

Material Supplementar de “Física de processos estocásticos aplicada a opções binárias no mercado financeiro”

Programa II: Realiza o “backtest” em paridades cambiais tais como EUR/USD utilizando o indicador oscilador estocástico.

```
program backtestEstocastico
implicit none
type TPonto
    real*8::op,hi,lo,cl,estK
end type TPonto
character(len=500)::fileName
character(len=100)::sl
integer::i,k,total,periodoK,tempo
type(TPonto),allocatable,dimension(:)::p
real*8::Ktop,Kbottom,taxaWin,taxaLoss
integer::nwin,nloss,noper

periodoK=14
Ktop=95.D0
Kbottom=5.D0
tempo=3 !em velas

fileName='EURUSDM1.txt'

total=findTotalOfLines(fileName)
print*, 'Total de velas detectados:', total
print*, fileName

allocate(p(1:total))

open(unit=1,file=trim(fileName))
open(unit=2,file='Close.out')
i=0
do
    read(1,'(A100)',end=111)sl
    i=i+1
    call getInput(sl,p(i)%op,p(i)%hi,p(i)%lo,p(i)%cl)
    write(2,*)i,p(i)%cl
enddo
111 continue
close(unit=1)
close(unit=2)

call calculaEstocastico(periodoK)

open(unit=1,file='winloss.txt')
nloss=0
nwin=0
noper=0
i=periodoK
```

```

do while (i.lt.(total-tempo-1))
  if(p(i)%estK.ge.Ktop)then
    !aposta na reversao pra baixo
    if(p(i+tempo)%cl.lt.p(i)%cl)then
      nwin=nwin+1
      write(1,*)'Win'
      i=i+tempo
    else
      if(p(i+tempo)%cl.gt.p(i)%cl)then
        nloss=nloss+1
        write(1,*)'Loss'
        i=i+tempo
      else
        !empate, nao conta.
      endif
    endif
  else
    if(p(i)%estK.le.Kbottom)then
      !aposta na reversao pra cima
      if(p(i+tempo)%cl.gt.p(i)%cl)then
        nwin=nwin+1
        write(1,*)'Win'
        i=i+tempo
      else
        if(p(i+tempo)%cl.lt.p(i)%cl)then
          nloss=nloss+1
          write(1,*)'Loss'
          i=i+tempo
        else
          !empate, nao conta.
        endif
      endif
    else
      !regiao neutra, nao interessa.
    endif
  endif
  i=i+1
enddo
close(unit=1)
noper=nwin+nloss
taxaWin=dfloat(nwin)/dfloat(noper)
taxaLoss=dfloat(nloss)/dfloat(noper)

print*, 'nwin:',nwin
print*, 'nloss:',nloss
print*, 'noper:',noper
print*, 'taxaWin:',taxaWin*100.D0
print*, 'taxaLoss:',taxaLoss*100.D0

deallocate(p)
stop

contains

! calcula o numero de linhas de um arquivo
function findTotalOfLines(fileName) result(value)

```

```

implicit none
character(len=500),intent(in)::fileName
integer::value
open(unit=1,file=trim(fileName))
value=0
do
    read(1,*,end=1111)
    value=value+1
enddo
1111 continue
close(unit=1)
end function findTotalOfLines

!Obtem dados de entrada apartir de EURUSDM1.txt
!exportado por https://www.metatrader5.com/
!<DATE-TIME,OPEN,HIGH,LOW,CLOSE,TICKVOL,VOL,SPREAD
!2018.09.07 00:00,1.16226,1.16226,1.16210,1.16210,2,0
!2018.09.07 00:01,1.16213,1.16236,1.16188,1.16188,6,0
!2018.09.07 00:02,1.16185,1.16189,1.16183,1.16188,9,0
subroutine getInput(s,op,hi,lo,cl)
implicit none
character(len=100),intent(in)::s
real*8,intent(out)::op,hi,lo,cl
integer::i,n
integer,dimension(1:6)::virg
n=0
do i=1,100
    if(s(i:i).eq.',')then
        n=n+1
        virg(n)=i
    endif
enddo
read(sl(virg(1)+1:virg(2)-1),*)op
read(sl(virg(2)+1:virg(3)-1),*)hi
read(sl(virg(3)+1:virg(4)-1),*)lo
read(sl(virg(4)+1:virg(5)-1),*)cl
end subroutine getInput

! calcula o oscilador estocastico
subroutine calculaEstocastico(periodo)
implicit none
integer::i,j,n
integer,intent(in)::periodo
real*8::maxHi,minLo,soma
! Curva de porcentagem K
open(unit=1,file='linhaK.out')
do j=1,(total-periodo+1)
    maxHi=p(j)%hi
    minLo=p(j)%lo
    n=(periodo+j-1)
    do i=j,n
        if(p(i)%hi.gt.maxHi)then
            maxHi=p(i)%hi
        endif
        if(p(i)%lo.lt.minLo)then
            minLo=p(i)%lo
        endif
    enddo
    soma=(maxHi-minLo)/100
    write(1,'(f10.2)',advance='no')soma
enddo
end subroutine calculaEstocastico

```

```
        endif
    enddo
    p(n)%estK=((p(n)%cl-minLo)/(maxHi-minLo))*1.D2
    write(1,*)n,p(n)%estK
enddo
close(unit=1)
end subroutine calculaEstocastico
end program backtestEstocastico
```