Nearest Correlation Matrix and Finding the Efficient Frontier

Niklas Forsström, Atlanta Chakraborty

November 2019

1 Introduction

Markowitz's modern portfolio theory tells us how one can form a portfolio with minimum variance for a given expected return, a so called efficient portfolio. The method leverages the covariance between the assets such as to minimize their collective variance. This gives potential investors the incentive to evaluate the covariance matrix for a given set of assets. The use of physical data naturally introduces missing datapoints, something that is commonly resolved by pairwise exclusion. The resulting approximate correlation matrix might not be positive semidefinite, thus sparking the need to project this matrix to the set of correlation matrices. This is more commonly referred to as the nearest correlation matrix problem.

This project was concerned with evaluating the performance of two algorithms designed to solve the nearest correlation matrix problem for a given dataset of historical stock prices. The algorithms that were evaluated are Newton's semismooth method vs primal-dual interior point algorithm. Finally the efficient frontier, in the absence of transaction cost, is found. This is done with the restriction that short-selling is forbidden.

1.1 Markowitz's Modern Portfolio Theory

Any given portfolio can be represented as a vector of portfolio weights w. Each element w_i represents the proportional allocation of asset i and constraints on the vector can be introduced depending on the investors requirements. Since w_i represents the proportion of wealth invested in asset i the vector w_i must sum to 1, this is also known as a budget constraint.

$$e^T w = \sum w_i = 1$$

If further constraints are added, the weights w_i are allowed to be negative, i.e. short selling is allowed. To prevent short selling one can also introduce the constraint

$$w \geq [0, \dots, 0]^T.$$

Markowitz's modern portfolio theory describes how one can construct a portfolio with minimal variance provided an expected return, also known as an efficient portfolio. By assuming constant variance and return of each asset during the period of interest, one can find the variance σ_p^2 and return R_p of the portfolio as:

$$R_p = w^T R = \sum w_i R_i$$

and

$$\sigma_p^2 = \sum_i w_i^2 \sigma_i^2 + \sum_i \sum_{j \neq i} w_i w_j \sigma_i \sigma_j \rho_{ij} = w^T M w$$

where R_i is the return on asset i, σ_i is the standard deviation of the returns for asset i, and ρ_{ij} is the correlation coefficient between the returns on assets i and j. The matrix M is defined by

$$M = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & \sigma_n \end{bmatrix} \times \begin{bmatrix} 1 & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \cdots & \cdots & 1 \end{bmatrix} \times \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & \sigma_n \end{bmatrix}$$

The correlation coefficient for two assets A and B can in practice be estimated with the the Pearson correlation coefficient

$$\rho(A,B) = \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{\overline{A_i-\mu_A}}{\sigma_A}\right)\left(\frac{B_i-\mu_B}{\sigma_B}\right) \tag{1}$$

Where N is the number of observations. Potential vacancies in the dataset can be handled by means of pairwise exclusion. This means that if the observation A_j is missing for some $j \in [1, N]$, the corresponding observation B_j is also neglected and the sum in (1) is evaluated as if these observations never existed. One benefit of this approach is that false assumptions regarding the distributions of the datasets are prevented. One drawback of the method is that the matrix constructed from all combinations of correlation coefficients might not be positive semidefinite. This is because different sets of time-points are used to evaluate the different correlation coefficients.

Once the correlation matrix is obtained finding the optimal allocation of asset weights can be formulated as the following quadratic programming problem. Note that the last inequality prevents short selling and could potentially be relaxed depending on the preferences of the investor. Q denotes the covariance matrix of rates of asset returns.

$$\begin{array}{cl} \min_x & x^TQx \\ \text{such that} & \sum_{i=1}^{n} x_i = 1 \\ & \mu^T x \geq R_p \\ & x_i \geq 0, i = 1, \ldots, n \end{array} \tag{2}$$

Then goal of an investor is to find the curve that connects any expected return with its lowest possible variance, more commonly known as the efficient frontier. We also note that estimating the correlation matrix is a critical step in finding the optimal allocations w and therefore also in finding the efficient frontier.

2 Optimization algorithms

A correlation matrix has the following properties :

(i) symmetric with 1s on the diagonal.

(ii) non negative eigenvalues

(iii) off- diagonal elements between -1 and 1.

2.1 Globalised semismooth Newton's Method

Given $G \in \mathcal{S}^n$, an approximate correlation matrix, consider the nearest correlation matrix problem:

$$(P) \quad \min \left\{ \frac{1}{2}||X-G||^2 \mid \operatorname{diag}(X)=e, X \in \mathcal{S}^n_+ \right\}.$$

Since the constraint set is a closed, convex set, our optimization problem has a unique minimiser. For $X \in \mathcal{S}^n, (y,Z) \in (\mathbf{R}^m, \mathcal{S}^n)$, we can write the Lagrangian function as

$$\begin{aligned} L(X,y,Z) &= \frac{1}{2}||X-G||^2 + \langle y, e-\operatorname{diag}(X)\rangle + \langle Z, -X\rangle \\ &= \langle e,y\rangle + \frac{1}{2}||G||^2 + \frac{1}{2}||X||^2 - \langle G+\operatorname{diag}(y)+Z, X\rangle. \end{aligned}$$

Thus, by setting $\nabla_X L(X,y,Z)=0$, we get

$$X-(G+\operatorname{diag}(y)+Z)=0.$$

On substituting the above X into $L(X,y,Z)$, we get

$$\begin{aligned} \theta(y,Z) &= \min\{L(X,y,Z)|X \in \mathcal{S}^n\} \\ &= \langle e,y\rangle + \frac{1}{2}||G||^2 - \frac{1}{2}||X||^2 \\ &= \langle e,y\rangle + \frac{1}{2}||G||^2 - \frac{1}{2}||G+\operatorname{diag}(y)+Z||^2 \end{aligned}$$

For $\mathcal{K}=\{0^m\}\times\mathcal{S}^n_+$, we have $\mathcal{K}^*=\mathbf{R}^m\times\mathcal{S}^n_+$.Thus the dual problem is given by

$$\begin{aligned} \max\{\theta(y,Z)|(y,Z)\in\mathcal{K}^*\} &= \max\left\{\langle e,y\rangle - \frac{1}{2}||G+\operatorname{diag}(y)+Z||^2 | y\in\mathbf{R}^m, Z\in\mathcal{S}^n_+\right\} + \frac{1}{2}||G||^2 \\ &= \min\left\{-\langle e,y\rangle + \frac{1}{2}||G+\operatorname{diag}(y)+Z||^2 | y\in\mathbf{R}^m, Z\in\mathcal{S}^n_+\right\} \\ &= \min\langle -e,y\rangle + \min\left\{\frac{1}{2}||G+\operatorname{diag}(y)+Z||^2 | Z\in\mathcal{S}^n_+\right\} \end{aligned}$$

The dual problem is thus equivalent to

$$(D') \quad \min\left\{h(y) := -\langle e,y\rangle + \frac{1}{2}||\Pi_+(G+\operatorname{diag}(y))||^2 \mid y\in\mathbf{R}^m\right\}$$

where $\Pi_+(.)$ denotes the projection onto the cone $\mathcal{S}^n_+$.

The KKT conditions for the primal and dual problems are given as follows:

$$X-G-\operatorname{diag}(y)-Z=0,$$

$$-Z\in N_{\mathcal{S}^n_+}(X) \iff \mathcal{S}^n_+ X \perp Z \in (\mathcal{S}^n_+)^* = \mathcal{S}^n_+$$

The last complementarity condition can be shows to be equivalent to

$$X=\Pi_+(X-Z).$$

First we note that the function $F(y) := -e + \text{diag}(\Pi_+(G + \text{diag}(y)))$ is semismooth for all $y \in \mathbf{R}^m$. Solving the Newton linear system

$$H_k \Delta y = -\nabla h(y^k)$$

where $H_k = \text{diag}(W_k \text{diag}()) \in \hat{\partial}^2 h(y^k)$ and $W_k \in \partial\Pi_+(G+\text{diag}(y^k))$ is the most crucial step in the algorithm. Consider the spectral decomposition of $U^k := G + \text{diag}(y^k) = Q\text{diag}(d)Q^t$ where the eigenvalues d are arranged in a descending fashion such that $d_1 \geq \cdots \geq d_r > 0 \geq d_{r+1} \geq \cdots \geq d_n$. Pick an element $W_k \in \hat{\partial}^2 h(y^k)$ such that

$$W_k[X] = Q(\Omega \circ (Q^t X Q))Q^t$$

where

$$\Omega_{ij} = \begin{cases} 1 & \text{if } 1 \leq i, j \leq r \\ \frac{d_i}{d_i - d_j} & \text{if } 1 \leq i \leq r, r+1 \leq j \leq n \\ \frac{d_j}{d_j - d_i} & \text{if } r+1 \leq i \leq n, 1 \leq j \leq r \\ 0 & \text{if } r+1 \leq i, j \leq n. \end{cases}$$

With this choic of W_k, one can now solve the Newton linear system approximately via the conjugate gradient method.

2.2 SDP

Our optimization problem can be written as

$$\min \left\{ ||X - G||_F \mid \text{diag}(X) = e, X \in \mathcal{S}^n_+ \right\}.$$

The Frobenius norm in the objective function prevents the problem from being formulated as a conic linear optimization problem. We thus make use of the symmetry of G and X to get a compact vector representation, i.e. make use of the mapping `svec` : $\mathbf{R}^{n \times n} \to \mathbf{R}^{n(n+1)/2}$ from a symmetric matrix to a flattened vector containing the (scaled) lower triangular part of the matrix:

$$\texttt{svec}(A) = (\alpha_{11} A_{11}, \alpha_{21} A_{21}, \alpha_{22} A_{22}, \ldots, \alpha_{nn} A_{nn})$$

where

$$\alpha_{ij} = \begin{cases} 1 & \text{if } j = 1 \\ \sqrt{2} & \text{if } j < i \end{cases}$$

Thus $||A||_F = ||\texttt{svec}(A)||_2$. This leads to an conic linear optimization problem [3], and we make use of the `sqlp` function [4] in `sdpt3`.

3 Dataset

The dataset, which is displayed in Figure 1, contains stock data from 1516 Nordic companies. The stocks have one datapoint per business day from $2017 - 01 - 01$ to 101 business days later. Roughly 3.8% of the datpoints in the dataset are missing, which is the reason why the approximated correlation matrix won't fulfill the conditions of a correlation matrix.

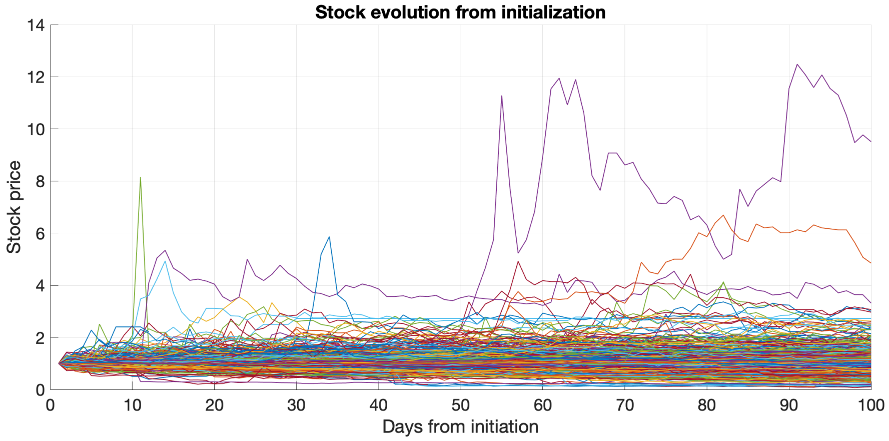

Figure 1: *The dataset of normalized price-paths for 1516 Nordic companies starting from 2017-01-01 and ending 101 business days later. The dataset contains roughly* 3.8% *missing data.*

The sample variance along with the daily average of the returns was evaluated for the assets in Figure 1. Figure 2 illustrates the average daily return along with the sample variance for the assets under 3 different scopes.

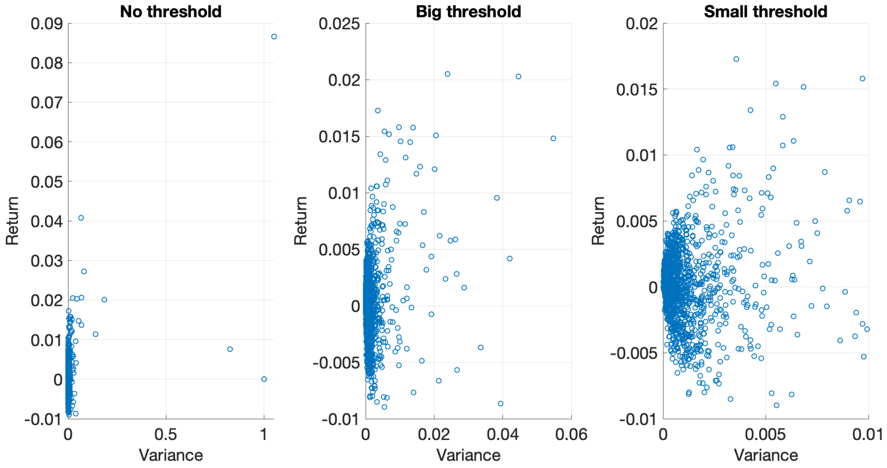

Figure 2: *Average daily returns along with variance of daily returns for the 1516 assets in figure 1. The different subfigures represents different thresholds for the maximal tolerated variance. The big threshold is set to 0.05 and the small threshold is set to 0.01.*

The biggest threshold includes all 1516 assets, but note that the most extreme values are unrealistic and might be due to effects such as stock splits etc. To limit the effect of these outliers a maximum threshold of 0.01 was set for the sample variance. All assets with a variance above the threshold were neglected in the proceeding analysis.

The quantity of interest is the correlation coefficient between the daily returns of the stocks. The daily return of a stock S_t at time t is defined as $S_t - S_{t-1}$. The correlation between the time series of 100 daily returns was determined using pairwise exclusion, and the resulting matrix is thus an approximation of a correlation matrix. An important scenario, which is depicted in Figure 3, is when two assets have no overlapping datapoints. For these combinations a correlation can't be computed and therefore the corresponding coefficient was imputed as zero.

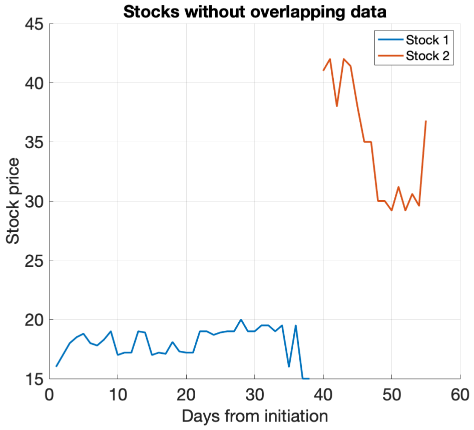

Figure 3: *Stock prices for two assets with no overlapping datapoints.*

4 Results

Following the discussions on the procedure to obtain the correlation matrix, we find that the matrix is not positive semidefinite. The eigenvalues of the matrix range from -14.68972 to 80.54129, out of which 768 of these eigenvalues are negative, which is quite a significant number for a matrix of dimension 1516. Using this as an approximate correlation matrix i.e. G, we first run globalised semi-smooth Newton's method as described in the previous section.

iter	gradient	primal objective	dual objective	reldualitygap
0	1.247e+01	3.122e-01	6.691e+02	4.614e-01
1	4.903e+00	6.164e+02	4.835e+02	1.207e-01
2	1.524e+00	6.035e+02	5.8e+02	1.986e-02
3	3.593e-01	6.019e+02	5.997e+02	1.764e-03
4	4.328e-02	6.017e+02	6.017e+02	4.088e-05
5	1.051e-03	6.017e+02	6.017e+02	6.533e-07

Final Dual Objective Function value: 6.01707e+02
Final primal Objective Function value: 6.01708e+02
Computing time for linear systems solving (cgs time): 1.3546e+01 secs
Computing time for eigenvalue decompositions: 2.75658e+01 secs
Computational time: 4.97949e+01 secs

The first $5*5$ sub-matrix of the nearest correlation matrix solution obtained is as follows:

	247SO	2CUREX	3KR	A1M	AAB
247SO	1.00000000	0.7291150	0.63087122	-0.4242252	-0.08894738
2CUREX	0.72911500	1.0000000	0.62262139	-0.3929131	-0.27713132
3KR	0.63087122	0.6226214	1.00000000	-0.6940429	-0.04770341
A1M	-0.42422518	-0.3929131	-0.69404291	1.0000000	0.35955348
AAB	-0.08894738	-0.2771313	-0.04770341	0.3595535	1.00000000

where the first row indicates 5 different stocks, and the table contains their associated correlations.

The above obtained matrix is symmetric and all of its entries are within $[-1, 1]$, with 1 along its diagonal. The eigenvalues of this matrix range from $-2.043063e-12$ to $8.012790e+01$. Thus we can see that the solution obtained is indeed a correlation matrix which is positive semi-definite.

We also ran the primal-dual interior point algorithm via the SDP solver as previously mentioned. However, we could not find a convergent solution in 100 iterations through the SDP solver, even thought the SDP solver was fast enough for smaller matrix dimensions! This is where the semismooth Newton's method gives an added advantage for matrix of larger dimensions.

We can thus conclude that it is feasible to compute nearest correlation matrix for problems for which spectral decomposition can be done, and if the matrix is of higher dimensions, globalised semismooth Newton's method performs the best!

5 Financial interpretations and discussion

After computing the nearest correlation matrix the quadratic programming problem (2) was optimized for a number of expected returns, and the efficient frontier is displayed in Figure 4.

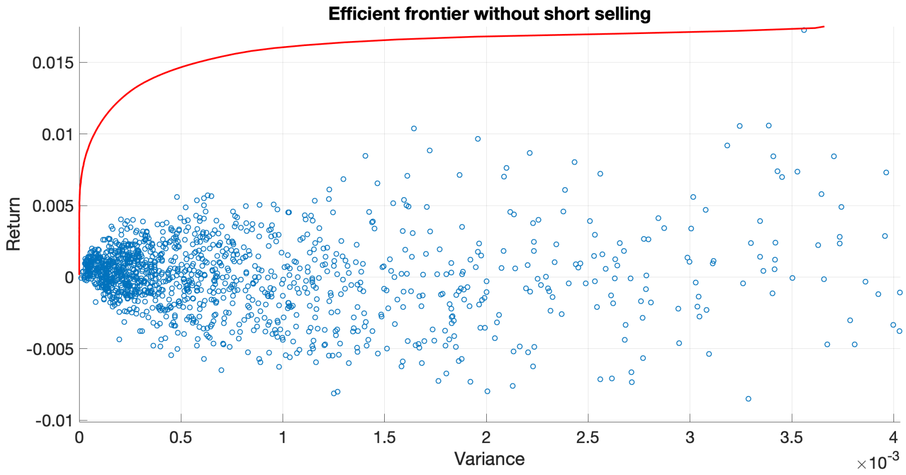

Figure 4: *Efficient frontier for the dataset when short selling is not allowed.*

From Figure 4 we note that the efficient frontier greatly outperforms any given asset. One reason behind the impressive looking result is the very large number of assets, and hence the ability to find negatively correlated assets. It should be noted that the efficient frontier is derived in the absence of transaction cost, and one would expect this to yield a large number of non-zero portfolio weights with small magnitudes. Figure 5 shows an empirical CDF over the portfolio weights for an expected return of 0.012. From the figure we note that the majority of portfolio weights are of the magnitude 10^{-7}.

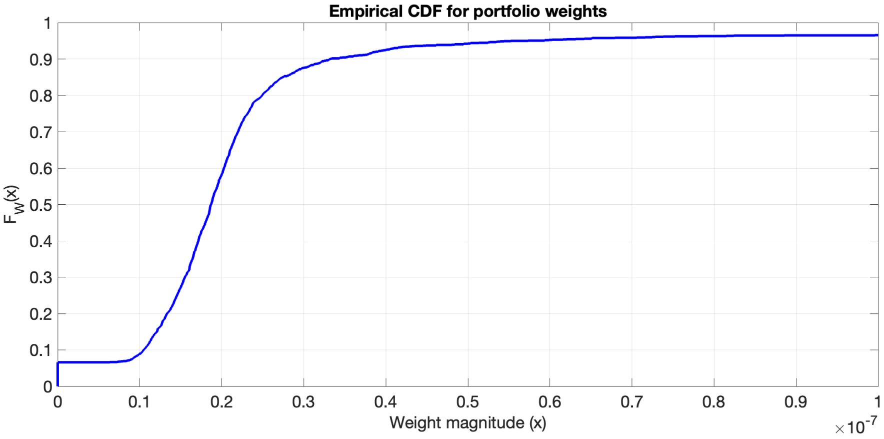

Figure 5: *Empirical CDF of the portfolio weights for an expected return of 0.012. The figure illustrates that the majority of portfolio weights are of the magnitude* 10^{-7}

When transaction costs are present a strategy like the one in Figure 5 becomes unreasonable. To encode the preference for a sparse portfolio one could include various types of penalty functions in the objective function (2). A natural candidate, if not for the budget constraint, would be the ℓ_1-penalty function. However, the presence of the budget constraint renders the ℓ_1-penalty useless [6]. It would therefore be interesting to further investigate the effects of other penalty functions.

Another reason as to why Figure 4 yields an impressive result is due to the in-sample testing. A more revealing evaluation of the portfolio strategy is displayed in Figure 6, which is using the same strategies that were obtained from Figure 4, but evaluated on the preceding 100 business days. In other words it's an out of sample test for the portfolio strategies.

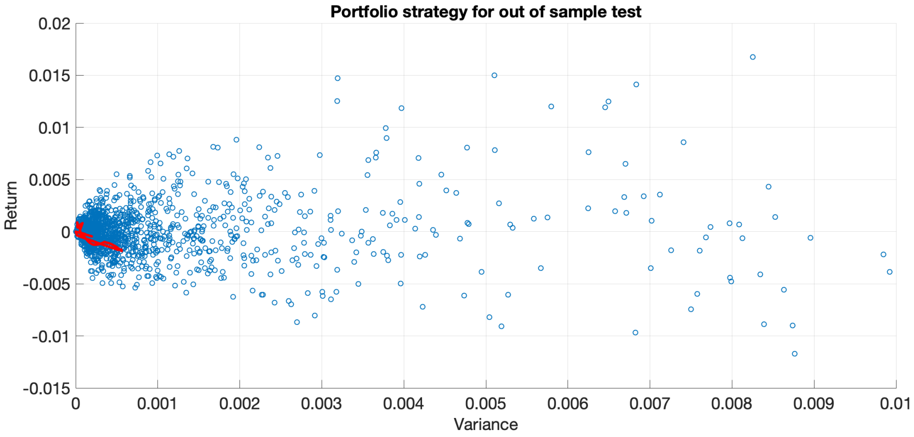

Figure 6: *out of sample test*

We note from Figure 6 that the performance does not look satisfactory since the majority of our portfolio strategies yield a negative return. This is not surprising given the simple model that the assets are assumed to follow. However, using this same approach one could evaluate the performance of a portfolio obtained through mean variance analysis based on a more sophisticated model.

6 MATLAB code

```matlab
% Niklas Forsstroem
% Final assignment script
%% plot the dataset
clf
grid on, hold on;
plot(raw_data(1:100,:)./raw_data(1,:),'linewidth',1)
%l = legend({'Barrier put option','Vanilla put option'},'Fontsize', 15);
xlabel('Days from initiation', 'Fontsize', 20)
ylabel('Stock price', 'Fontsize', 20)
title('Stock evolution from initialization', 'Fontsize', 20)
set(gca,'FontSize',20)
%% Plot 2 stocks with no overlapping data
clf
grid on, hold on;
plot(raw_data(1:100,620),'linewidth',2)
plot([nan(39,1);raw_data(40:100,167)],'linewidth',2)
l = legend({'Stock 1','Stock 2'},'Fontsize', 15);
set(gca,'FontSize',20)

xlabel('Days from initiation', 'Fontsize', 20)
ylabel('Stock price', 'Fontsize', 20)
title('Stocks without overlapping data', 'Fontsize', 20)
%%
%==============================EFFICIENT FRONTIER PART==============================
% This part is to be used with the pre-constructed correlation matrix only
% You should use the MARKOWITZ WORKSPACE!!
% note that all vectors are row vectors due to correlation matrix

% note that the training set is using the first 101 datapoints of the
% raw_data - dataset (after normalizing)

% VARIABLE DESCRIPTIONS:

%=============================================================================
%% Scatter of the individual stocks
sigma = std(differences(1:100,:))';
returns = mean(differences(1:100,:))';

% for some stocks ve cant evaluate sigma and return, impute with large
% values for sigma and 0 for return
sigma(isnan(sigma)) = 1;
returns(isnan(returns)) = 0;


% find all stocks with variance > threshold to be excluded (small: 0.1, big: 0.25)
```

```
threshold = 0.1;
include = sigma < threshold;


% plot the stocks below the threshold
%subplot(1,3,2)
hold off
scatter(sigma(include).^2, returns(include));

set(gca,'FontSize',20)
title('Small threshold', 'Fontsize', 20)
xlabel('Variance', 'Fontsize', 20)
ylabel('Return', 'Fontsize', 20)
grid on
%% Calculate the efficient frontier (using quadprog)
% ==========================================================================
% Uses matlabs quadprog function to solve the mean variance problem for a
% set of expected returns yeilding the efficient frontier. It uses the
% correlation matrix obtained in previous steps
% ==========================================================================

hold on

%fortfolio variance as a function of portfolio weights
M = diag(sigma)*training_correlations* diag(sigma);
Aeq = [returns'; not(include)'; ones(1,1516)]; % equality constraint


datapoints = 50; % number of points constituting efficient frontier
vars = zeros(datapoints,1); % vector of variances for eeach portfolio
weights = zeros(1516,datapoints); % each column is a portfolio
exp_return = linspace(0.0002,0.02,datapoints);
for i = 1:datapoints
    beq = [exp_return(i);0;1]; % equality constraint
    weights(:,i) = quadprog(M,zeros(1516,1),-eye(1516),zeros(1516,1),Aeq,beq); % find efficient portf.
    vars(i) = weights(:,i)'*M*weights(:,i); %calculate variance of the potf.
end

%% plot the efficient frontier
hold on
plot(vars, exp_return,'r','linewidth', 2)
set(gca,'FontSize',20)
```

```
xlabel('Variance', 'Fontsize', 20)
ylabel('Return', 'Fontsize', 20)
title('Efficient frontier without short selling', 'Fontsize', 20)
grid on

%% plot cumulative disteibution of portfolio weigths
% ===========================================================================
% plots the empirical cdf of the portfolio weights for some given
% index ? [1,datapoints]. this is to study the magnitude of most weights
% ===========================================================================
index_to_plot = 30 %the index of interest (index ? [1,datapoints])

my_plot = cdfplot(weights(:,30));
set( my_plot, 'LineWidth', 3, 'Color', 'b');
axis([0 10e-8 0 1]);
set(gca,'FontSize',20)
xlabel('Weight magnitude (x)', 'Fontsize', 20)
ylabel('F_W(x)', 'Fontsize', 20)
title('Empirical CDF for portfolio weights', 'Fontsize', 20)
grid on


%% Out of sample testing using the portfolio strategy
% Scatter of the individual stocks
sigma = std(differences(101:200,:))';
returns = mean(differences(101:200,:))';

% for some stocks ve cant evaluate sigma and return, impute with large
% values for sigma and 0 for return
sigma(isnan(sigma)) = 0;
returns(isnan(returns)) = 0;


% find all stocks with variance > threshold to be excluded (small: 0.1, big: 0.25)
threshold = 0.1;
include = sigma < threshold;


% plot the stocks below the threshold
%subplot(1,3,2)
hold off
scatter(sigma(include).^2, returns(include));

set(gca,'FontSize',20)
title('Small threshold', 'Fontsize', 20)
xlabel('Variance', 'Fontsize', 20)
```

```
ylabel('Return', 'Fontsize', 20)
grid on

% Use the portfolio weights from before to evaluate the risk and return of
% the different portfolios

tested_portf_returns = returns'*weights;
M_test = diag(sigma)*testing_correlations* diag(sigma);
tested_portf_var = diag(weights'*M_test*weights);
%% plot the supposedly efficient frontier
hold on
plot(tested_portf_var, tested_portf_returns,'r','linewidth', 3)
set(gca,'FontSize',20)
xlabel('Variance', 'Fontsize', 20)
ylabel('Return', 'Fontsize', 20)
title('Portfolio strategy for out of sample test', 'Fontsize', 20)
grid on
```

References

[1] Higham, N.J., *Computing the nearest correlation matrix—A problem from finance,* IMA J. Numer. Anal., 22(3):329–343, 2002.

[2] Qi, H.D., Sun, D. *A quadratically convergent Newton method for computing the nearest correlation matrix,* SIAM J. Matrix Anal. Appl., 28(2):360–385, 2006.

[3] Rahman, A. *The Nearest Correlation Matrix* `https://cran.r-project.org/web/packages/sdpt3r/vignettes/nearcorr.pdf` 2019

[4] Toh, K.C., Todd, M.J., Tutuncu, R.H., *SDPT3- a Matlab software package for semidefinite-quadratic-linear programming,* Optimization methods and software, 2001.

[5] Toh, K.C. *Theory and Algorithms for Non linear programming, NUS Module* Aug- Dec 2019.

[6] P.J. Kremer, S. Lee, M. Bogdan, S. Paterlini *Sparse Portfolio Selection via the sorted ℓ_1- Norm,* arXiv, 1710.02435, 2017.