function output=backtest(v\_rtn,Y,enter\_disp,exit\_disp,v\_corr,enter\_corr,exit\_corr,tday)

%% Calculate Performance statistics based on current spread

trades=0;

Enter=[];

Exit=[];

Long=[];

Short=[];

ret\_v=[];

hp\_v=[];

is\_open=0;

direction=0;

hp\_TH=40;

s = zeros(size(Y,1),size(Y,2));

r\_trade = zeros(size(Y,1),1);

CPnL=zeros(size(Y,1),1);

if enter\_disp>0

for j=2:size(Y,1)

if is\_open~=0 %that means position opened

s(j,1)=s(j-1,1);

r\_trade(entryIDX)=0;

current\_r=[0;s(entryIDX:j-1, :) .\* v\_rtn(entryIDX:j-1,:)];

r\_trade(j)= sum(current\_r);

cumul\_r= [0;s(1:j-1, :) .\* v\_rtn(1:j-1,:)];

CPnL(j)= sum(cumul\_r);

%exit

if (direction==1 && Y(j)<exit\_disp) || v\_corr(j) > exit\_corr || r\_trade(j)<-15 || j-entryIDX > hp\_TH

trades=trades+1;

ret\_v=[ret\_v;r\_trade(j)];

hp\_v=[hp\_v;j-entryIDX];

Exit=[Exit;j];

is\_open=0;

direction=0;

end

else %position not opened

CPnL(j)= CPnL(j-1);

if Y(j)>= enter\_disp && v\_corr(j) <= enter\_corr

s(j, 1) = 1;

entryIDX=j;

is\_open=1;

direction=1;

Enter=[Enter;j];

Long=[Long;j];

end

end

end

else

for j=2:size(Y,1)

if is\_open~=0 %that means position opened

s(j,1)=s(j-1,1);

r\_trade(entryIDX)=0;

current\_r=[0;s(entryIDX:j-1, :) .\* v\_rtn(entryIDX:j-1,:)];

r\_trade(j)= sum(current\_r);

cumul\_r= [0;s(1:j-1, :) .\* v\_rtn(1:j-1,:)];

CPnL(j)= sum(cumul\_r);

%exit

if (direction==-1 && Y(j)>exit\_disp) || v\_corr(j) > exit\_corr || r\_trade(j)<-15 || j-entryIDX > hp\_TH

trades=trades+1;

ret\_v=[ret\_v;r\_trade(j)];

hp\_v=[hp\_v;j-entryIDX];

Exit=[Exit;j];

is\_open=0;

direction=0;

else

end

else %position not opened

CPnL(j)= CPnL(j-1);

if Y(j)<= enter\_disp && v\_corr(j) <= enter\_corr

s(j, 1) = -1;

entryIDX=j;

is\_open=1;

direction=-1;

Enter=[Enter;j];

Short=[Short;j];

else

end

end

end

end

if isempty(Enter)==0

last\_enter=m2xdate(tday(Enter(end)),0);

else

last\_enter=m2xdate(700000,0);

end

if isempty(Enter)==0 && isempty(Exit)==0

if size(Enter,1)>=1

stdev=std(ret\_v);

end

last\_exit=m2xdate(tday(Exit(end)),0);

wins=size(find(ret\_v>0),1);

av\_ret=mean(ret\_v);

hp=round(mean(hp\_v));

winp=wins/trades;

yr\_trades=round(trades/(size(Y,1)/250));

else

last\_exit=m2xdate(700000,0);

av\_ret=-100;

stdev=0;

hp=-100;

winp=-100;

yr\_trades=-100;

end

%% build output

output=[enter\_disp,exit\_disp,enter\_corr,exit\_corr,av\_ret,stdev,yr\_trades,hp,winp,last\_exit,last\_enter];