

蜗牛股票量化分析软件使用帮助(4.3.0.6)

蜗牛股票量化分析软件是一款专业的股票量化分析软件，软件支持基于日线、周线、半月线、月线以及分时、LEVEL2数据分析，支持单策略，策略批处理，另外还支持竞价选股。以下简称**蜗牛股票量化分析软件**为蜗牛量化

软件功能配置和收费

当前，蜗牛量化按功能以及模块销售，如果发现软件缺少文档中的某些功能，则说明软件不支持该模块，非软件设计问题

| 模块 | 收费 | 购买链接 | 说明 |
|---------------|--------------|---|--|
| 蜗牛量化基础版 | 免费 | N/A | 功能有限，软件可在 http://http://snail007.com/ 下载 |
| 基础收费版 | 960元 | http://snail007.com/buy | 支持支持函数调用；支持变量标记；支持=, >=以及<=布尔表达;支持动态股票池分析模式；支持通达信数据源 |
| 竞价选股 | 480元 | 同上 | 支持竞价选股功能 |
| 软件自动更新 | 480元 | 同上 | 支持股票(480日)日线数据更新，可选用通达信数据(免费) |
| 软件自动更新功能 | 240元 | 同上 | 支持股票分时数据更新 |
| Sea脚本扩展 | 360元 | 同上 | 支持Sea脚本扩展，可定义常用策略组，用于提高量化效率 |
| 策略批处理 | 480元 | 同上 | 支持策略批处理，一次执行可执行多个策略，本模块适合量化老鸟 |
| 周线/半月线/月线量化分析 | 960元 | 同上 | 支持基于周线，半月线，月线量化分析 |
| 参考选股 | 960元 | 同上 | 支持基于指数配合股票或股票配合股票模式的量化分析功能 |
| 分时量化支持 | 3600元 | 同上 | 支持股票分分钟级别的量化T+1分析，指令元以tick开头，需要分时数据支持 |
| LEVEL2单股资金流查看 | 1800元 | 同上 | 支持查看单股LEVEL2资金流，需要LEVEL2数据支持 |
| LEVEL2量化指令支持看 | 9600元 | 同上 | 支持基于LEVEL2的资金流量化指令 |
| Boosting策略组 | 12800 | 同上 | 支持基于boosting的人工智能策略组 |
| 策略定时执行 | 480元 | 同上 | 支持收盘后定点执行策略 |
| 配对交易 | 480元 | 同上 | 支持股票配对交易量化分析功能 |
| 双曲线量化模式 | 960元 | 同上 | 支持基于双曲线量化模式，当前仅支持弹簧均线 |
| 软件定制 | 3000元起步 | 同上 | 以需求难易程度定 |
| 策略出售 | 2400/3600元一套 | 同上 | 事先提供盈亏图,3600元为参考策略，需要参考量化功能支持 |

目前，软件售后提供免费的一年的同版本更新服务；软件当前不提供注册码，只提供单用户对应的**单编译版本**，软件仅供购买者本人使用。如果发现盗用情况，则不再提供后续服务支持，解释权在蜗牛量化软件提供商。本说明文档包含了蜗牛量化各个分支版本的所有功能，实际功能以购买时确定功能为准。

量化工作机制

蜗牛量化在执行过程中，针对一个股票，量化游标会从股票日线数据起始位置开始向后逐一搜索，在执行的某一节点，称当期游标位置对应日期为**量化日**。每个量化指令的最后一个参数即为量化计算日相对于量化日的日期偏移。该值必须小于或者等于0。例如：

```
buy.close,0>ma,5,0
```

含义:量化日收盘价格超过当日5日均线则构成一个买点

```
buy.close,-1>ma,5,-1
```

含义:量化前一日满足收盘价超过当日5日均线则构成一个买点

软件量化配置项

标准软件指令分为3个类型，分别是配置项，买点以及卖点。配置项用于软件和量化参数的配置；买点和卖点用于设定股票买入以及卖出条件。对多个买点，同时满足则为策略买入条件；对于多个卖点，满足一个则为卖出条件

注释

策略脚本行以#标记为注释，例子如下:

```
#蜗牛股票量化分析软件是一款股民专用的策略分析软件
```

```
#本软件主要用户炒股策略的量化测定
```

配置

```
config.source.k.dir=data
```

设置软件目录下的data文件夹为股票日线文件夹

```
config.source.ticket.dir=ticket
```

设置软件目录下的ticket文件夹为股票分时数据文件夹

```
config.source.lv2.dir=lv2
```

设置软件LEVEL2数据文件夹

```
config.source.ticket.maxtickloaded=8
```

设置软件分时数据单次最大加载文件数，默认为5

```
config.source.k.format=default
```

设置软件日线数据格式为默认模式

```
config.source.ref.index=sh000300
```

指定沪深300为参考指数

```
config.source.market=China
```

指定量化股票市场（可指定为US，HK，futures）

```
config.candlestick.theme=
```

指定盈亏图，蜡烛图风格，当前支持八种风格，分别是

ChartThemeLight，ChartThemeBlueCerulean，ChartThemeDark，ChartThemeBrownSand，ChartThemeBlueNcs，ChartThemeHiq

```
config.source.ref.code=sz000985
```

设定sz000985为参考量化股票(仅支持参考量化功能支持)

```
config.trade.buy.order=volasc
```

如果单日有多个股票同时满足买入条件，则以量化前一日成交量降序排列为买入次序，可选模式：
voldesc,volasc,inreasedesc,increaseasc，含义分别为成交量降序，成交量升序，涨幅降序，涨幅升序

```
config.trade.sale.point=close
```

设定股票卖出时机为尾盘，可选时机为open(早盘)

```
config.trade.sale.rapid=yes
```

是否使用立即止盈模式，达到指定盈利则立刻卖出

```
config.trade.max.hold.day=3
```

设定最大股票持有日期为3日，昨日买入，今日卖出为2日

```
config.trade.min.exclude=6
```

设定股票卖出后禁止买入最小日期间隔

```
config.trade.max.profit=64.0
```

设定持股最大盈利卖出点数，如果持股盈利达到该点数则以config.trade.sale.point设定时间卖出股票

```
config.trade.max.lost=-9.0
```

设定单股止损点，达到则以config.trade.sale.point设定时间卖出

```
config.trade.max.count=3
```

设定单日最大可买入股票数目

```
config.trade.max.allow.increase=3.2  
config.trade.max.allow.decrease=-3.2
```

如果股票满足买入条件，第二日开盘股票开盘涨跌在config.trade.max.allow.increase和
config.trade.max.allow.decrease之间则为实买入股票，否则不作为买入股票计入统计

```
config.trade.tradecost=0.8
```

设定股票买入卖出交易成本，单位为每千(单项)在卖出日一次计入

```
config.source.lv2.v1=5.0
```

设置LEVEL2资金流散户资金单笔上限为5万元

```
config.source.lv2.v2=50.0
```

设置LEVEL2资金流中单单笔上限为50万

```
config.source.lv2.v3=100.0
```

设置LEVEL2资金流大单上限为100万元，之上则为超大买卖单

```
config.source.from=1980-01-01
config.source.to=2018-12-31
```

设定股票回测日期区间

```
config.trade.max.subnewstock=60
```

针对单股仅测试前60日内数据（即次新股模式）

```
config.trade.启用每日累加收益=?
```

指定盈亏统计算法，可选值为yes，no；默认为no，如为yes，统计盈亏时则把单日持股涨幅累加计算；如为no，则把单日持股涨幅平均作为单日统计涨幅；可以理解设置为yes，会放大盈亏，视为带有杠杆性质。

```
config.source.minactive=0
config.source.maxactive=100
```

设置股价可接受活跃度区间[0,100]，一个股票只有一个活跃度，值越大说明股票越活跃。注意，此指令带有未来函数

```
config.runningMode=回测
```

设定股票运行模式，可选回测，选股基于配对交易，默认为回测

```
config.source.pool=6000**
config.source.pool=000***
config.source.pool=00**01
config.source.pool=0009**
config.source.pool=300506,000056
config.source.pool=随机50
config.source.pool=我的股票池
```

设定回测股票池，如果参数值为随机50，随机100，随机300之一，则实际股票池为随机选出的50，100或者300个股票；如果为**我的股票池**，则之前需要在菜单**配置-股票池**中自定义股票池清单；如果使用类似000***等包含通配符的模式，则股票池为参考股票代码以000开头的股票列表；如果股票池为类似300506,000056模式则以给定股票列表为准
NOTE：免费版本不支持通配符类型的股票池

```
config.trade.max.subnewstock=60
```

定义次新股，当前软件无法区分次新股，可以借助本参数模式次新股选股，参数值60含义为股票日线数据前60个日为可能次新股

```
config.trade.buy.movepoint=1.0
```

设定股票买入滑动涨跌幅，含义为以第二日开盘涨跌上浮给定涨幅为买入点，如果该涨幅无法买入，则指定股票不纳入统计；可选范围为[-2.0 - 2.0]，默认为0.0

买卖点

软件买点以buy.开头，卖点以sale.开头

指令元

本软件量化指令元可分为2类，一类是布尔指令元，一类是数值指令元，前者可作为独立的策略语句，后者则需要使用判断语句组合为独立的策略语句

一个简单的例子如下：

```
buy.close,0>ma,5,0
```

```
buy.sea.大阳线,0
```

在这个例子中

```
close,0  
ma,5,0  
sea.大阳线,0  
即为指令元
```

下面逐一介绍指令元

OHLC指令元

OHLC指令元系列用于获取股票的open/high/low/close以及vol等基础数据

```
ohlc.open,offset
```

获取股票开盘价，offset为量化偏移(offset含义下同不再赘述)

```
ohlc.high,offset
```

获取股票最高价

```
ohlc.low,offset
```

获取股票最低价

```
ohlc.close,offset
```

获取股票收盘价

```
ohlc.increase,offset
```

获取股票涨幅

```
ohlc.increase,flag,offset
```

获取股票给定类型的涨幅

flag=1，结果为开盘价相对于昨日涨幅

flag=2，结果为最高价相对于昨日涨幅

flag=3，结果为最低价相对于昨日涨幅

```
ohlc.swing,offset
```

获取股票振幅，计算公式为 $swing=(high-low)/low*100.0$

```
ohlc.swing,flag,offset
```

flag=1，计算最高价相对于开盘价，收盘价平均的涨幅

flag=2，计算股票开盘价，收盘价平均相对于最低价的涨幅

flag=3，计算股票收盘价相对于开盘价涨幅

flag=4，计算股票开盘价相对于最低价的涨幅

flag=5，计算股票最高价相对于收盘价涨幅

```
ohlc.vol,flag
```

获取股票当日成交量

```
ohlcv.总市值, flag
```

获取股票总市值（单位：万元）

```
ohlcv.换手率, flag
```

获取股票换手率数据，注意事项，此处换手率是根据股票最新流通市值和量化日成交量数据换算出的，如果中间有股票解禁则可能会有偏差，如果需要精确的换手率数据请使用tick.turnover指令

布林轨道

布林轨道指令元用于计算布林轨道参数

```
boll.upper, offset  
boll.mid, offset  
boll.lower, offset  
boll.width, offset
```

以上指令分别用于获取布林通道(20,2)的上轨，中轨，下轨和轨道宽度($\text{boll.width} = \text{boll.upper} - \text{boll.mid}$)

对于收费版本，可以使用以下格式的带参数布林指标，如下：

```
boll.upper, n, m, offset  
boll.mid, n, m, offset  
boll.width, n, m, offset  
其中  $n$  为布林指标中的取点数， $m$  为布林带宽度系数， $\text{boll.width}$  计算公式为  $(\text{boll.upper} - \text{boll.mid}) / \text{boll.mid} * 100.0$ 
```

MA/EMA/VOLMA指令元系列

ma指令用于计算股票简单移动均线值

```
ma, n, offset
```

获取股票n日均线值

ema指令用于计算股票指数移动均线值

```
ema, n, offset
```

获取股票n日加权平均值

volma指令用于计算股票简单移动均线值

```
volma, n, offset
```

获取股票n日成交量平均值

MACD指令元

```
macd.diff, offset  
macd.bar, offset  
macd.dea, offset
```

获取股票MACD的diff,bar和dea值，默认参数为12,26,9

对于收费版本，可以自定义MACD参数，格式如下

```
macd.diff,a,b,c,offset  
macd.bar,a,b,c,offset  
macd.dea,a,b,c,offset
```

KDJ指令元

```
kdj.k,offset  
kdj.d,offset  
kdj.j,offset
```

获取股票日线KDJ的k, d, j值，默认参数为9,3,3

DMI指令元

```
dmi.pdi,n,m,offset  
dmi.mdi,n,m,offset  
dmi.adx,n,m,offset  
dmi.adxr,n,m,offset
```

获取过DMI指标的pdi, mdi, adx和adxr值，指定参数为n,m,默认为14,6

其他指令元

```
psy,n,offset
```

获取给定周期n的心理线psy指标值

```
bias,n,offset
```

获取给定周期n的乖离率bias指标值

```
wr,n,offset
```

获取给定周期n的威廉wr指标值

```
ad,offset
```

获取给定的积累派发线指标ad值

```
pvi,offset
```

获取正成交量指标pvi值

```
bop,offset
```

获取力量平衡指标bop值

```
obv,offset
```

获取能量潮指标obv值

```
wad,offset
```

获取威廉多空力度线wad值

```
tr,offset
```

获取股价真实波动值,如果需要atr值,请参考如下代码

```
@tr=tr,0
```

```
@atr=ma[@tr,value,14.0]
```

```
value,data
```

构造一个值为data的常量,例如:

```
buy.increase,0>value,4.0
```

含义是若量化日股票涨幅超过4.0%则为买入条件

分时指令

```
tick.形态良好,score,offset
```

形态良好指令用于评估指定时间分时线走势是否为形态良好, score为形态的分,取值为[0,100]。需要注意的是形态良好估分必须大于60,否则会预编译错误,如果当日分时得分大于等于指定值,则为真,否则为假

举例,信立泰2018年3月6日分时走势相当良好,图例如下:



```
tick.形态恶化,score,offset
```

形态恶化用于评估指定股票给定日期分时走势是否为恶化形态,所谓的恶化形态就是股价持续走低,且均线不断下移。举例,伊利股份2018年3月7日走势即为恶化形态,图例如下:



```
tick.早盘涨停,length,offset
```

早盘涨停用于判断指定股票分时是否为早盘涨停,参数length用于指定早盘区间,取值为[0,60),时间单位为分钟,如果股票在该时间段内涨停,则该指令为真;需要注意的是在调用该指令前需要先使用k.涨停指令判断为涨停,否则会出现无

法预期的结果。示例如下：

```
buy.k.涨停,-1  
buy.tick.早盘涨停,30,-1
```

```
tick.尾盘涨停,length,offset
```

尾盘涨停用于判断股票是否为指定时间段的尾盘涨停
例如：

```
k.涨停,-1  
tick.尾盘涨停,30,-1  
含义是判断股票是否为尾盘30分钟内涨停
```

```
tick.尾盘拉升,length,increase,offset
```

尾盘拉升判断股票是不是在尾盘length分钟内拉升increase幅度以上

```
tick.区间成交量,start,length,offset
```

计算股票在给定时间段的成交量，单位-手
例如：

```
tick.区间成交量,0,60,0  
判断股票在开盘60分钟内成交量
```

```
tick.分时最大成交量,offset
```

获取股票1分钟内最大成交量

```
tick.最高价区间,start,length,offset  
tick.最低价区间,start,length,offset
```

判断日内分时最高价，最低价对应时间是否在指定时间段内

```
tick.成交量,offset  
tick.成交额,offset  
tick.换手率,offset  
tick.外盘,offset  
tick.内盘,offset  
tick.总市值,offset  
tick.流通市值,offset  
tick.市净率,offset  
tick.市盈率,offset  
tick.量比,offset  
tick.买1,offset  
tick.买2,offset  
tick.买3,offset  
tick.买4,offset  
tick.买5,offset  
tick.卖1,offset  
tick.卖2,offset  
tick.卖3,offset  
tick.卖4,offset  
tick.卖5,offset  
tick.open,offset  
tick.close,offset  
tick.high,offset  
tick.low,offset
```

以上指令含义自明

```
tick.正弦十字星,offset  
tick.反正弦十字星,offset
```

这二个指令用于判断当日分时是否为正弦十字星和反正弦十字星；正弦十字星走势类似正弦曲线，先高后低最后收平；反正弦十字星与之相反。调用之前需要先使用k.十字星指令进行校准，用法如下：

```
k.十字星,0  
tick.反正弦十字星,0
```

```
tick.走势僵硬度
```

该指定用于计算股票走势不自然的程度，幅度为0-100，如果值越大则说明走势越不自然，使用前需要排除掉涨停，跌停的情况

以下为K线组合类指令

```
k.上升三法,offset  
k.上缺口,offset  
k.下缺口,offset  
k.下降三法,offset  
k.乌云盖顶,offset  
k.倒锤子线,offset  
k.倾盆大雨,offset  
k.光头,offset  
k.光头光脚阳线,offset  
k.光头光脚阴线,offset  
k.光头阳线,offset  
k.光头阴线,offset  
k.光脚,offset  
k.光脚阳线,offset  
k.光脚阴线,offset  
k.光身,offset  
k.十字星,offset  
k.双飞乌鸦,offset  
k.多方炮,offset  
k.好友反攻,offset  
k.尖三兵,offset  
k.旭日东升,offset  
k.涨停,offset  
k.涨势尽头,offset  
k.清晨之星,offset  
k.看涨孕育线,offset  
k.穿刺,offset  
k.红三兵,offset  
k.跌停,offset  
k.锤子线,offset  
k.阳吞阴,offset  
k.阳抱阳,offset  
k.阴包阴,offset  
k.阴吞阳,offset  
k.黄昏之星,offset  
k.黑三兵,offset  
k.大实体短影,offset
```

大实体短影含义是股票当日实体较大，上下长影都很短的单K线

以上为常见k线形态组合，需要留意的有2个，一个是大部分K线组合没有严格的判断标准，如果对内置k线形态不满足，可以使用前面的指令构建自己的k线组合；二是类似清晨之星，穿刺之类形态，在使用之前需要先判断出目前趋势，应该在使用使用k线组合之前需要使用均线或者MACD等指标判断当前趋势，然后使用方可。

非日线周期指令

除了基本的日线指令，蜗牛量化还提供分时周期和周线，月线的指令。注意分时周期数据都是就基于一分钟周期数据加工

而成，周期内最低价，最高价可能和实际最低价，最高价略有差异；目前蜗牛量化提供5,10,15,30,60分钟周期的指标，例子如下：

```
buy.分时5分钟.ohlc.increase,0>value,0.5  
buy.分时15分钟.macd.diff,0>分时15分钟.macd.dea,0
```

就是在指令元前加入周期标示即可，由于计算是从依据最近5日分时数据，所以针对一些较大周期的迭代计算类指标计算可能不太准(比如macd,ema等，ma, ohlc等不在此列)；另外一些特定的指令元无法使用非日线周期标示，比如分钟5分钟.ohlc.流通市值

大周期指令则有周线，半月线，月线3种，例子如下：

```
buy.周线.ohlc.increase,0<value,3.0  
buy.半月线.ma,5,0>月线.ma,5,0
```

当然对于周线,月线类判断是否涨停，跌停是无意义的

Level2专用指令

```
lv2.散户买单金额,offset  
lv2.散户卖单金额,offset  
lv2.中买单金额,offset  
lv2.中卖单金额,offset  
lv2.大买单金额,offset  
lv2.大卖单金额,offset  
lv2.超大买单金额,offset  
lv2.超大卖单金额,offset  
lv2.中买单数,offset  
lv2.中卖单数,offset  
lv2.大买单数,offset  
lv2.大卖单数,offset  
lv2.超大买单数,offset  
lv2.超大卖单数,offset
```

示例

```
buy.lv2.散户买单金额,0 < lv2.散户卖单金额,0  
量化日散户买入小于散户卖出金额则构成买点
```

周线/半月线/月线指令

除了支持日线分析，蜗牛量化还支持周线，半月线，月线的量化分析，使用的时候只需要在常用指令元加上周线,半月线,月线前缀即可

```
buy.周线.increase,0>value,0
```

含义为如果量化日周线涨幅超过0.0%则构成一个买点

目前不支持周线/半月线/月线的指令元有tick分时指令元，lv2指令元,value指令元等

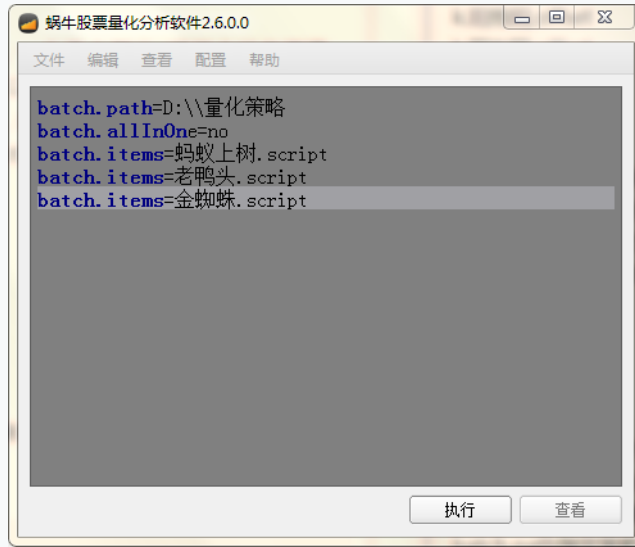
note:注意事项，当前对周线/半月线/月线的计算是依据日线合成的，对于周线是依据量化日向前逐5日计算的，以上述为例，如果量化日为周三，则过去五日周三到上一周四，并非常见软件的连续5个工作日，半月线，月线雷同

批量策略

以下为策略批处理指令，仅支持批量策略分析的软件支持该功能

```
batch.path=策略文件夹  
batch.allInOn=yes/no  
batch.items=策略脚本
```

batch.path指定策略脚本所在文件夹路径
batch.allInOn取值为yes或者no，用于指定策略结果单独显示还是分开显示
batch.items指向具体的策略脚本文件名
示例如下：



函数清单

蜗牛量化当前提供二种类型的函数调用方式，一种是常规类型的plus/minus等函数组；另外一种接近字符串求值的express函数

```
plus[a;b;c]
```

plus函数用于计算a/b/c之和。本函数支持对2个及2个以上数值的求和
示例如下：

```
@var=plus[vol,-1;vol,-2;vol,-3]
buy.@var>vol,0
含义为如果量化日成交量大于前3日成交量之和则设定为买点
```

```
minus[a;b]
```

minus函数计算a-b值

```
multi[a;b]
```

multi函数计算a*b乘积

示例：

```
@var=multi[vol,-1;value,1.5]
buy.@var 目的是计算量化日成交量是否在昨日成交量的1.5倍以上
```

```
div[a;b]
```

div函数用于计算a/b值

```
min[a;b]
max[a;b;c;d]
```

min/max函数用于计算给定值的最小值和最大值，参数可以有多个

```
if[a;b;c]
```

if函数根据a值结果返回b或者c值，其中a表达式必须为布尔类型，示例如下
@value=if[ohlc.open,0>ohlc.close,0;ohlc.open,0;ohlc.close]

```
not[statement]
```

not函数对布尔类型的statement求反，示例如下
@value=not[ohlc.increase,0>value,1.0]

```
ma[statement;n]
```

ma函数用于计算给定statement的n日均值，比如
@ma5 = ma[ohlc.close,5;value,5.0]

```
sma[statement;n;m]
```

sma函数用于计算给定statement的n日移动均值，其中m为权重

```
ema[statement;m]
```

ema函数求取给定statement的加权均值,调用示例如下
@var=ema[ohlc.close,0;value,12.0]

```
stdev[a,n]
```

sd函数用于计算指定a序列的标准差，示例如下
@var=sd[ohlc.close,0;value,20.0]

```
absdev[a,n]
```

sd函数用于计算指定a序列的绝对值标准差，示例如下
@var=sd[ohlc.close,0;value,20.0]

```
variance[a,n]
```

sd函数用于计算指定a序列的方差，示例如下

```
@var=sd[ohlc.close,0;value,20.0]
```

```
minVar[a,n]
```

minVar用于计算指定a序列给定n个值的最小值，示例如下
@var=minVar[ohlc.close,0;value,5.0]
结果则为量化日最近5日收盘最小值

```
maxVar[a,n]
```

maxVar用于计算指定a序列给定n个值的最大值

```
express[...]
```

express为字符串类型的表达式求知，支持的函数有

sum/cos/sin/max/min/mean/stdevp/stdev/open/close/high/low/vol/increase/minOpen/minClose/minHigh/minLow/maxOpen/m
其中open,close,high,low,vol函数分别计算给定量化日的开盘,收盘,最高,最低,成交量值参数为量化日偏移。

increase计算的是a,b中较大值相对较小值的涨幅

使用示例

```
@var=express[(open(0)+close(0)*0.5)]  
@var=express[(open(0)+close(0)/2.0)]
```

计算的是量化日开盘和收盘的中间价,等同于以下表达式

```
@var=express[mean(open(0),close(0))]
```

min/maxOHLC min/maxVOL系列函数用于量化日最近n日给定变量最大或最小值,比如

```
minVol(60,0)
```

结果为量化日最近60日最小成交量

```
@plusVar[a;b]
```

计算a指令的b个序列和

```
prev[a;b]
```

prev指令用于计算相对于量化的前b单位日期的a值,比如

```
@k=prev[ohlc.close,0,value,5.0]
```

@k值为量化日前5日收盘价

竞价选股

以下为竞价选股指令集(当前仅提供2018-06-01日后竞价数据)

```
bid.买1,flag  
含义为获取竞价时买1买单数,下同  
bid.买2,flag  
bid.买3,flag  
bid.买4,flag  
bid.买5,flag  
含义为获取竞价时卖1卖单手数,下同  
bid.卖1,flag  
bid.卖2,flag  
bid.卖3,flag  
bid.卖4,flag  
bid.卖5,flag  
bid.五档买单和,flag  
bid.五档卖单和,flag  
bid.成交量,flag  
获取竞价时成交量  
bid.成交额,flag  
获取竞价时成交额  
bid.价格,flag  
获取竞价是价格,不复权  
bid.涨幅,flag  
获取竞价时涨幅,也就是当日开盘涨幅
```

Per指令集

per指令标记用于获取股票日线的不复权数据，同时可以用于获取量化日全市场前50或100个股票（当前仅提供2018-07-01日后数据）

```
per.换手率,offset
获取给定日期股票的精确换手率数据
per.昨收,offset
per.low,offset
per.high,offset
per.open,offset
per.close,offset
获取不除权的OHLC数据
per.成交额,offset
per.成交量,offset
获取股票交易额和成交量,单位分别是万,手
per.振幅,offset
per.涨幅,offset
获取股票振幅和涨幅
per.流通市值,offset
per.总市值,offset
获取股票市值
per.per,offset
per.pb,offset
获取财务per/pb数据
per.换手率前50,offset
per.换手率前100,offset
per.涨幅前50,offset
per.涨幅前100,offset
per.成交量前50,offset
per.成交量前100,offset
per.成交额前50,offset
per.成交额前100,offset
per.高价前50,offset
per.高价前100,offset
per.低价前50,offset
per.低价前100,offset
per.小市值前50,offset
per.小市值前100,offset
per.大市值前50,offset
per.大市值前100,offset
用于判断量化日测试股票在全市场中是否满足给定条件（注意:返回的是数值，不是布尔值）
```

参考量化(选股)指令

参考量化意思是在选个股的时候，根据大盘指数或者其他股票走势做量化分析。基本的参考量化指令为ref，使用时在指令元加入ref标示即可，如下例

```
config.source.ref.code=000300
buy.ref.ohlc.close,0>ma,5,10
buy.ref.ma,5,0>ma,60,0
buy.ohlc.swing,3,0>value,4.8
buy.k.涨停,0
```

此小策略意思是如果沪深300指数量化日收盘价高于5日均线，且5日均线高于60日均线，然后分析个股，如果此量化日给定个股满足涨停且收盘价相对开盘价超过4.8涨幅则标记为买入

如果涉及函数调用，可以如下写

```
buy.@var>value,2.0
```

其中@var为函数计算值

Sea脚本扩展指令集

SEA (Snail Extendable Addon) 脚本 是基于蜗牛量化基础指令集的指令簇，其中定义了常用的指令集合，比如MACD金叉判断，指令可分为2句，如下：

```
buy.macd.diff,0>macd.dea,0  
buy.macd.diff,-1
```

对用的sea脚本指令为

```
sea.MACD金叉,0  
其SEA脚本写法如下：  
  
command=MACD金叉  
type=common  
arguments=offset  
items=macd.diff,offset> macd.dea,offset  
items=macd.diff,offset-1
```

目前type类型定义为common常量即可，arguments参数可选1-n个具体以需要而定
items为指令条，在预编译的时候具体的参数会被替换为给定值

```
sea.中阳线,offset  
sea.中阴线,offset  
sea.向下加速度线,offset  
sea.大阳线,offset  
sea.大阴线,offset  
sea.小阳线,offset  
sea.小阴线,offset  
sea.揉搓线,offset  
sea.真阳假阴,offset  
sea.真阴假阳,offset  
sea.螺旋桨,offset  
sea.连涨3天,offset  
sea.连跌3天,offset
```

Boosting策略组

Boosting策略基于人工智能中的boosting算法，具体是针对一个具体形态，然后分拆其他可能相关的参数和指标，来判断对策略结果的影响程度；使用本模块能够优化已有策略或快速开发新策略

一个简单的boosting脚本例子如下：

```
config.source.k.dir=data  
#设置分笔数据文件目录  
config.source.ticket.dir=tick  
#设置K线数据格式  
config.source.k.format=default  
#定义单日股票买入排序,默认为成交量降序(可选升序)  
config.trade.buy.order=desc  
#定义股票卖出时机,默认为尾盘(可选第二天开盘价-早盘)  
config.trade.sale.point=close  
#是否使用立即止盈模式,达到设定盈利立刻卖出  
config.trade.sale.rapid=yes  
#设置单股最大持有日期  
config.trade.max.hold.day=3  
#设定单股买入后再次买入禁买日期间隔  
config.trade.min.exclude=6
```



```

#设置止盈点
config.trade.max.profit=64.0
#设定止损点
config.trade.max.lost=-9.0
#设定交易成本(千分之)
config.trade.tradecost=0.8
#设定单日最大可买股票数
config.trade.max.count=3
#设定股票买入最大允许涨幅
config.trade.max.allow.increase=3.2
#设定股票买入最大允许跌幅
config.trade.max.allow.decrease=-3.2
boosting.做多成功率=60.0
boosting.做多最大回撤=-18.0
boosting.做多平均单笔收益=1.8
buy.k.穿刺,0
items=buy.ohlc.vol,-1>ohlc.vol,0^ohlc.vol,-1 items=buy.ohlc.close,0>ma,5,0^ohlc.close,0
items=buy.ohlc.close,0>ma,10,0^ohlc.close,0 items=buy.ohlc.close,0>ma,20,0^ohlc.close,0
items=buy.ohlc.close,0>ma,60,0^ohlc.close,0 items=buy.ohlc.ma,5,0>ma,10,0^ma,5,0
items=buy.ohlc.ma,5,0>ma,20,0^ma,5,0 items=buy.ohlc.vol,0>ohlc.vol,-1^ohlc.vol,0

```

基本配置和普通的量化配置一致，不同之处在于增加了3个配置项，用于设定可接受的策略下限

```

boosting.做多成功率=?
boosting.做多最大回撤=?
boosting.做多平均单笔收益=?

```

boosting.做多成功率用于指定策略成功率下限，默认为60%，

boosting.做多最大回撤用于可接受的最大回撤，默认为-21%

boosting.做多平均单笔收益用于指定平均单笔收益，默认为1.2%

针对做多，如果计算的这3个值均大于设置的值，则在最终展示页面会看到具体的策略脚本和策略分析结果

```
items=buy.ohlc.vol,-1>ohlc.vol,0^ohlc.vol,-1
```

items=?

items项用于指定具体的boosting策略配置单元,其中以^分割，上面的一条在执行的时候会自动拆分为2个策略，分别是buy.ohlc.vol,-1>ohlc.vol,0和buy.ohlc.vol,-1 < ohlc.vol,0

再看上面的整个范例，设置了8个items，每个有2个单元，因此该boosting策略组共有 $2^8=256$ 个策略

还可以使用@BOOSTING_VAR1 - @BOOSTING_VAR9 共9个变量来指定变量可选值，例如如下：

```
@BOOSTING_VAR1=list[0.5;1.0;1.5;2.0]
```

这里@BOOSTING_VAR1可取值分别为0.5,1.0,1.5,2.0

也可以写成以下格式

```
@BOOSTING_VAR1=for[5;10;1]
```

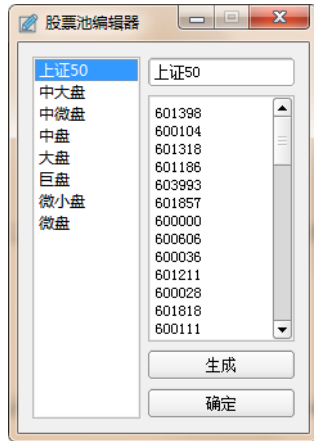
@BOOSTING_VAR1取值为5-10

使用此变量的时候只需要在策略代码中调用即可，如下

```
buy.ma,@BOOSTING_VAR1,0<ohlc.close,0
```

注意：写好的boosting策略脚本需要保存为boosting-script后缀的文件，然后再导入方可执行，批量策略也需要保存为batch-script后缀，再次导入即可

股票池编辑



左侧列表为已存在的股票池列表，其中中大盘，中微盘，中盘，大盘，巨盘，微小盘，微盘七个可以通过点击生成按钮自动从data目录下的stocks文件自动生成

另外，可以输入股票池名称和股票代码列表可以增加和编辑股票池清单

LEVEL2资金流分析



使用LEVEL2股票资金流分析功能可以查看指定股票按日资金流流动情况

一般而言，如果大的资金流入大于大的资金流出，同时根据其他技术分析就可以判断股价短期是否有上涨空间

配对交易版本指令

```
config.mode=pairtrade
config.unit.pairtrade.mincor=90.0
config.unit.pairtrade.maxcor=100.0
config.unit.pairtrade.stock1=sh600519
config.unit.pairtrade.stock2=sz000858
config.unit.pairtrade.最少数据量=60
config.unit.pairtrade.长久持股=false
config.unit.pairtrade.自动中轴=no
config.unit.pairtrade.中轴偏差=1.0
config.unit.pairtrade.移除极点数=3
config.unit.pairtrade.自动宽带=false
config.unit.pairtrade.宽带=6.0
config.unit.pairtrade.k=0.0
```

```
config.unit.pairtrade.relative=no
```

config.mode设置为pairtrade，就是配对交易模式

config.unit.pairtrade.mincor和config.unit.pairtrade.maxcor设置股价相关度上下边界

config.unit.pairtrade.stock1和stock2设置配对股票代码

config.unit.pairtrade.最少数据量指定最少需要的股票数据量

config.unit.pairtrade.长久持股可选yes, no;长久选股指定是否尽可能的持股

config.unit.pairtrade.自动中轴，指定是否使用软件自动计算的中轴

config.unit.pairtrade.中轴偏差指定中轴偏差的范围，可选取值为[-20,20]

config.unit.pairtrade.移除极点数，指定移除的极点数，范围为[0,20]

config.unit.pairtrade.自动带宽，是否使用自动计算的带宽值

config.unit.pairtrade.带宽，取值为[0-50]，指定配对交易带宽

config.unit.pairtrade.k，指定配对交易的K值，此值可以不设定，如果设定之后，没有特殊情况，就不要再修改

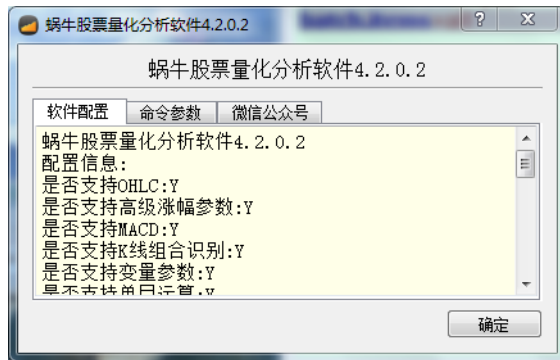
config.unit.pairtrade.relative，指定中轴和带宽的计算取值是相对还是绝对的。通常情况取值为yes，这样可以保证选股数据日期固定不变，不会出现买卖点漂移情况。

双曲线选股量化指令

暂无

配置查看

针对具体蜗牛量化软件可以在菜单中的关于页面查看具体的配置和指令支持情况，示例如下：



参考文章

1. 格力美的股价相似度计算 <https://zhuanlan.zhihu.com/p/43833357>
2. 蜗牛量化Sea脚本 <http://snail007.com/archives/1024>
3. 量化僵尸股 <https://www.jianshu.com/p/0a92f92d0db1>
4. 蜗牛参考量化 <https://www.jianshu.com/p/6e790d71e273>
5. 布林葫芦量化策略 <https://www.jianshu.com/p/f53f65e4fd6f>
6. 通用股份LV2资金流分析 <https://www.jianshu.com/p/9a0fa0a435f7>
- 7.