

## MinimosQuadrados.mql

```
1. #property copyright "Copyright 2014, Cleiton Gomes"
2. #property link      "cleitoncsg@gmail.com"
3.
4. #define ALAVANCAGEM 0.25
5. #define QUANTIDADE_CANDLES 34
6. #define AJUSTE_SL 8
7. #define AJUSTE_CA 0.1
8. #define SEXTA_FEIRA 5
9.
10. extern double take_profit_fixo, stop_loss_fixo;
11.
12. int ticket=0;
13. string nome = "CSG";
14. double coeficienteAngular;
15.
16. int start(){
17.     bool venda, compra;
18.     double take_profit, stop_loss;
19.     double produto_coeficienteAngular_cotacao;
20.
21.     produto_coeficienteAngular_cotacao =
        calculoCoeficienteLinear(QUANTIDADE_CANDLES);
22.
23.     if( DayOfWeek() != SEXTA_FEIRA ){
```

```

24.         if(coeficienteAngular < 0){
25.             coeficienteAngular = coeficienteAngular*(-1);
26.         }
27.         if(produto_coeficienteAngular_cotacao > 1){
28.             compra = true;
29.         }
30.         if(produto_coeficienteAngular_cotacao < 0){
31.             venda = true;
32.         }
33.         take_profit = (Ask + (coeficienteAngular)*AJUSTE_CA);
34.         stop_loss = (Bid -
            AJUSTE_SL*(coeficienteAngular)*AJUSTE_CA);
35.
36.     }
37.
38.     if( ((compra == true && OrdersTotal() == 0 && venda !=
        true)) ){
39.         take_profit_fixo = take_profit;
40.         stop_loss_fixo = stop_loss;
41.         RefreshRates();
42.         while (IsTradeContextBusy()) Sleep(5);
43.         ticket=
            OrderSend(Symbol(),OP_BUY,ALAVANCAGEM,Ask,0,stop_loss_fixo,
44.         take_profit_fixo,nome,AccountNumber(),0,Yellow);
45.     }

```

```
46.     if( ((venda == true && OrdersTotal() == 0 && compra !=
        true)) ){
47.         take_profit_fixo = take_profit;
48.         stop_loss_fixo = stop_loss;
49.
50.         RefreshRates();
51.         while (IsTradeContextBusy()) Sleep(5);
52.         ticket=
            OrderSend(Symbol(),OP_SELL,ALAVANCAGEM,Bid,0,stop_loss_fixo,
53.         take_profit_fixo,nome,AccountNumber(),0,Green);
54.     }
55.
56.     double ponto_positivo, ponto_negativo;
57.
58.     for(int j=0; j < OrdersHistoryTotal();j++){
59.         OrderSelect(j,SELECT_BY_POS,MODE_HISTORY);
60.         if(OrderSymbol()!=Symbol()) continue;
61.         if(OrderMagicNumber() != AccountNumber())
            continue;
62.         if(OrderProfit() > 0){
63.             ponto_positivo++;
64.         }
65.         else{
66.             ponto_negativo++;
67.         }
```

```

68.         }
69.
70.     Comment(
71.         "Margem da Conta = ", AccountMargin() ,"\n",
72.         "Ordens em lucro = ", ponto_positivo ,"\n",
73.         "Ordens em prejuizo = ", ponto_negativo ,"\n",
74.         "STOP LOSS ", stop_loss_fixo ,"\n",
75.         "TAKE PROFIT", take_profit_fixo ,"\n",
76.         "COEFICIENTE LINEAR ", calculoCoeficienteLinear(34)
77.         ,"\n",
78.         "COEFICIENTE ANGULAR ", coeficienteAngular ,"\n",
79.         ""
80.     );
81.     return(0);
82. }
83.
84. double calculoCoeficienteLinear(int quantidadeVelas){
85.     double soma_x = 0, soma_y = 0;
86.     double numerador, denominador;
87.     double variacaoLinear;
88.     int i;
89.
90.     for(i = 1; i < quantidadeVelas; i++){
91.         soma_x = soma_x + Open[i];

```

```
92.         soma_y = soma_y + Close[i];
93.     }
94.
95.     for(i = 1; i < quantidadeVelas; i++){
96.         numerador = Open[i]*(Close[i] -
            soma_x/quantidadeVelas);
97.         denominador = Close[i]*(Open[i] -
            soma_y/quantidadeVelas);
98.     }
99.     variacaoLinear = numerador/denominador;
100.    coeficienteAngular = soma_y/quantidadeVelas -
        (variacaoLinear*soma_x/quantidadeVelas);
101.
102.    return variacaoLinear;
103. }
```