CF963 Computational Models in Economics and Finance

Registration Number: 2010397

Task 1

```
% read csv file and consider closing price for 7MA and 14MA
A = readtable('JET.L.csv');
data = A.Close;

% step 1 finds 7MA and length of 7MA
type = 'simple';
sevenm = movavg(data ,type,7);
sevenma_length = length(sevenm);

% step 2 finds 14MA and length of 7MA
fourteenm = movavg(data ,type,14);

% plot figure for 7MA and 14MA
% plot(data)
% hold on
% plot(fourteenm)
% plot(sevenm)
% hold off
% legend('Actual','fourteenm','sevenm')

budget = 1000000;
share = 0;
final_result = string([]);
header = {'Day', 'Buy/Sell', 'Date', 'Budget', 'Share'};

% start selling and buying using condition
for i=15: sevenma_length
   if(sevenm(i) > fourteenm(i))
      buyshare = floor(budget/data(i));
      if(buyshare > 0)
         budget = budget - (buyshare * data(i));
         share = share + buyshare;
         date = string(A.Date(i), "yyyy-MM-dd");
         final_result(end+1, :) = [i "th row buy on " date budget share];
      end
   elseif (sevenm(i) < fourteenm(i) && share > 0)
      amount = (share * data(i));
      budget = budget + amount;
      share = share - share;
      date = string(A.Date(i), "yyyy-MM-dd");
      final_result(end+1, :) = [i "th row sell on" date budget share];
   end
```

```
end
% step 3 gets when buy and sell and how much buy and what's the budget when buy and sell
data_matrix = [header; final_result];

% step 4 Profit at the end of process
profit = budget – 1000000;
```

Description:

Firstly, read csv file then use closing for calculate 7MA and 14MA, buying and selling on closing price. I have taken budget 1M and share 0.

Step 1 compute 7MA, **step 2** compute 14MA, **step 3** got day and date on which buy and sell occur, amount of share buys and budget when buy and sell occur, **step 4** compute profit at the end of whole process.

A. Step 1 and Step 2 compute 7MA and 14MA

On Day of 15, 37, 103 buy the share and on day of 23, 67, 123 sell shares

Buying Share

"15th row buy on 2020-08-14"

"37th row buy on 2020-09-16"

"103rd row buy on 2020-12-17"

Selling Share

"23rd row sell on 2020-08-26"

"67th row sell on 2020-10-28"

"123rd row sell on 2021-01-19"

B.

Buying Detail

15 Day buy 111 shares

37 Day buy 118 shares

103 Day buy 133 shares

Selling Detail

23 Day sell 111 shares

67 Day sell 118 shares

123 Day sell 133 shares

C. Profit = 48166

Task 2

```
% Step 1 decide random 20 buyer price

buyer = randi([1 200],1,20);
bid = [];
total_unit = 0;

% Step 2 set seller unit
for seller=1:length(buyer)

unit_seller(seller) = 30;
end

% Simulator for 10 rounds
for i=1:10
   % set bid
   for j=1:length(buyer)
     bid(j) = randi([1 buyer(j)]);
         quantity_buy(j) = randi([1 5]);
   end
   [maximum_bid,maximum_bid_index] = max(bid(:));
   quantity = quantity_buy(maximum_bid_index);

   % set asks
   intial = 10;
   for s=1:length(buyer)
      ask(s) = randi([intial 200]);
      intial = intial + 10;
   end
   [minimum_ask,minimum_ask_index] = min(ask(:));
   unit = unit_seller(minimum_ask_index);

   % check seller has unit > buyer quantity
   if(unit > quantity)
      remaining_unit = unit - quantity;
      unit_seller(minimum_ask_index) = remaining_unit;
   else
      need_from_other_seller = quantity - unit;
      remaining_unit = 0;
      unit_seller(minimum_ask_index) = remaining_unit;
      [minimum_second_ask,minimum_second_ask_index] =
min(setdiff(ask(:),min(ask(:))));
      unit_other = unit_seller(minimum_second_ask_index);
      if(unit_other > need_from_other_seller)
         remaining_unit_s = unit_other - need_from_other_seller;
         unit_seller(minimum_second_ask_index) = remaining_unit_s;
      end
   end
   % Step 3 Total unit
```

```
    total_unit = total_unit + quantity;
    % Step 4 find difference between bid and ask
    spread(i) = maximum_bid - minimum_ask;
end

total_unit;
spread;
array_unit_price = [];
for x=1:length(buyer)
    array_unit_price(end+1,:) = [unit_seller(x) ask(x)];
end
% Step 5 remaining unit and price
array_unit_price;

% plot figure for spread
plot(10)
hold on
plot(spread)
hold off
legend('Simulation','spread')
```

Description:

Step 1 random 20 buyers, **step 2** set seller unit, **step 3** compute total unit, **step 4** compute spread, **step 5** remaining unit and price after final round, and in last plot spread.

A. 10 round Simulator

B. Spread [59 119 18 5 136 92 106 72 125 117] and figure

C. 30 units of the item were traded in total

D. remaining unit and price

unit	price
25	122
25	38
20	100
25	188
25	58
30	114
30	101
30	107
30	129
30	193
30	155
30	185
30	154
30	167
30	198
30	165
30	172
30	194

30	200
30	200

Task 3

Part A

$\Pi1 (q1, q2) = Pq1 - c1$

$\Pi1 (q1, q2) = q1 (120 - Q) - 10 - 3q1$

$\Pi1 (q1, q2) = q1(120 - q1 - q2) - 10 - 3q1$

$\Pi1(q1, q2) = 120q1 - q1^2 - q1q2 - 10 - 3q1$

$\Pi1(q1, q2) = 117q1 - q1^2 - q1q2 - 10$

Find a and b

$d(ax^2 + bx + c)/ dx = 2ax + b = 2(-1)q1 + 117 = 117 - 2q1$

The derivative of $\Pi1(q1, q2)$ with respect to q1 should be 0.

$117 - 2q1 - q2 = 0$

Find q1,

$q1 = (117 - q2) / 2$

$\Pi2 (q1, q2) = Pq2 - c2$

$\Pi2 (q1, q2) = q2(120 - Q) - 12 - 6q2$

$\Pi2 (q1, q2) = q2(120 - q1 - q2) - 12 - 6q2$

$\Pi2(q1, q2) = 120q2 - q1 q2 - q2^2 - 12 - 6q2$

$\Pi2(q1, q2) = 114q2 - q1 q2 - q2^2 - 12$

Find a and b

$d(ax^2 + bx + c)/ dx = 2ax + b = 2(-1)q2 + 114 = 114 - 2q2$

The derivative of $\Pi2(q1, q2)$ with respect to q1 should be 0.

$114 - 2q2 – q1 = 0$

Find q2,

$q2 = (114 - q1) / 2$

Put value of q1 in $114 - 2q2 – q1 = 0$ formula,

$114 - 2q2 - ((117 - q2) / 2) = 0$

$- 117 + q2 = 2 (2q2 - 114)$

$228 - 117 = 4q2 - q2$

$3q2 = 111$

$q2 = 37$

Find out q1,

$q1 = (117 - 37) / 2 = 40$

$q1 = 40$

Find P,

$P = 120 – Q = 120 - (q1 + q2) = 120 - 40 - 37 = 43$

$P = 43$

Find $\Pi1(q1, q2)$,

$\Pi1(q1, q2) = 117q1 - q1^2 - q1q2 – 10 = 117(40) - (40)^2 - (40)(37) - 10$

$\Pi1(q1, q2) = 1590$

Find $\Pi2(q1, q2)$,

$\Pi2(q1, q2) = 114q2 - q1\, q2 - q2^2 – 12 = 114(37) - (37)^2 - (40)(37) - 12$

$\Pi2(q1, q2) = 1357$

Find Consumer surplus,

$CS = (40+37)\ (120-43) / 2$

$CS = 2964.5$

Find Total surplus,

$TS = (1590+1357+2964.5)$

$TS = 5911.5$

Profit function for both firm

$\Pi1(q1, q2) = 117q1 - q1^2 - q1q2 - 10$

$\Pi2(q1, q2) = 114q2 - q1\, q2 - q2^2 - 12$

Profit for both firm

$\Pi1(q1, q2) = 1590$

$\Pi2(q1, q2) = 1357$

Production quantities

$q1 = 40$

$q2 = 37$

Consumer Surplus

CS = 2964.5

Total Surplus

TS = 5911.5

Part B,

$r2(q1) = 57 - q1/2$

$P = 120 – Q$

$c1 = 10 + 3q1$

$\Pi1\ (q1, q2) = Pq1 – c1$

$\Pi1\ (q1, q2) = q1(120 – (q1+ r2(q1))) - 10 - 3q1$

$\Pi1\ (q1, q2) = q1(120 – (q1+ (57 - q1/2))) - 10 - 3q1$

$\Pi1\ (q1, q2) = q1(120 – q1 - 57 + q1/2) - 10 - 3q1$

$\Pi1\ (q1, q2) = 120q1 - q1^2 - 57q1 + q1^2/2 – 10 - 3q1$

$\Pi1\ (q1, q2) = 60q1 - q1^2/2 – 10$

Find a and b

$d(ax^2 + bx + c)/ dx = 2ax + b = 2(-1/2)q1 + 60 = 60 - q1$

The derivative of $\Pi1(q1, q2)$ with respect to q1 should be 0.

$60 - q1 = 0$

Find q1,

q1 = 60

Find q2,

$q2 = r2(q1)$

$r2(q1) = 57 - q1/2$

$r2(q1) = 57 - 60/2 = 57 – 30 = 27$

q2 = 27

Profit function for firm 1

$\Pi1\ (q1, q2) = 60q1 - q1^2/2 – 10$

Production quantities

q1 = 60

q2 = 27

Task 4

I have selected Deep Reinforcement Learning for Trading paper for task 4.

There are different methods for financial trading. Nowadays, machine learning algorithms are popular. during this paper author discuss about deep reinforcement learning. The matter statement is need good predictive power signal which consistently produce good directional calls. Technical analysis contrasts with fundamental analysis where a security’s historical price data is employed to check price patterns. Technicians trades supported a mixture of indicators like the Relative Strength Index and Bollinger Bands. However, because of the dearth of research on market conditions, these signals predictability isn't strong, often resulting in false breakouts. Algorithmic trading may be a more systematic approach that involves mathematical modelling and automatic execution. RL will be categorized into three methods: critic-only, actor-only, actor-critic approach for trading. Describe three RL algorithm employed in work, Deep Q-learning Network, Policy Gradients, Advantage Actor-Critic methods. Dataset consists of various asset classes most liquid futures contracts including commodities, equity indexes, fixed income, and FX markets. Use Long Short-term Memory neural networks to model both actor and critic networks. Use two-layer LSTM networks with 64 and 32 units in all models, and Leaky Rectifying Linear Units are used as activation functions. As dataset consists of various asset classes, we train a separate model for every asset class. Train models by grouping contracts within the identical asset class will improves performance. Dataset ranges from 15 years so train model at every 5 year, using all data available up to it point to optimise the parameters. Model parameters are then fixed for the subsequent 5 years to supply out-of-sample results. In total, our testing period is from 9 years. Experiments performs on baseline models and three methods. Experiment with an extra layer of portfolio-level volatility scaling is applied for every model and supported different asset classes. RL algorithms gives the higher performance except equity indexes where a long-only strategy is best. DQN performed best among all models and the contender is that the A2C approach. A2C generates larger turnovers, resulting in smaller average returns per turnover.

The experiments derived that the algorithms could follow large market trends without changing positions and might also scale down, or hold, through consolidation periods. Investigate the performance of methods under different transaction costs. Our algorithm can tolerate larger cost rates and DQN and A2C can still produce positive profits. Discrete and continuous action spaces are considered to form reward functions. I found that they adopt classical time series momentum features together with technical indicators to represent state space and acquire trade positions directly using RL. They form a straightforward portfolio by giving equal weights to every contract. The methodology used is suitable from my point of view, but I believed we can add SoftMax, Batch normalization and Dropout layer during this for testing results. we can also try Data augmentation techniques like signal shifting, signal filtering and artificial noise addition. In my point of view methodology is proper but i would like to do with another layers and a few other techniques on same dataset to check which is that the best model for trading.