

correlacaoLinear.c

```
1 //+-----+
2 /|          MÃ©todo de CorrelacÃ£o Linear |
3 /|          Copyright Â© 2014, Cleiton Gomes; Vanessa Barbosa |
4 /|          http://www.softwarecsg.com.br |
5 //+-----+
6
7 #include <stdio.h>
8 #include <stdlib.h>
9 #include <string.h>
10 #include <math.h>
11 #define QUANTIDADE_CANDLES 100
12 #define TAMANHO_STRING 50
13
14 char nomeRobo[TAMANHO_STRING], nomeTipoGrafico[2];
15 double quantidadeCandes;
16 double tendencia;
17
18 double calculaCorrelacao(int tempoCorrelacao);
19 void detectaRoboETipoDeGrafico();
20
21 int main(){
22     FILE *arquivo;
23     FILE *arquivoTendencia;
24
25     detectaRoboETipoDeGrafico();
26     printf("METODO CORRELACAO EM C LIGADO\n");
27     //printf("Correlacao Linear em C: %f\n",calculaCorrelacao(quantidadeCandes));
28
29     arquivo = fopen("../correlacaoResposta.txt", "wt");
30     fprintf(arquivo, "%f", calculaCorrelacao(quantidadeCandes));
31
32     fclose(arquivo);
33     arquivoTendencia = fopen("../tendencia.txt", "wt");
34     //printf("tendencia : %f\n", tendencia);
35     fprintf(arquivoTendencia, "%f", tendencia);
36
37     fclose(arquivoTendencia);
38
39     return 0;
40 }
```

```

41
42 void detectaRoboETipoDeGrafico(){
43     FILE *arquivo;
44     char temporariaQuantidadeCandle[10];
45
46     arquivo = fopen("../criterioEntrada.txt","rt");
47
48     if(arquivo == NULL){
49         printf("Arquivo nulo\n");
50     }
51
52     fscanf(arquivo,"%s",nomeRobo);
53     //printf("nome robo: %s\n", nomeRobo);
54     fscanf(arquivo,"%s",nomeTipoGrafico);
55     //printf("Nome tipo grafico: %s\n", nomeTipoGrafico);
56     fscanf(arquivo,"%s",temporariaQuantidadeCandle);
57     //printf("Quantidade candles %s\n", temporariaQuantidadeCandle);
58     quantidadeCandes = atoi(temporariaQuantidadeCandle);
59
60     fclose(arquivo);
61 }
62
63 double calculaCorrelacao(int tempoCorrelacao){
64
65     FILE *arquivo;
66     int c;
67     double somaOrdenadas = 0, somaAbcissas = 0,
68         somaOrdenadasQuadrado = 0, somaAbcissasQuadrado = 0,
69         somaXvezesY = 0, correlacao,
70         numeroAbcissa, numeroOrdenada,
71         numerador, denominador_1,denominador;
72
73     double leituraCotacoes[tempoCorrelacao];
74
75     if( (strcmp(nomeTipoGrafico,"M1")) == 0)
76         arquivo = fopen("../MQL4/Files/M1.csv","rt");
77     else if( (strcmp(nomeTipoGrafico,"M5")) == 0)
78         arquivo = fopen("../MQL4/Files/M5.csv","rt");
79     else if( (strcmp(nomeTipoGrafico,"M15")) == 0)
80         arquivo = fopen("../MQL4/Files/M15.csv","rt");
81     else if( (strcmp(nomeTipoGrafico,"M30")) == 0)
82         arquivo = fopen("../MQL4/Files/M30.csv","rt");

```

```

83  else if( (strcmp(nomeTipoGrafico,"H1")) == 0)
84      arquivo = fopen("../MQL4/Files/H1.csv","rt");
85  else if( (strcmp(nomeTipoGrafico,"H4")) == 0)
86      arquivo = fopen("../MQL4/Files/H4.csv","rt");
87  else if( (strcmp(nomeTipoGrafico,"D1")) == 0)
88      arquivo = fopen("../MQL4/Files/D1.csv","rt");
89  else if( (strcmp(nomeTipoGrafico,"MN1")) == 0)
90      arquivo = fopen("../MQL4/Files/MN1.csv","rt");
91  else if( (strcmp(nomeTipoGrafico,"W1")) == 0)
92      arquivo = fopen("../MQL4/Files/W1.csv","rt");
93  else
94      printf("Erro, tabela nao encontrada\n");
95
96  for(c=0; c<= tempoCorrelacao; c++){
97      fscanf(arquivo, "%lf",&leituraCotacoes[c]);
98  }
99
100  for(c=0; c< tempoCorrelacao; c++){
101      numeroAbcissa = leituraCotacoes[c];
102      numeroOrdenada = leituraCotacoes[c+1];
103
104      somaAbcissas = somaAbcissas + numeroAbcissa;
105      somaAbcissasQuadrado += (numeroAbcissa*numeroAbcissa);
106      somaOrdenadas = somaOrdenadas + numeroOrdenada;
107      somaOrdenadasQuadrado += (numeroOrdenada*numeroOrdenada);
108      somaXvezesY = somaXvezesY + (numeroOrdenada*numeroAbcissa);
109  }
110
111  numerador
= ((tempoCorrelacao*somaXvezesY)-((somaAbcissas)*(somaOrdenadas)));
112  printf("Numerador %lf\n", numerador);
113  denominador_1
= ((tempoCorrelacao*somaAbcissasQuadrado)-(somaAbcissas*somaAbcissas))*
114
((tempoCorrelacao*somaOrdenadasQuadrado)-(somaOrdenadas*somaOrdenadas));
115
116  denominador = sqrt(denominador_1);
117  printf("Denominador %lf\n", denominador);
118  tendencia = somaAbcissas - somaOrdenadas;
119  correlacao = numerador/denominador;
120
121  fclose(arquivo);

```

```
122 printf("Correlacao: %lf\n", correlacao);
123 return correlacao;
124 }
```

fibonacci.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #define TAMANHO_STRING 50
4
5 char nomeRobo[TAMANHO_STRING], nomeTipoGrafico[2];
6
7 void detectaRoboETipoDeGrafico();
8 double calculoSuporte(int quantidadeVelas);
9 double calculoResistencia(int quantidadeVelas);
10 double calculoRegressaoFibonacci(double fatorDeRegressao, int quantidadeVelas);
11 void lerTendencia();
12
13 double quantidadeCandes;
14 double tendencia;
15
16 int main(){
17     FILE *arquivo;
18     FILE *arquivoSuporteResistencia;
19
20     detectaRoboETipoDeGrafico();
21     lerTendencia();
22     //printf("Suporte = %lf, resistencia =
23     %lf\n",calculoSuporte(13),calculoResistencia(13));
24     //printf("Regressao De Fibonacci = %lf\n",calculoRegressaoFibonacci(0.23, 13));
25     printf("METODO FIBONACCI EM C LIGADO\n");
26
27     arquivo = fopen("../fibonacciResposta.txt", "wt");
28
29     //fprintf(arquivo, "%f\n", calculoRegressaoFibonacci(0.23, quantidadeCandes));
30     fprintf(arquivo, "%f\n", calculoRegressaoFibonacci(0.38, quantidadeCandes));
31     //fprintf(arquivo, "%f\n", calculoRegressaoFibonacci(0.62, quantidadeCandes));
32     fclose(arquivo);
33
34     arquivoSuporteResistencia = fopen("../suporteResistencia.txt", "wt");
35     fprintf(arquivoSuporteResistencia, "%f\n", calculoSuporte(quantidadeCandes));
```

```

35  fprintf(arquivoSuporteResistencia, "%f\n", calculoResistencia(quantidadeCandes));
36
37  return 0;
38 }
39
40 void detectaRoboETipoDeGrafico(){
41  FILE *arquivo;
42  char temporariaQuantidadeCandle[10];
43
44  arquivo = fopen("../criterioEntrada.txt", "rt");
45
46  if(arquivo == NULL){
47      printf("Arquivo nulo\n");
48  }
49
50  fscanf(arquivo, "%s", nomeRobo);
51  //printf("nome robo: %s\n", nomeRobo);
52  fscanf(arquivo, "%s", nomeTipoGrafico);
53  //printf("Nome tipo grafico: %s\n", nomeTipoGrafico);
54  fscanf(arquivo, "%s", &temporariaQuantidadeCandle);
55  //printf("Quantidade candles %s\n", temporariaQuantidadeCandle);
56  quantidadeCandes = atoi(temporariaQuantidadeCandle);
57
58  fclose(arquivo);
59 }
60
61 double calculoSuporte(int quantidadeVelas){
62  FILE *arquivo;
63  double cotacao[quantidadeVelas];
64  double suporte = 777;
65  int i;
66
67  if( (strcmp(nomeTipoGrafico, "M1")) == 0)
68      arquivo = fopen("../MQL4/Files/M1.csv", "rt");
69  else if( (strcmp(nomeTipoGrafico, "M5")) == 0)
70      arquivo = fopen("../MQL4/Files/M5.csv", "rt");
71  else if( (strcmp(nomeTipoGrafico, "M15")) == 0)
72      arquivo = fopen("../MQL4/Files/M15.csv", "rt");
73  else if( (strcmp(nomeTipoGrafico, "M30")) == 0)
74      arquivo = fopen("../MQL4/Files/M30.csv", "rt");
75  else if( (strcmp(nomeTipoGrafico, "H1")) == 0)
76      arquivo = fopen("../MQL4/Files/H1.csv", "rt");

```

```

77     else if( (strcmp(nomeTipoGrafico,"H4")) == 0)
78         arquivo = fopen("../MQL4/Files/H4.csv","rt");
79     else if( (strcmp(nomeTipoGrafico,"D1")) == 0)
80         arquivo = fopen("../MQL4/Files/D1.csv","rt");
81     else if( (strcmp(nomeTipoGrafico,"MN1")) == 0)
82         arquivo = fopen("../MQL4/Files/MN1.csv","rt");
83     else if( (strcmp(nomeTipoGrafico,"W1")) == 0)
84         arquivo = fopen("../MQL4/Files/W1.csv","rt");
85     else
86         printf("Erro, tabela nao encontrada\n");
87
88     for(i = 0; i <= quantidadeVelas; i++){
89         fscanf(arquivo, "%lf",&cotacao[i]);
90         if(suporte > cotacao[i])
91             suporte = cotacao[i];
92     }
93
94     fclose(arquivo);
95     return suporte;
96 }
97
98 double calculoResistencia(int quantidadeVelas){
99     FILE *arquivo;
100     double cotacao[quantidadeVelas];
101     double resistencia = 0;
102     int i;
103
104     if( (strcmp(nomeTipoGrafico,"M1")) == 0)
105         arquivo = fopen("../MQL4/Files/M1.csv","rt");
106     else if( (strcmp(nomeTipoGrafico,"M5")) == 0)
107         arquivo = fopen("../MQL4/Files/M5.csv","rt");
108     else if( (strcmp(nomeTipoGrafico,"M15")) == 0)
109         arquivo = fopen("../MQL4/Files/M15.csv","rt");
110     else if( (strcmp(nomeTipoGrafico,"M30")) == 0)
111         arquivo = fopen("../MQL4/Files/M30.csv","rt");
112     else if( (strcmp(nomeTipoGrafico,"H1")) == 0)
113         arquivo = fopen("../MQL4/Files/H1.csv","rt");
114     else if( (strcmp(nomeTipoGrafico,"H4")) == 0)
115         arquivo = fopen("../MQL4/Files/H4.csv","rt");
116     else if( (strcmp(nomeTipoGrafico,"D1")) == 0)
117         arquivo = fopen("../MQL4/Files/D1.csv","rt");
118     else if( (strcmp(nomeTipoGrafico,"MN1")) == 0)

```

```

119     arquivo = fopen("../MQL4/Files/MN1.csv","rt");
120     else if( strcmp(nomeTipoGrafico,"W1") == 0)
121         arquivo = fopen("../MQL4/Files/W1.csv","rt");
122     else
123         printf("Erro, tabela nao encontrada\n");
124
125     for(i = 0; i <= quantidadeVelas; i++){
126         fscanf(arquivo, "%lf",&cotacao[i]);
127
128         if(resistencia < cotacao[i])
129             resistencia = cotacao[i];
130     }
131
132     fclose(arquivo);
133     return resistencia;
134 }
135
136 double calculoRegressaoFibonacci(double fatorDeRegressao, int quantidadeVelas){
137     double variacaoDePontos;
138     double fibonacci;
139
140     if(tendencia>0){
141         variacaoDePontos =
calculoSuporte(quantidadeVelas)-calculoResistencia(quantidadeVelas);
142         fibonacci = (variacaoDePontos*fatorDeRegressao) +
calculoResistencia(quantidadeVelas);
143         return fibonacci;
144     }
145
146     else{
147         variacaoDePontos =
calculoResistencia(quantidadeVelas)-calculoSuporte(quantidadeVelas);
148         fibonacci = (variacaoDePontos*fatorDeRegressao) +
calculoSuporte(quantidadeVelas);
149         return fibonacci;
150     }
151 }
152
153 void lerTendencia(){
154     FILE *arquivoTendencia;
155     double temporariaTendencia;
156

```

```

157  arquivoTendencia = fopen("../tendencia.txt", "rt");
158
159  if(arquivoTendencia == NULL){
160      printf("Arquivo nulo\n");
161  }
162
163  fscanf(arquivoTendencia, "%lf", &temporariaTendencia);
164  tendencia = temporariaTendencia;
165  //printf("Tendencia: %lf\n", tendencia);
166
167  fclose(arquivoTendencia);
168 }

```

minimosQuadrados.c

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #define TAMANHO_STRING 50
4
5  double calculoCoeficienteLinear();
6  double calculoCoeficienteAngular();
7
8  char nomeRobo[TAMANHO_STRING], nomeTipoGrafico[2];
9
10 void detectaRoboETipoDeGrafico();
11 int quantidadeVelas;
12
13 int main(){
14     FILE *arquivo;
15
16     detectaRoboETipoDeGrafico();
17     printf("METODO MINIMOS QUADRADOS LIGADO\n");
18
19     arquivo = fopen("../minimosQuadradosResposta.txt", "wt");
20
21     fprintf(arquivo, "%f\n", calculoCoeficienteLinear(quantidadeVelas));
22     fprintf(arquivo, "%f\n", calculoCoeficienteAngular(quantidadeVelas));
23
24     //printf("Angular: %lf\n", calculoCoeficienteAngular(quantidadeVelas));
25     //printf("Linear: %lf\n", calculoCoeficienteLinear(quantidadeVelas));
26     fclose(arquivo);

```



```

27
28     return 0;
29 }
30
31 void detectaRoboETipoDeGrafico(){
32     FILE *arquivo;
33     char temporariaQuantidadeCandle[10];
34
35     arquivo = fopen("../criterioEntrada.txt","rt");
36
37     if(arquivo == NULL){
38         printf("Arquivo nulo\n");
39     }
40
41     fscanf(arquivo,"%s",nomeRobo);
42     //printf("nome robo: %s\n", nomeRobo);
43     fscanf(arquivo,"%s",nomeTipoGrafico);
44     //printf("Nome tipo grafico: %s\n", nomeTipoGrafico);
45     fscanf(arquivo, "%s",temporariaQuantidadeCandle);
46     //printf("Quantidade candles %s\n", temporariaQuantidadeCandle);
47     quantidadeVelas = atoi(temporariaQuantidadeCandle);
48
49     fclose(arquivo);
50 }
51
52 double calculoCoeficienteLinear(int quantidadeVelas){
53     double variacaoLinear;
54     double media_y,media_x;
55     FILE *arquivo;
56     double x[quantidadeVelas], y[quantidadeVelas];
57     double soma_x = 0, soma_y = 0;
58     double variacaoAngular;
59     int i,c;
60     double leituraCotacoes[quantidadeVelas+1];
61
62     if( (strcmp(nomeTipoGrafico,"M1")) == 0)
63         arquivo = fopen("../MQL4/Files/M1.csv","rt");
64     else if( (strcmp(nomeTipoGrafico,"M5")) == 0)
65         arquivo = fopen("../MQL4/Files/M5.csv","rt");
66     else if( (strcmp(nomeTipoGrafico,"M15")) == 0)
67         arquivo = fopen("../MQL4/Files/M15.csv","rt");
68     else if( (strcmp(nomeTipoGrafico,"M30")) == 0)

```

```

69     arquivo = fopen("../MQL4/Files/M30.csv", "rt");
70     else if( strcmp(nomeTipoGrafico, "H1")) == 0)
71         arquivo = fopen("../MQL4/Files/H1.csv", "rt");
72     else if( strcmp(nomeTipoGrafico, "H4")) == 0)
73         arquivo = fopen("../MQL4/Files/H4.csv", "rt");
74     else if( strcmp(nomeTipoGrafico, "D1")) == 0)
75         arquivo = fopen("../MQL4/Files/D1.csv", "rt");
76     else if( strcmp(nomeTipoGrafico, "MN1")) == 0)
77         arquivo = fopen("../MQL4/Files/MN1.csv", "rt");
78     else if( strcmp(nomeTipoGrafico, "W1")) == 0)
79         arquivo = fopen("../MQL4/Files/W1.csv", "rt");
80     else
81         printf("Erro, tabela nao encontrada\n");
82
83     for(c=0; c<= quantidadeVelas; c++){
84         fscanf(arquivo, "%lf", &leituraCotacoes[c]);
85     }
86
87     for(c=0; c<= quantidadeVelas; c++){
88         x[c] = leituraCotacoes[c];
89         y[c] = leituraCotacoes[c+1];
90     }
91
92     for(i = 0; i < quantidadeVelas; i++){
93         soma_x = soma_x + x[i];
94         soma_y = soma_y + y[i+1];
95     }
96
97     media_x=soma_x/quantidadeVelas;
98     media_y=soma_y/quantidadeVelas;
99
100     variacaoLinear= media_y - (calculoCoeficienteAngular(quantidadeVelas) *
media_x);
101     fclose(arquivo);
102
103     return variacaoLinear;
104 }
105
106 double calculoCoeficienteAngular(int quantidadeVelas){
107     FILE *arquivo;
108     double x[quantidadeVelas], y[quantidadeVelas];
109     double soma_x = 0, soma_y = 0;

```

```

110  double variacaoAngular;
111  int i,c;
112  double numerador=0, denominador=0;
113  double leituraCotacoes[quantidadeVelas+1];
114  double media_x, media_y;
115
116  if( (strcmp(nomeTipoGrafico,"M1")) == 0)
117      arquivo = fopen("../MQL4/Files/M1.csv","rt");
118  else if( (strcmp(nomeTipoGrafico,"M5")) == 0)
119      arquivo = fopen("../MQL4/Files/M5.csv","rt");
120  else if( (strcmp(nomeTipoGrafico,"M15")) == 0)
121      arquivo = fopen("../MQL4/Files/M15.csv","rt");
122  else if( (strcmp(nomeTipoGrafico,"M30")) == 0)
123      arquivo = fopen("../MQL4/Files/M30.csv","rt");
124  else if( (strcmp(nomeTipoGrafico,"H1")) == 0)
125      arquivo = fopen("../MQL4/Files/H1.csv","rt");
126  else if( (strcmp(nomeTipoGrafico,"H4")) == 0)
127      arquivo = fopen("../MQL4/Files/H4.csv","rt");
128  else if( (strcmp(nomeTipoGrafico,"D1")) == 0)
129      arquivo = fopen("../MQL4/Files/D1.csv","rt");
130  else if( (strcmp(nomeTipoGrafico,"MN1")) == 0)
131      arquivo = fopen("../MQL4/Files/MN1.csv","rt");
132  else if( (strcmp(nomeTipoGrafico,"W1")) == 0)
133      arquivo = fopen("../MQL4/Files/W1.csv","rt");
134  else
135      printf("Erro, tabela nao encontrada\n");
136
137  for(c=0; c<= quantidadeVelas; c++){
138      fscanf(arquivo, "%lf",&leituraCotacoes[c]);
139  }
140
141  for(c=0; c<= quantidadeVelas; c++){
142      x[c] = leituraCotacoes[c];
143      y[c] = leituraCotacoes[c+1];
144  }
145
146  for(i = 0; i < quantidadeVelas; i++){
147      soma_x = soma_x + x[i];
148      soma_y = soma_y + y[i+1];
149  }
150
151  media_x=soma_x/quantidadeVelas;

```

```
152  media_y=soma_y/quantidadeVelas;
153
154  for (i = 0; i < quantidadeVelas; ++i)
155  {
156      numerador+=(x[i]-media_x)*(y[i]-media_y);
157      denominador+= (x[i]-media_x)*(x[i]-media_x);
158  }
159  variacaoAngular=numerador/denominador;
160  fclose(arquivo);
161
162  return variacaoAngular;
163 }
```