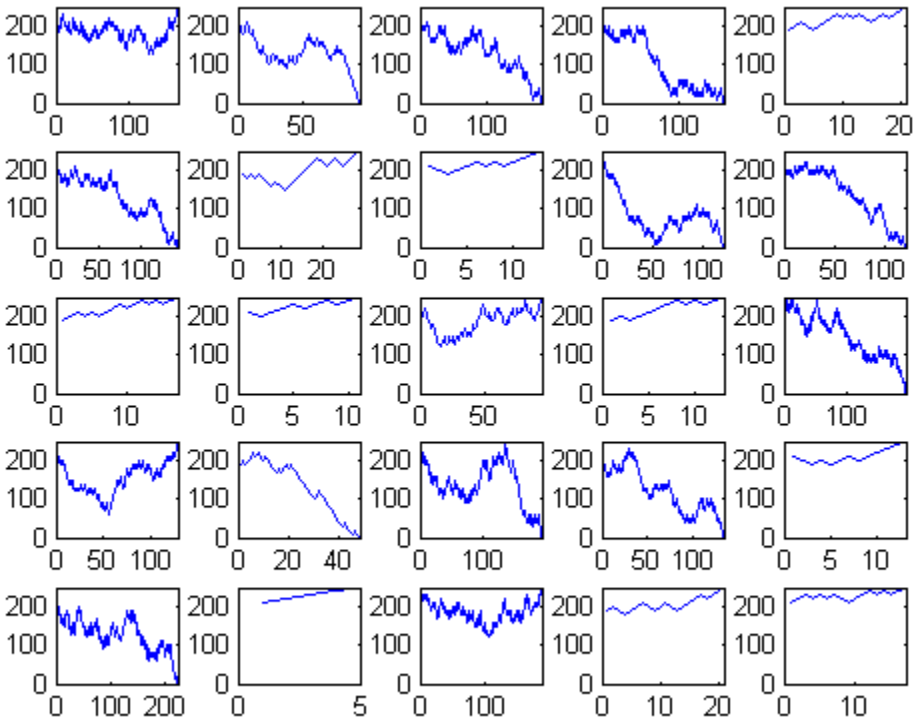
Roulette Simulation

```
Dr. Tim Placek (placetd@auburn.edu)
CHEN 3600 - Lab 4
Spring 2012
The solution presented looks at the winning/losing pattern for a 25 day
period in which the subject is trying to make $50 by betting $10 with
a daily bankroll of $200.  maxgames = 1000 is deliberately "large".


clear;
clc;
wallet = 200;
bet = 10;
hope2make = 50;
max_games = 1000;
days = 25;
winning=0;
losing=0;
for i = 1:days
    [k ret_code profit ] = one_day( hope2make, wallet, bet, max_games );
    subplot(5,5,i);
    plot([1:length(profit)], profit);
    axis([0 k 0 250])
    if ret_code==1; winning = winning + 1; end
    if ret_code==-1; losing = losing + 1; end
end
disp(['Winning  ' num2str(winning)]);
disp(['Losing   ' num2str(losing)]);
net = winning*2*bet-losing*200;
disp(['Net      ' num2str(net)]);
disp('Assuming 25 days of play and no non-win/non-lose situations')


Winning  14
Losing   11
Net      -1920
Assuming 25 days of play and no non-win/non-lose situations
```



External Functions Referenced

```
%{
function [ k retcode profit ] = one_day( hope2make, wallet, bet, max_games )
% This function simulates a single day of playing roulette making
% even money bets.  Play stops when "hope2make" is profet, "wallet" is
% exhausted or "max_games" is exceeded.
p = 18/38;
k = 0;      % game counter
moola = wallet;  % starting funds
target_winnings = wallet+hope2make;
while (k==0) || ~(won | lost | time2quit)
    k=k+1;
    won_spin = rand()<=p;
    moola = moola - bet;
    if won_spin
        moola = moola + 2*bet;
    end
    profit(k)=moola;
    won = moola >= target_winnings;
    lost = moola <= 0;
    time2quit = k==max_games;
end
retcode=0;   % default is time to quit
if won; retcode=1; end;
if lost; retcode=-1; end;
end
%}
```

Published with MATLAB® 7.10