

自适应均线的介绍

主讲：宁永健

Email:
nyj@icetech.com.tw

课程大纲

- 1. 自适应均线函数代码及原理的介绍;
- 2. 自适应均线模型示例的讲解;



自适应均线的原理（1）

- 趋势类的指标，是我们运用最多，也最容易在市场中获利的方法。有句名言：**截断损失，让利润奔跑，是趋势类模型的真实反映**。均线系统，则是其中最广泛的运用模型。但根据我们的经验，在震荡多的走势上，使用较慢的均线，在趋势快速展开的走势上运用更快的均线；那么此时我们就引入了一种自适应均线的模型。
- 基本概念：
 - 1. 价格轨迹的效率（位移与路程之比）；
 - 2. 动态平均算法（例：指数移动平均的权重）



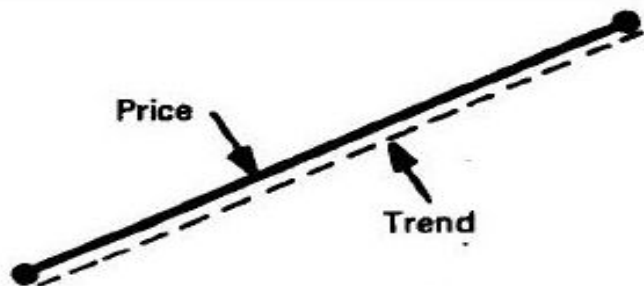
价格轨迹的效率

- 在行情的走势图图中，可以大致分为两种走势：一种是一直上攻的走势，被称为高效率的，因为每一天收盘价格的变动都直接贡献于总的涨幅；另一种是反复震荡的走势，被称为低效率的，很多次收盘价格的变化相互抵消。类似于物理学中路程和位移的概念，如果走过的路程很长，但是位移很小，在实现位移的目标考量下，这样的运动可以称为低效率的。
- 那么价格轨迹的效率就可以理解为：
- 位移和路程之比；



上涨走势的效率性

(a) No Noise.
Any trend speed will work.



(b) More Noise.
A slower trend must be used.



(c) Lots of Noise.
The slowest trend is best.



动态平均算法

- 动态平均是一个迭代的定义，比如时间序列 y_t 是另一个时间序列 x_t 的动态平均，意味着： $y_t = ax_t + (1-a)y_{t-1}$
- 示例： $XAverage = XAverage[1] + var0 * (PriceValue - XAverage[1]);$ //指数移动平均的计算方式 此时的 $var0$ 和 a 效果一样，就是我们的加权因子，也可以理解为权重；
- 而自适应均线的加权因子，就是由价格轨迹的效率去计算的（加权因子的计算原理，当然也可以有其他的计算方式）；
- $a = E * (2 / (fastlen + 1) - 2 / (slowlen + 1)) + 2 / (slowlen + 1);$ //E 代表我们刚才所说价格轨迹的效率；
E的取值范围： $0 < E < 1$



自适应均线指标源码

```
1 inputs:
2     Price( Close ),
3     EffRatioLength( 10 ),
4     FastAvgLength( 5 ),
5     SlowAvgLength( 60 ) ;
6 variables:
7     var0( 0 ) ,    flag(0);
8
9 var0 = AdaptiveMovAvg( Price, EffRatioLength, FastAvgLength, SlowAvgLength ) ;
10 Plot1( var0, "MAA" ) ;
11
12 condition1 = Price crosses over var0 ;
13 if condition1 then
14     flag=1;
15     if flag=1 then
16         SetPlotColor(1,red);
17
18 condition2 = Price crosses under var0 ;
19 if condition2 then
20     flag=0;
21     if flag=0 then
22         SetPlotColor(1,yellow);
23
```


指标的图表显示



模型开发的过程

对于任何一个新的策略模型的开发，都不能只是把其他的类似策略源码直接复制过来，"查看历史的回测绩效如"去判断这个模型的好坏。我的观点是：先有思路，才有量化。

比如这个自适应均线的指标，我的思路是先把均线指标去画出来，去观察其在图表上的走势，再去根据观察到的规律转化为代码，才会有后续量化的部分。通过上面的图表显示，我不会直接去采用均线金叉死叉，去做多做空的操作。而是发现在不同颜色的均线下，图表K的走势会有一定的规律。



自适应均线的思路

- 模型概念：
- 通过观察图表得出，当K线走在黄线下面时，去执行做空由一定几率获得较大收益；相反K线走在红线上方时，做多有一定几率获得较大的收益。
- 那么此时我的思路：//代码显示
- `condition1 = Price crosses over var0 ;`
- `if condition1 then begin`
- `flag=1;`
- `end;`
- `condition2 = Price crosses under var0 ;`
- `if condition2 then begin`
- `flag=0;`
- `end;`
- 此时的自适应均线，只是作为一个做多或做空的一个过滤条件，并不是直接参与到委托的条件中；



模型滤网的介绍（1）

- ATR: 平均真实波幅
- if averagetrue range(M) < > 0 then
- value1 = averagetrue range(N) / averagetrue range(M);
- 通过短期和长期ATR 的比值，来判断市场上价格波动的移动力度，作为突破入场的一个契机；



模型滤网的介绍（2）

- 如图所示：在红黄交叉显示频繁的时候，会频繁入场，此时为了减少交易次数，进行如下限制；

```
if condition1 or condition2 then  
value2=0;  
if flag=1 or flag=0 then  
value2=value2+1;  
condition3=value2>5;
```



其他进出场条件介绍

- 1. 三连阳或三连阴时才会进场（此时的进场是用到了K棒形态的概念）；
- 2. 每次自适应均线变红色或黄色的后续时段内，只会入场一次，直到由黄变红，或由红变黄；
• （此语句可以减少进场次数）；
- 3. 当自适应均线颜色变化之后，立刻对所持有的部位了结；



出场止盈止损模块

```
1 if EntryPrice(0)<>0 then begin
2   if mp>0 then begin
3     B_stop=maxlist(B1,absvalue(entryprice(0)-loss point*minmove));
4     sell ("B1 Exit") from entry("B1") next bar at B_stop stop;
5   end;
6   if mp<0 then begin
7     S_stop=minlist(S1,absvalue(entryprice(0)+loss point*minmove));
8     buytocovert ("S1 Exit") from entry("S1") next bar at S_stop stop;
9   end;
10
11 //short profit
12 Inputs:Sprofit(40);
13 if mp[1]=0 and mp<0 then
14   mylow=Low
15 else if mp<0 and mp[1]<0 then begin
16   if low<mylow then
17     mylow=low;
18
19 if PosTradeProfit(0,0)>Sprofit*bigpointvalue and PosTradeEntryPrice(0,0)>0 then begin
20   value30=mylow+absvalue(mylow-postradeentryprice(0,0))*ratio;
21   buytocovert("S1_profit") from entry("S1") next bar at value30 stop;
22 end;
23 end;
24 //Buy profit
25 inputs:Bprofit(60);
26
27 if mp[1]=0 and mp>0 then
28   myhigh=high
29 else if mp>0 and mp[1]>0 then begin
30   if high>myhigh then
31     myhigh=high;
32
33 if postradeprofit(0,0)>Bprofit*bigpointvalue and PosTradeEntryPrice(0,0)>0 then begin
34   value20=myhigh-absvalue(myhigh-PosTradeEntryPrice(0,0))*ratio;
35   sell ("B1_profit") from entry("B1") next bar at value20 stop;
36 end;
```


模型绩效图





• Thank you for your time!