

# MFIN 706: Assignment 2 – Risk-Based and Robust Portfolio Selection Strategies

Dr. Oleksandr Romanko

June 21, 2016

---

**Due:** Tuesday, July 12, 2016, not later than 7:00 p.m.

Use **MATLAB** for all MFIN 706 assignments.

**You should hand in:**

- Your report.
- Compress all of your MATLAB code files and output files into a file **StudentID.zip** (which can be uncompressed by 7-zip (<http://www.7-zip.org>) software under Windows), and submit it via Avenue to Learn portal no later than 7:00 p.m. on July 12.

**Where to hand in:** Online via Avenue to Learn portal.

---

## Introduction

The purpose of this assignment is to compare computational investment strategies based on selecting portfolio with **equal risk contributions** and using **robust mean-variance optimization**. In Assignment 1, you have already implemented and compared computational investment strategies based on minimizing portfolio variance, maximizing Sharpe ratio as well as “equally weighted” and “buy and hold” portfolio strategies.

You are proposed to test six strategies (use your implementation of strategies 1-4 from Assignment 1):

1. “Buy and hold” strategy;
2. “Equally weighted” (also known as “ $1/n$ ”) portfolio strategy;
3. “Minimum variance” portfolio strategy;
4. “Maximum Sharpe ratio” portfolio strategy;
5. “Equal risk contributions” portfolio strategy;
6. “Robust mean-variance optimization” portfolio strategy.

If a portfolio strategy relies on some targets, e.g., **target return estimation error**, you need to **select those targets consistently for each holding period**. It is up to you to decide how to select those targets.

Feel free to use **MATLAB prototypes** provided in class.

# Questions

## 1. (70 %) Implement investment strategies in MATLAB:

You need to test two portfolio re-balancing strategies:

1. “Equal risk contributions” portfolio strategy: compute a portfolio that has equal risk contributions to **standard deviation** for each period and re-balance accordingly. You can use IPOPT example from the lecture notes, but **you need to compute the gradient of the objective function yourself** (to validate your gradient computations **you may use finite differences method**). The strategy should be implemented in the function `strat_equal_risk_contr`.
2. “Robust mean-variance optimization” portfolio strategy: compute a robust mean-variance portfolio for each period and re-balance accordingly. You can use CPLEX example from the lecture notes, but you need to **select target risk estimation error** and **target return** according to your preferences as an investor (provide justification of your choices). The strategy should be implemented in the function `strat_robust_optim`.

所以也可以用cplex??

Design and implement a rounding procedure, so that you always trade (buy or sell) an integer number of shares.

Design and implement a validation procedure in your code to **test that each of your strategies is feasible** (you have enough budget to re-balance portfolio, you correctly compute transaction costs, funds in your cash account are non-negative).

There is a file `portf_optim2.m` on the course web-page. You are required to complete the code in the file.

Your MATLAB code should use only CPLEX and IPOPT optimization solvers.

## 2. (15 %) Analyze your results:

- Produce the following output for the 12 periods (**years 2005 and 2006**):

```
Period 1: start date 01/03/05, end date 02/28/05
Strategy "Buy and Hold", value begin = $ 1008145.00, value end = $ 994730.00
Strategy "Equally Weighted Portfolio", value begin = $ 1008145.00, value end = ...
Strategy "Minimum Variance Portfolio", value begin = $ 1008145.00, value end = ...
Strategy "Maximum Sharpe Ratio Portfolio", value begin = $ 1008145.00, value end = ...
Strategy "Equal Risk Contributions Portfolio", value begin = $ 1008145.00, value end = ...
Strategy "Robust Optimization Portfolio", value begin = $ 1008145.00, value end = ...

...

Period 12: start date 11/01/06, end date 12/29/06
Strategy "Buy and Hold", value begin = $ 1074110.00, value end = $ 1141435.00
Strategy "Equally Weighted Portfolio", value begin = ..., value end = ...
Strategy "Minimum Variance Portfolio", value begin = ..., value end = ...
Strategy "Maximum Sharpe Ratio Portfolio", value begin = ..., value end = ...
Strategy "Equal Risk Contributions Portfolio", value begin = ..., value end = ...
Strategy "Robust Optimization Portfolio", value begin = ..., value end = ...
```

- Plot one chart in MATLAB that illustrates the daily value of your portfolio (for each of the six trading trading strategies) over the years 2005 and 2006 using daily prices provided. Include the chart in your report.

- Plot one chart in MATLAB to show dynamic changes in **portfolio allocations under strategy 6**. In each chart,  $x$ -axis represents the rolling up time horizon,  $y$ -axis denotes portfolio weights between 0 and 1, and distinct lines display the position of selected assets over time periods. Does your robust portfolio selection strategy reduce trading as compared with strategies 3 and 4 that you have implemented in Assignment 1?
- Compare your “equal risk contributions” and “robust mean-variance optimization” trading strategies **between each other** and **to four strategies implemented in Assignment 1** and discuss their performance relative to each other. Which strategy would you select for managing your own portfolio and why?

### 3. (15 %) Test your trading strategies for **years 2008 and 2009**:

- Download daily closing prices for the same twenty stocks for years 2008 and 2009 (file **MIE1622H\_Assign2\_data\_2008-2009.zip**). If necessary, transform data into a format that your code can read.
- Test six strategies that you have implemented for the 12 periods (years 2008 and 2009). **Use the same initial portfolio that you are given in Assignment 1. Compute your expected returns and covariance matrix for period 1 from the historical data for November-December 2007.** Produce the same output for your strategies as in Question 2.
- Plot one chart in MATLAB that illustrates the daily value of your portfolio (for each of the six trading strategies) over the years 2008 and 2009 using daily prices that you have downloaded. Include the chart in your report.
- Plot three charts in MATLAB for strategies 3, 4 and 6 to show dynamic changes in portfolio allocations using the new data set. Does your robust portfolio selection strategy reduce trading as compared with the strategies 3 and 4?
- Compare and discuss relative performance of your six trading strategies during 2005-2006 and 2008-2009 time periods. Which strategy would you select for managing your own portfolio during 2008-2009 time period and why?

但源数据变成2008-2009年的

why need this

### MATLAB Script to be Completed

```
clc;
clear all;
format long

% Input files
input_file_returns = 'Returns.csv';
input_file_prices = 'Daily_closing_prices.csv';

% Read expected returns
if(exist(input_file_returns,'file'))
    fprintf('\nReading returns datafile - %s\n', input_file_returns)
    fid1 = fopen(input_file_returns);
    % Read instrument tickers
    hheader = textscan(fid1, '%s', 1, 'delimiter', '\n');
    headers = textscan(char(hheader{:}), '%q', 'delimiter', '\t');
    tickers = headers{1}(2:end);
    % Read time periods
    vheader = textscan(fid1, '%q %*[\n]');
    periods = vheader{1}(1:end);
    fclose(fid1);
    data_returns = dlmread(input_file_returns, '\t', 1, 1);
```

```

else
    error('Returns datafile does not exist')
end

% Read daily prices
if(exist(input_file_prices,'file'))
    fprintf('\nReading daily prices datafile - %s\n', input_file_prices)
    fid2 = fopen(input_file_prices);
    % Read dates
    vheader = textscan(fid2, '%q %*[\n]');
    dates = vheader{1}(2:end);
    fclose(fid2);
    data_prices = dlmread(input_file_prices, '\t', 1, 1);
else
    error('Daily prices datafile does not exist')
end

% Initial positions in the portfolio
init_positions = [5000 1000 2000 0 0 0 0 2000 3000 6500 0 0 0 0 0 1000 0 0 0];

% Initial value of the portfolio
init_value = data_prices(1,:) * init_positions;
fprintf('\nInitial portfolio value = $ %10.2f\n\n', init_value);

% Initial portfolio weights
w_init = (data_prices(1,:) .* init_positions)' / init_value;

% Number of periods, assets, trading days
N_periods = length(periods);
N = length(tickers);
N_days = length(dates);

% Convert dates into array [year month day]
format_date = 'mm/dd/yyyy';
dates_array = datevec(dates, format_date);
dates_array = dates_array(:,1:3);

% Number of strategies
strategy_functions = {'strat_buy_and_hold' 'strat_equally_weighted' 'strat_min_variance'
    'strat_max_Sharpe' 'strat_equal_risk_contr' 'strat_robust_optim'};
strategy_names = {'Buy and Hold' 'Equally Weighted Portfolio' 'Minimum Variance Portfolio'
    'Maximum Sharpe Ratio Portfolio' 'Equal Risk Contributions Portfolio'
    'Robust Optimization Portfolio'};
N_strat = 1; % comment this in your code
%N_strat = length(strategy_functions); % uncomment this in your code
fh_array = cellfun(@str2func, strategy_functions, 'UniformOutput', false);

for (period = 1:N_periods)
    % Compute current year and month, first and last day of the period
    if(dates_array(1,1)==5)
        cur_year = 5 + floor(period/7);
    else
        cur_year = 2005 + floor(period/7);
    end
    cur_month = 2*rem(period-1,6) + 1;
    day_ind_start = find(dates_array(:,1)==cur_year & dates_array(:,2)==cur_month, 1, 'first');
    day_ind_end = find(dates_array(:,1)==cur_year & dates_array(:,2)==(cur_month+1), 1, 'last');
    fprintf('\nPeriod %d: start date %s, end date %s\n', period, char(dates(day_ind_start)),
        char(dates(day_ind_end)));

    % Compute expected return and covariance matrix for period 1
    if(period==1)
        mu = data_returns(period,:);
        Q = dlmread(['covariance_' char(periods(period)) '.csv'], '\t', 'A1..T20');
    end

    % Prices for the current day
    cur_prices = data_prices(day_ind_start,:);

```

```

% Execute portfolio selection strategies
for(strategy = 1:N_strat)

    % Get current portfolio positions
    if(period==1)
        curr_positions = init_positions;
        curr_cash = 0;
        portf_value{strategy} = zeros(N_days,1);
    else
        curr_positions = x{strategy,period-1};
        curr_cash = cash{strategy,period-1};
    end

    % Compute strategy
    [x{strategy,period} cash{strategy,period}]=fh_array{strategy}(curr_positions,curr_cash,mu,Q,cur_prices);

    % Verify that strategy is feasible (you have enough budget to re-balance portfolio)
    % Check that cash account is >= 0
    % Check that we can buy new portfolio subject to transaction costs

    %%%%%%%%%% Insert your code here %%%%%%%%%%

    % Compute portfolio value
    portf_value{strategy}(day_ind_start:day_ind_end) = data_prices(day_ind_start:day_ind_end,:) *
                                                         x{strategy,period} + cash{strategy,period};

    fprintf(' Strategy "%s", value begin = $ %10.2f, value end = $ %10.2f\n',char(strategy_names{strategy}),
            portf_value{strategy}(day_ind_start), portf_value{strategy}(day_ind_end));

end

% Compute expected returns and covariances for the next period
cur_returns = data_prices(day_ind_start+1:day_ind_end,:) ./ data_prices(day_ind_start:day_ind_end-1,:) - 1;
mu = mean(cur_returns)';
Q = cov(cur_returns);

end

% Plot results
% figure(1);
%%%%%%%%% Insert your code here %%%%%%%%%%

```

## Sample MATLAB Function for Trading Strategy

```

function [x_optimal cash_optimal] = strat_buy_and_hold(x_init, cash_init, mu, Q, cur_prices)

    x_optimal = x_init;
    cash_optimal = cash_init;

end

```