，所以因子越大，股票收益越高。

10组1日（20130101-20161231，中证500）



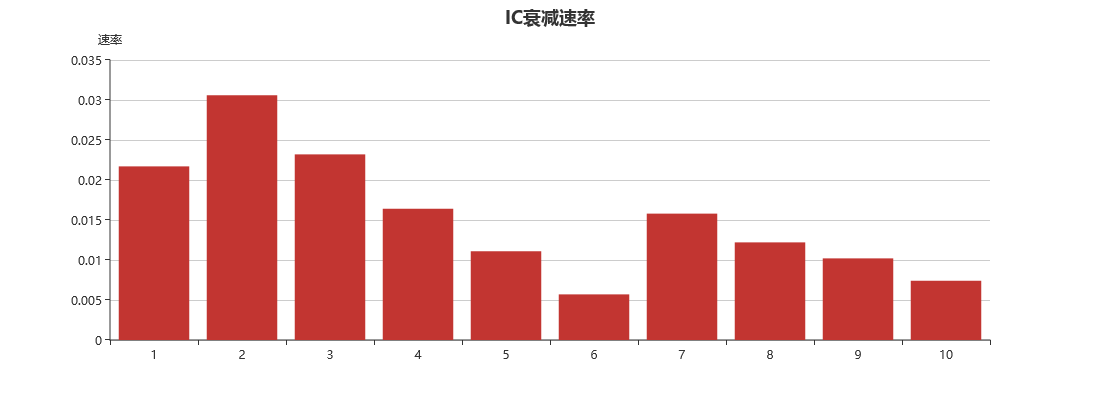
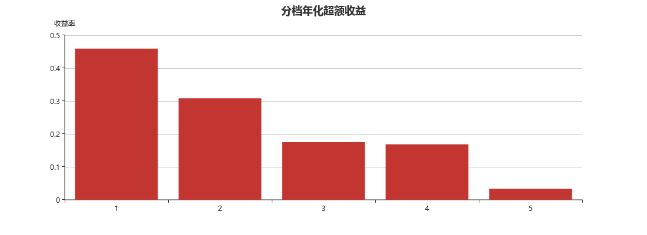




该因子分10组，基本上呈四个类型：1和2，3/4/5/6，7/8，9/10。IC衰减比较慢，该因子动量敞口较大。下面分5组检验。

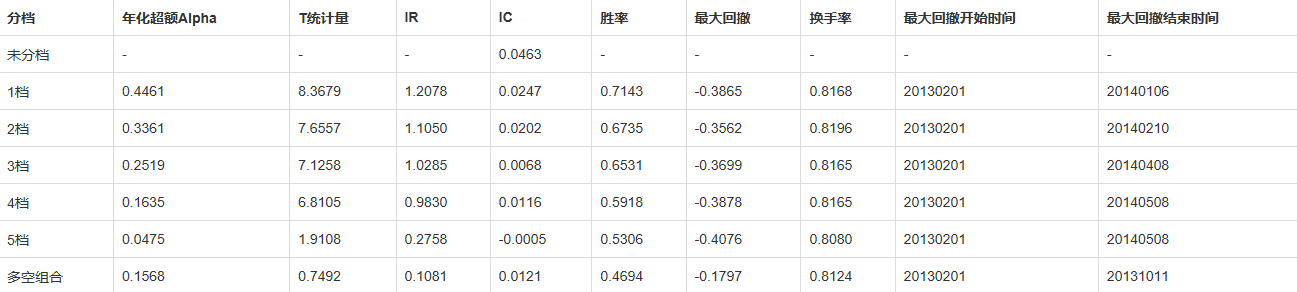
5组5日（20130101-20161231，中证500）

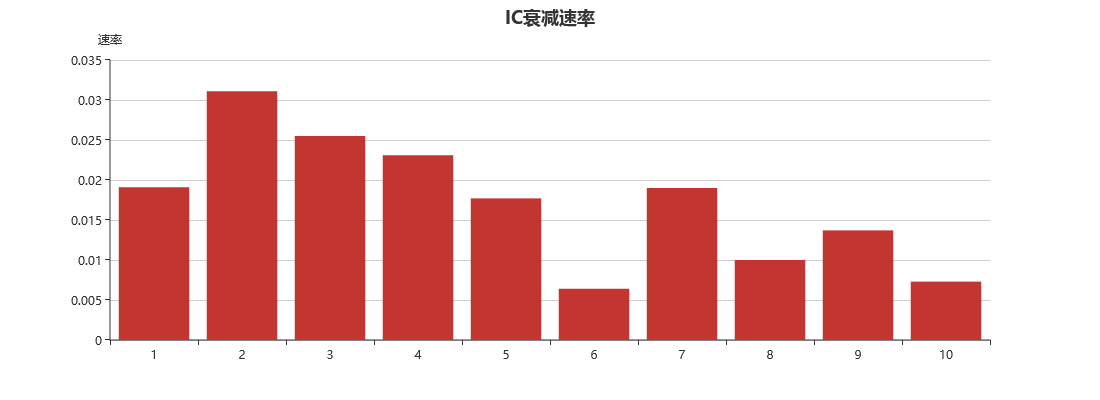
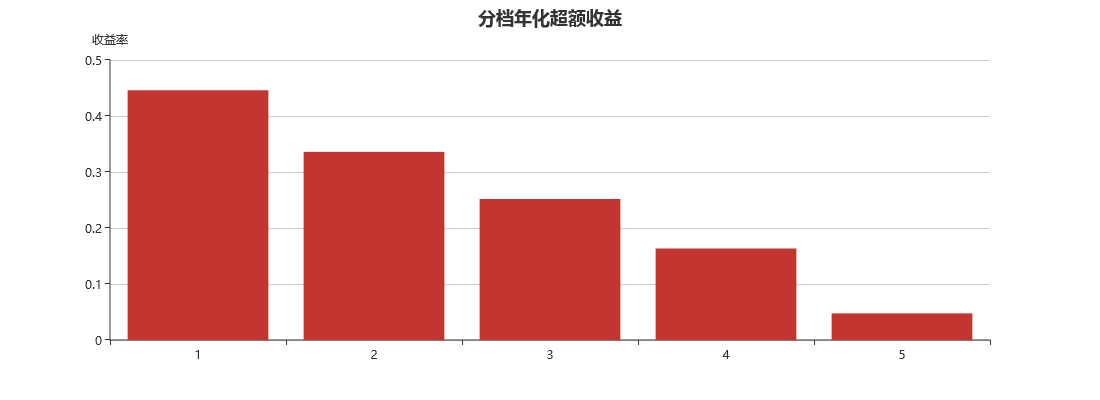




这下单调性好了，各组之间区分度也有了，可是这IC衰减速度还是这般无解。

5组20日（20130101-20161231，中证500）





20日周期，收益率没有太大变化，但最大回撤降低了，好事。

20日调仓周期，选因子最大的100只股：

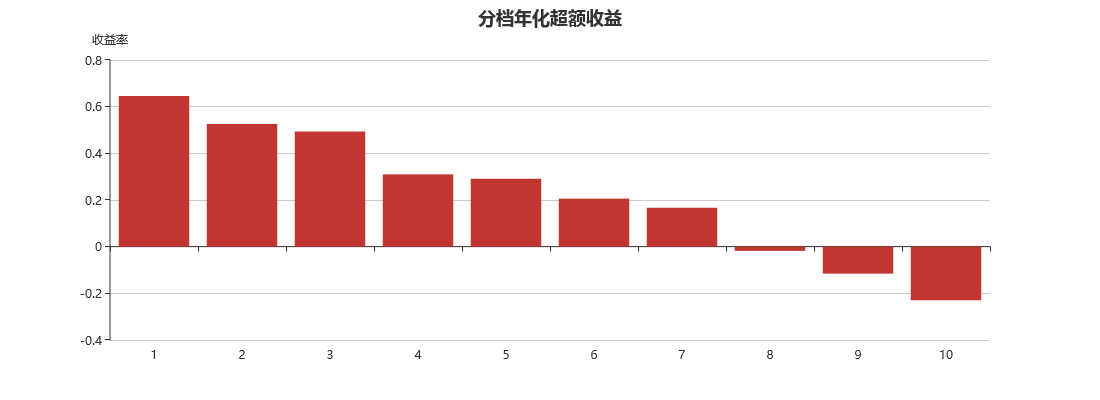


|  |
| --- |
| #083 |

额。。。这个因子看起来有点复杂。分子第二部分是两次rank，是多余的，一次rank就够了。分子分母有一个相同的部分：

这是最高价与最低价的差额占5日平均收盘价的比例。至于该因子具体现实含义，看不透！

10日周期分组收益情况如下：

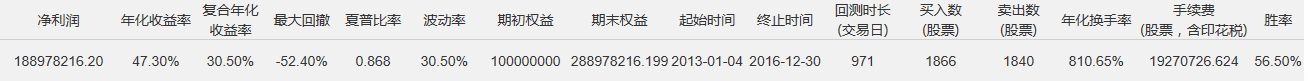


1日，5日单调性相似。该因子单调性还是非常好的。

选因子最大的100只股票，10日的调仓周期，回测结果如下：



情况并不乐观。然后是20日周期的结果：

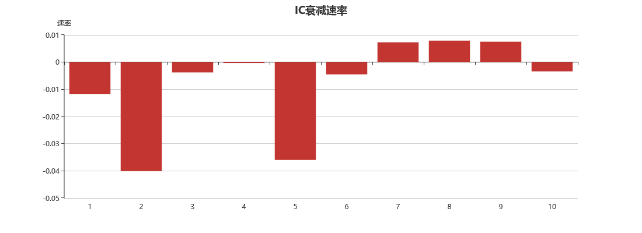
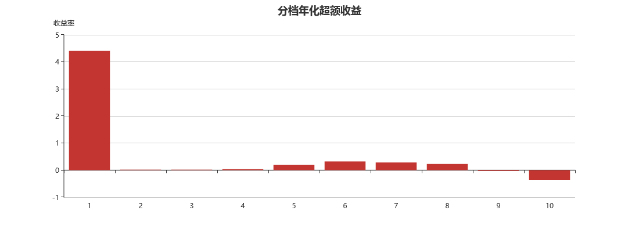


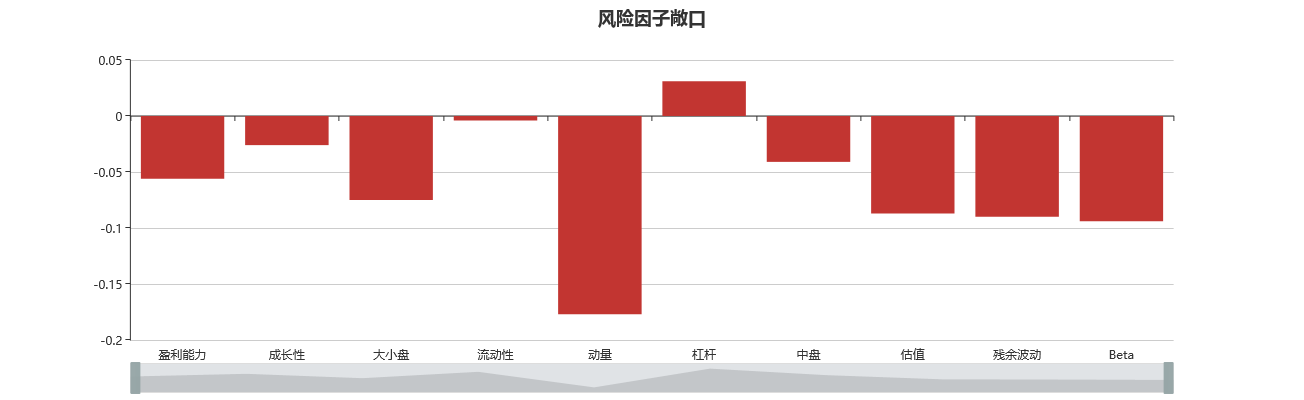
|  |
| --- |
| #101 |

这个因子有点怪异，样子上。分母后面加的0.001，应该是防止high=low的情况下，分母为零，无法相除。不考虑那个0.001，这就是个简单的蜡烛线实体占总体的比例。

10组1日（20130101-20161231，中证500）



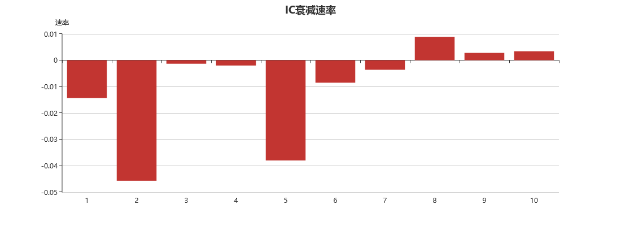
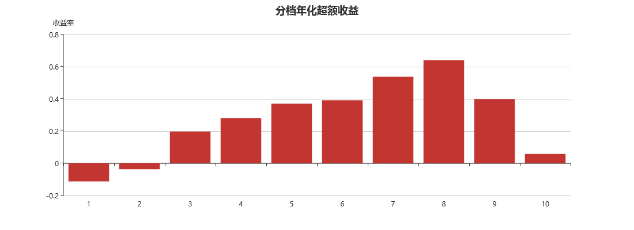




这里应该是有问题的，该因子最大的股票应该是那些当天直接拉涨停的股票，后续一般也会有很大的涨幅，但这样的股票是买不进去的，所以第一组的巨额收益是不具有参考性的；风险因子敞口上，动量达到了15%，这也印证了这一点。并且后面几组的单调性似乎也是乱的。从IC上看，第1组合第10组的IC都有0.03左右，可以多这两组进行分析，剔除掉涨停跌停股票，该因子应该还是有效的。

10组5日（20130101-20161231，中证500）

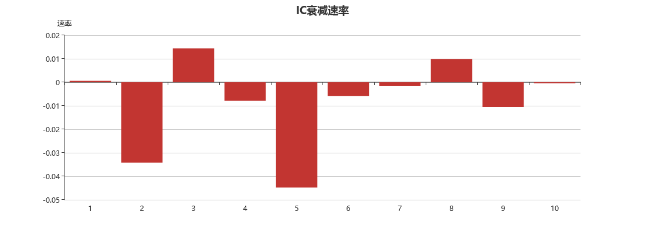
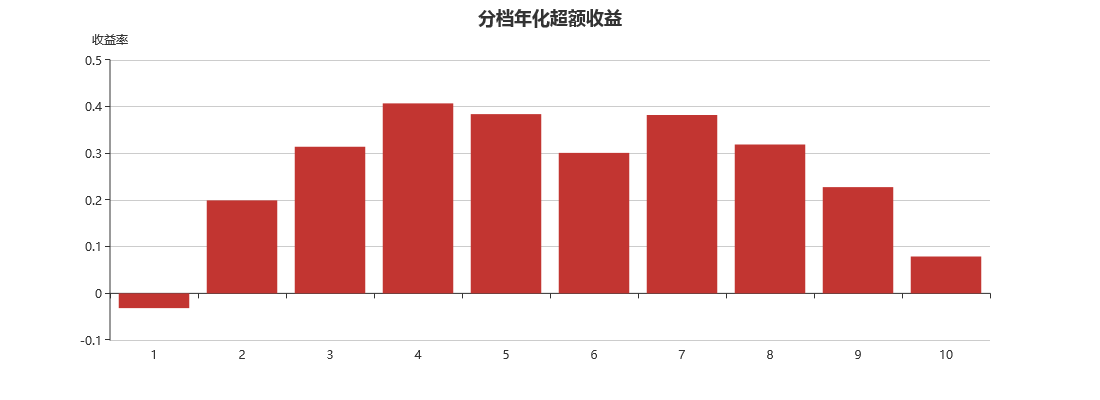




以5日为周期，这就比较有意思了。第一组居然是收益率最低的，收益率最高的是第8组。好想知道第8组的因子是个什么样的特征！

10组20日（20130101-20161231，中证500）

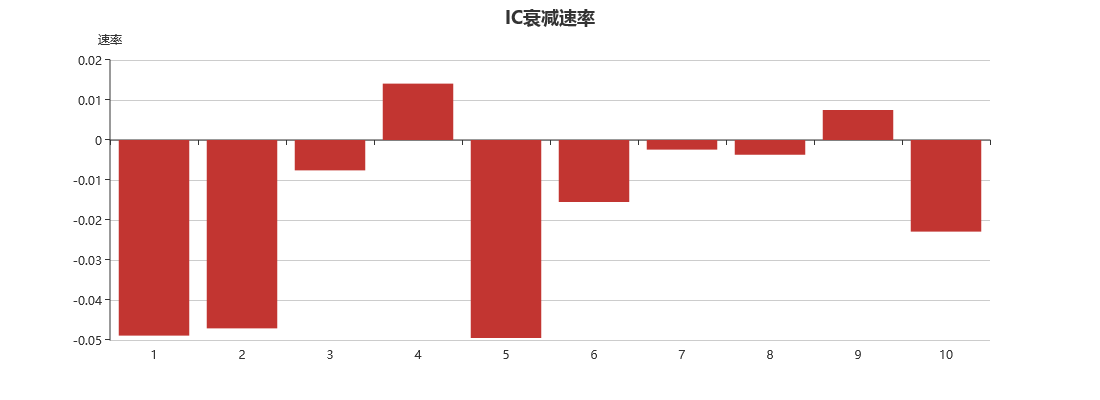
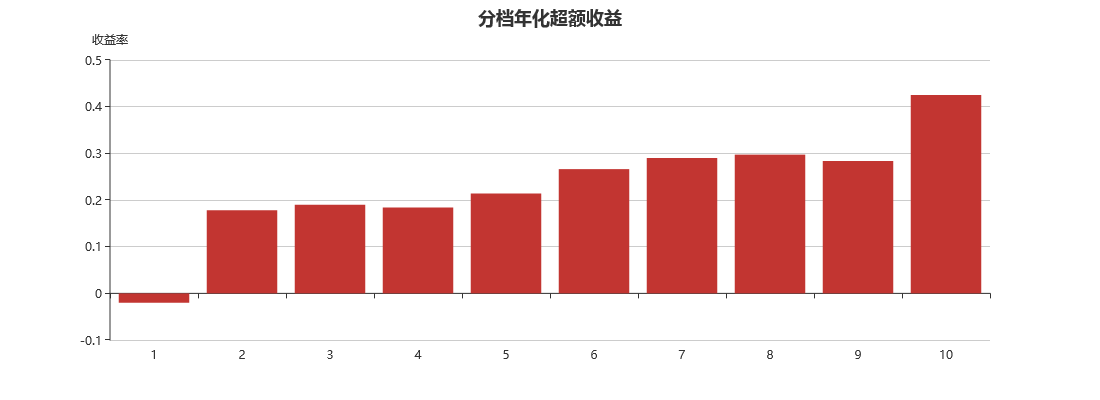




怪异！说是倒U型，中间有个凹。还得去看看每组的因子特征是怎样的。

10组1月（20130101-20161231，中证500）





不解了，20日与1月分明差不多，可这结果差别简直完全不一样啊。

以上因子的相关性如下：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 002 | 013 | 015 | 016 | 041 | 044 | 055 | 083 |
| 002 | 1.0000 | 0.3335 | 0.1526 | 0.1438 | -0.0441 | 0.1952 | 0.4148 | 0.0292 |
| 013 | 0.3335 | 1.0000 | 0.5126 | 0.5381 | -0.0994 | 0.4843 | 0.4845 | -0.0019 |
| 015 | 0.1526 | 0.5126 | 1.0000 | 0.3414 | -0.0318 | 0.4674 | 0.2842 | 0.0193 |
| 016 | 0.1438 | 0.5381 | 0.3414 | 1.0000 | -0.1353 | 0.3458 | 0.2483 | -0.0148 |
| 041 | -0.0441 | -0.0994 | -0.0318 | -0.1353 | 1.0000 | 0.0041 | -0.0314 | 0.0461 |
| 044 | 0.1952 | 0.4843 | 0.4674 | 0.3458 | 0.0041 | 1.0000 | 0.3523 | 0.0269 |
| 055 | 0.4148 | 0.4845 | 0.2842 | 0.2483 | -0.0314 | 0.3523 | 1.0000 | 0.0372 |
| 083 | 0.0292 | -0.0019 | 0.0193 | -0.0148 | 0.0461 | 0.0269 | 0.0372 | 1.0000 |

041号因子与083号因子与其他因子的相关性较低。

41号因子与83号因子组成的多因子选股策略，回测结果如下（调仓周期为20日，选择合成因子最大的100只股）

