Portfolio Optimization

Problem Presenter: Randall Selkirk (Mapleridge Capital Corporation)

Academic Participants: Nabeel Butt (University of Western Ontario), Haohan Huang (York University), Kin Hung Kan (University of Western Ontario), Lung Kwan Tsui (University of Pittsburgh), John Chadam (University of Pittsburgh)

Report prepared by: Lung Kwan Tsui[1]and John Chadam[1]

Abstract. Two strategies are devised to maximize the Sharpe ratio of a portfolio consisting of 35 risky assets. The first one uses periodically updated optimal weights from standard Markowitz/Sharpe portfolio theory. The second strategy removes a fixed number of assets that have highest positive correlation with the rest of the portfolio. Both approaches perform better (have larger Sharpe ratio) than the existing strategies.

1 The Problem

The challenge presented by the Mapleridge team was to devise an allocation strategy among 35 risky assets that they had identified with the goal of maximizing the Sharpe ratio over a prescribed time horizon. Their interest in the project arose from the counter-intuitive results that they had obtained in house. Specifically, using the optimal allocation weights (i.e., obtained via the Markowitz/Sharpe procedure) on out-of-sample data, produced returns with a Sharpe ratio that was exceeded by less sophisticated strategies such as equal weighting among the risky assets (the maximum entropy - least information strategy) and weighting each asset by the inverse of its standard deviation (see Figure 1). The question then is why did this counter-intuitive result occur and the challenge was to use this information to devise allocation strategies to improve performance.

2 Background

Suppose one can invest (long and short) in N risky assets S_i ($N = 35$ here). Suppose that the expected returns of the individual assets in the time interval $[0, T]$ are μ_i, $i = 1, ..., N$ and the covariance matrix of these returns is $C = [\sigma_{ij}]$, $i, j = 1, ..., N$. If one creates a portfolio of these N risky assets by allocating the proportion (weight) w_i of the total
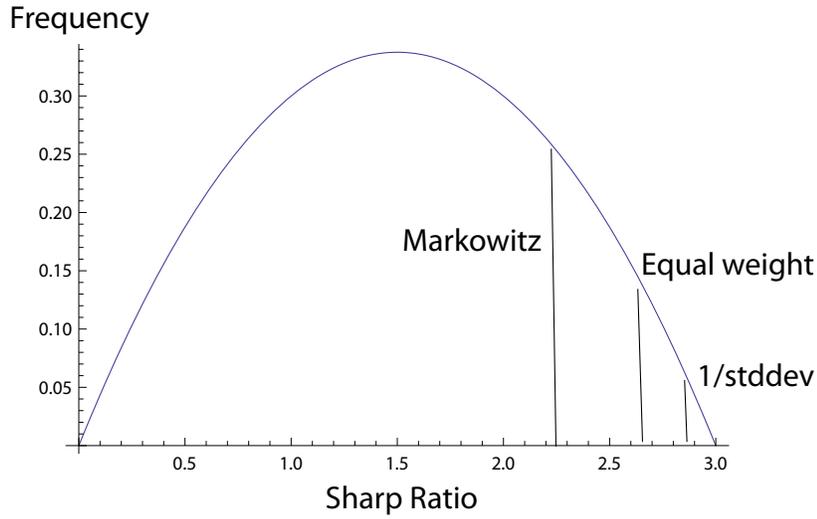

Frequency

Figure 1 The frequency distribution of Sharpe ratios for portfolios with random allocation among 35 assets highlighting the values for the Markowitz optimal weights, equal weights and inverse standard deviation weights.

investment to the i-th asset, then the expected return on this portfolio at the horizon T is

$$\Pi_p = \sum_{i=1}^{N} w_i u_i \tag{2.1}$$

with the standard deviation of the portfolio returns given by

$$\sigma_p = \sqrt{\sum_{i,j=1}^{N} \sigma_{ij} w_i w_j} \tag{2.2}$$

A common risk measure that one might consider maximizing in the allocation procedure is the Sharpe ratio Π_p/σ_p. Indeed, Markowitz/Sharpe portfolio theory is built on the result that the weights w_i, $i = 1, ..., N$ obtained from the optimization problem

$$\max_{w_i} \frac{\sum_{i=1}^{N} w_i \mu_i - r_f}{\sqrt{\sum_{i,j=1}^{N} \sigma_{ij} w_i w_j}} \tag{2.3a}$$

subject to

$$\sum_{i=1}^{N} w_i = 1 \tag{2.3b}$$

where r_f is the (T period) return on a risk-free bond (that is also available for investment) provides the weights for the so-called capital market portfolio. By investing in this specific

portfolio of the risky assets and the bond the investor obtains the best return for a given risk (alternatively, the lowest risk for a given return) among all portfolios involving the bond and N risky assets.

In applying this theoretical framework to their data the team from Mapleridge found that they did not obtain the expected optimal performance. Beginning with 2606 days of daily returns for each of their 35 risky assets, they used the first 1303 days to estimate the values of μ_i and σ_{ij}. The Markowitz-Sharpe weights, w_i^M, for the capital market portfolio obtained from the optimization problem with $r_f = 0$ were used to construct a portfolio of the 35 risky assets and the performance was calculated using the last half (out-of-sample) data. The counter-intuitive result was that the maximum Sharpe ratio was not obtained with this portfolio. In fact, as Figure 1 shows, the portfolio that requires no knowledge about the asset returns, $w_i = 1/N = 1/35$, $i = 1, .., 35$, provided a higher Sharpe ratio and the portfolio with $w_i = (\sigma_{ii})^{-1}(\sum_{j=1}^{N} 1/\sigma_{jj})^{-1}$ does even better. Notice that the latter attempts to minimize the risk by penalizing the riskiest assets but does not take advantage of knowledge of the expected returns or the correlation dependence, σ_{ij}, $i \neq j$, of the assets.

3 Two Approaches

It is well known that the Markowitz/Sharpe optimization procedure (2.3) to obtain the weight w_j^M is quite sensitive to the underlying assumptions and data. For example, if the returns are not Gaussian, then skewness and kurtosis corrections are required [1]. Moreover, errors in estimating μ_i and σ_{ij} from the data lead to large variations in the values of w_i^M [2]. Finally, if the data is not stationary, it is difficult to use data from the past to implement strategies for the future. The team began the project by using the standard tests and, as expected with real-world data, it possessed all three of the above deficiencies.

During the week the team came up with two approaches to these problems arising from the data. The first is a straightforward, direct attempt to minimize the effect of the above difficulties arising from the data. The idea is to periodically update the optimal weights from (2.3) using the best possible estimates of the parameters μ_i, σ_{ij} at that time. Operationally, there are two time scales to be determined - the reallocation times and the lookback interval over the past data that provides the best estimate for the input parameters μ_i, σ_{ij} to the optimization procedure (2.3). Of course, frequent reallocation can lead to prohibitive transaction costs. In discussions with the Mapleridge team it was agree that transaction costs could be adequately modeled by simply reducing the portfolio value at a reallocation time t_j by a fixed percentage, 0.04%; i.e.,

$$\Pi(t_j+) = (1 - 0.0004)\Pi(t_j-) \tag{3.1}$$

Two independent computer codes (one is attached as an Appendix) were written to assure the validity of our numerical experimentations. The first check was to reproduce the original, single period, results of the Mapleridge team. Next, ad hoc experimentation with the codes indicated that optimal performance was achieved with a reallocation period of 20 days and a lookback interval of 300 days (see Figure 2). Using data from the more distant past distorts the estimates of the parameters used in the optimization step (2.3).

The second approach is based on the observation that although the yearly (250-day) returns fluctuate wildly, their covariances do not. The implication is that even though the
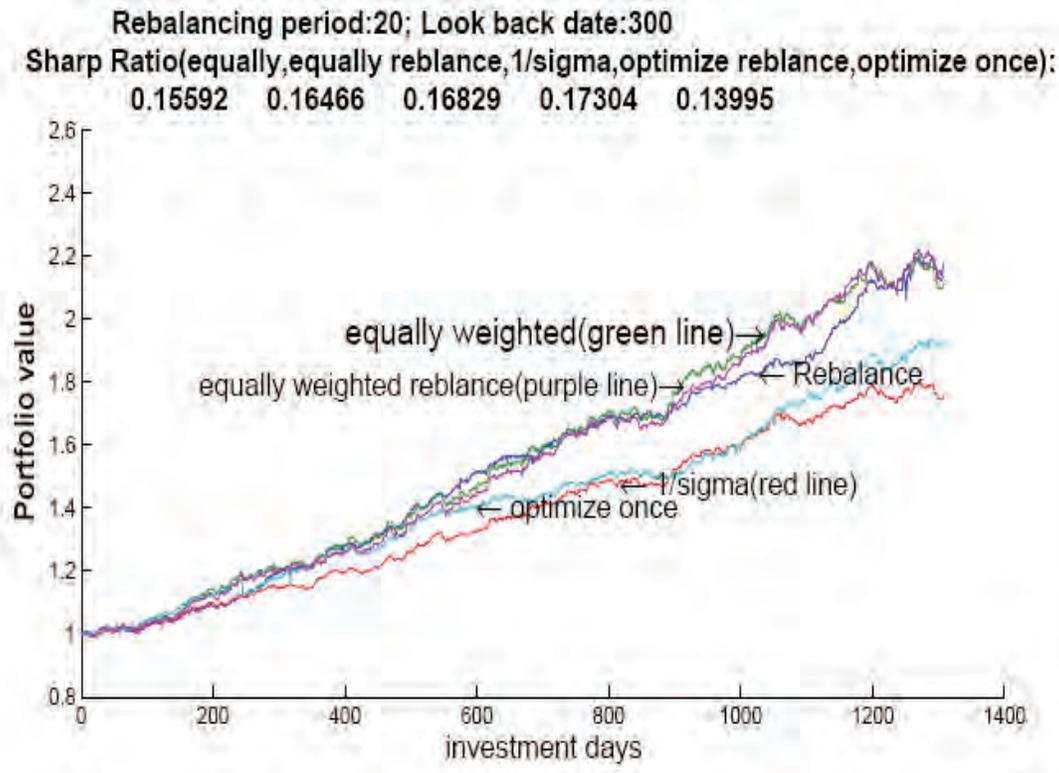
Figure 2 Time series of the portfolio value starting from $1 for different allocation methods. The order of magnitude of the Sharpe ratio appearing at the top are different from those in Fig. 1 since different scaling is used.

data is not stationary, it does provide reasonable estimates of the covariances. The risk-lowering strategy that is developed here is to remove those risky asset that contribute the most positively correlated risk. That is, at each reallocation time (taken to be 1 year) we completely remove from the portfolio a fixed number of risky assets, S_i, for which $\sum_{j \neq i}^{N} \sigma_{ij}$ has the largest positive value. This reallocation process is repeated each year using updated data to calculate the covariance matrix. Numerical experimentation with this covariance reduction method suggests that the highest Sharpe ratio was obtained when about 7 of the assets with the highest positive correlations were removed each year (See Figure 3). In spite of diminishing performance with additional removals, this covariance reduction method still out-performed the equally weighted portfolio when as many as 17 assets were removed.

4 Conclusions

The real-world data provided by Mapleridge on the daily returns of 35 proprietary assets were, as expected, neither Gaussian nor stationary. Moreover, the annual returns fluctuated wildly. Evidently the extent of the departure from these idealized assumption that underlie Markowitz Portfolio Theory was sufficient to negate the optimality of its performance. Two approaches were proposed and examined to improve performance (increase the Sharpe ratio) by reducing the effects of the above deficiencies in the data. Both were based on periodic reallocation in the portfolio using updated estimates of the relevant parameters required
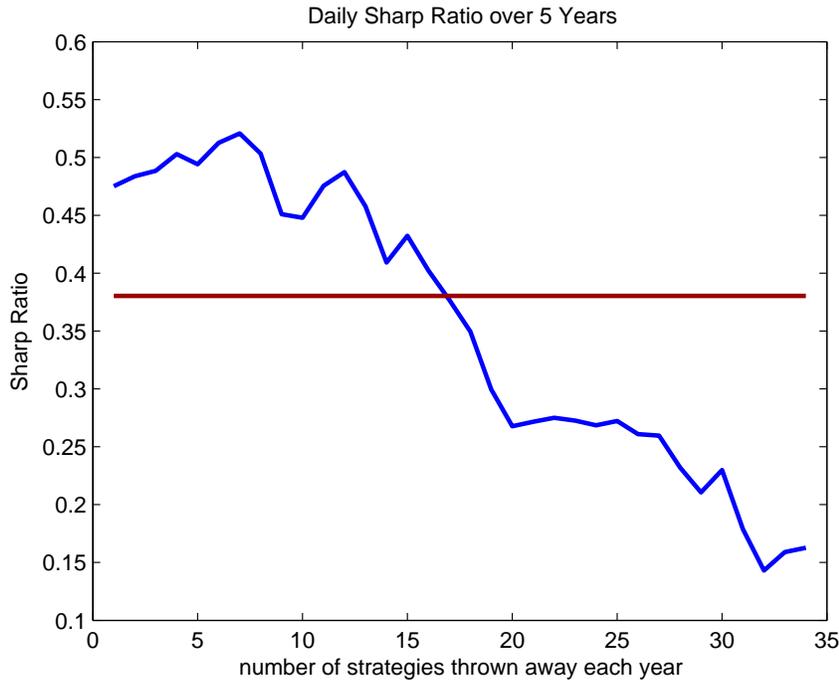
Figure 3 Sharpe ratio obtained from the reallocation process which removes assets with the highest positive correlations.

for the particular reallocation procedure. The first simply used updated values for the weights of the capital market portfolio as in the standard Markowitz approach while the second removed from the portfolio (assigned a zero weight to) those assets S_i that had the highest positive values of $\sum_{j \neq i}^{35} \sigma_{ij}$. Both approaches are ad hoc and the results, though convincing, are preliminary. This is, of course, due to the short amount of time available to the Mitacs team. Some longer term projects that were suggested by this preliminary study include incorporation of higher moments to overcome the non-normality of the returns and a formulation of the optimal reallocation time as a stopping time problem. Finally, the results for maximizing the Sharpe ratio could be compared with those using other measures such as Sortino ratio, maximum drawdown, etc. The Sharpe ratio may not be the optimal performance measure for non-Gaussian returns [3].

5 References

1. C.R. Harvey, J. C. Liechty, M.W. Liechty, and P. Müller, Portfolio Selection with Higher Moments, Working Paper W70, Fuqua School of Business, Duke University, July 2003.
2. F.J. Fabozzi, P.N. Kolm, D. Pachamanova and S.M. Focardi, Robust Portfolio Optimization and Management, John Wiley and Sons, 2007.
3. D. Sornette, P. Simonetti and J.V. Anderen, "Nonlinear" Covariance Matrix and Portfolio Theory for Non-Gaussian Multivariate distributions, Physics Report 335, no.2, 19-92, 2000.

Appendix A: Matlab Code

A.1 mainfunction.m

```
digits(10)
global PP;
PeriodVector=[20,20,20,20,20,20,60,60,60,60,125,125,125,250,250,250];
PPVector=[1300,600,300,120,80,40,1300,600,300,120,1300,600,300,1300,600,300];

% Period=60;
% SR_interval=1;
% StopNumber=1300;

for N=7:16
Period=PeriodVector(N);
SR_interval=1;
StopNumber=1300;
PP=PPVector(N)-Period;
[SR0,Truely_Price0]=Trans(Period,SR_interval,StopNumber);
disp('Period,SR_interval,SR');
[Period,SR_interval,SR0];
% SR0*sqrt(256)/sqrt(SR_interval)

% n=length(Truely_Price0);
% Truely_Price(n)
[SR1,Truely_Price1]=Equallyweighted(SR_interval,StopNumber);
SR1;


% figure('2')
[SR2,Truely_Price2]=one_over_sigma(SR_interval,StopNumber);
SR2;


[SR3,Truely_Price3]=Opone(SR_interval,StopNumber);
SR3;
hold all;
plot(Truely_Price0)
plot(Truely_Price1)
plot(Truely_Price2)
plot(Truely_Price3)
text(1200,Truely_Price0(1200),'\leftarrow Rebalance',...
     'HorizontalAlignment','left')
 text(1200,Truely_Price1(1200),'\leftarrow equally weighted',...
     'HorizontalAlignment','left')
 text(1200,Truely_Price2(1200),'\leftarrow 1/sigma',...
     'HorizontalAlignment','left')
```

```
text(1200,Truely_Price3(1200),'\leftarrow optimize once',...
    'HorizontalAlignment','left')



[SR4,Truely_Price4]=Second_Trans(Period,SR_interval,StopNumber);
plot(Truely_Price4)
 text(1200,Truely_Price4(1200),'\leftarrow equally weighted reblance',...
    'HorizontalAlignment','left')
tt=strcat('Rebalancing period:   ', num2str(Period));
tt2=strcat('; Look back date:    ',num2str(PPVector(N)));
tt3=strcat('Sharp Ratio(equally,equally reblance,1/sigma,...
optimize reblance,optimize once): ',num2str([SR1,SR4,SR2,SR0,SR3]));
ttt=strcat(tt,tt2);
tt5=sprintf('%s \n %s',ttt,'Sharp Ratio(equally,equally reblance,1/sigma,...
optimize reblance,optimize once): ');
title(sprintf('%s \n %s',tt5,num2str([SR1,SR4,SR2,SR0,SR3])));
save_name=strcat(strcat('figure',num2str(N)),'.jpg');
 saveas(gcf,save_name);
 hold off;
end
```

A.2 Trans.m

```
function [SR,Truely_Price]=Trans(Period,SR_interval,StopNumber)
global PP;
% Input:
% Period:  reallocate period, unit: Number of days!
% SR_interval;%Sharp ratio interval, has to be less than StopNumber
% (default 500!), unit: Number of days!
% StopNumber:Default 500, the number of days we used as historical
% data, unit: Number of days!
% Output:
% SR: sharp ratio of Period we need
%$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$



% Period=256;% reallocate period 1 yr
% SR_interval=125;%Sharp ratio interval,
% has to be less than StopNumber/2 (default 500!)
AllReturn=xlsread('Fields_Mitac_MapleridgeDataSet.csv');
[m,n]=size(AllReturn);
No_of_Asset=n;
No_of_Day=m;
% StopNumber=500;
for i=1:No_of_Day-Period
    % The return for each strategy between the period
    Period_Return(i,:)=prod(1+AllReturn(i:i+Period-1,:))-1;
```

```
end

% PP=1300-Period;
Old=Period_Return(StopNumber-Period+1-PP+1:StopNumber-Period+1,:);
correlation_old=corrcoef(Old);
mean_old=mean(Old);
std_old=std(Old);

H = ones([1,No_of_Asset]);
%iniValues=H*1/No_of_Asset;
LB = zeros([1,No_of_Asset]);
UB =ones([1,No_of_Asset]);
Quadraticfun = @(x)(-mean_old*x)/sqrt((std_old'.*x)'...
*correlation_old*(std_old'.*x));
opts = optimset('fmincon');
opts.LargeScale='off';
opts.Display='off';
Aeq=H;
beq=[1];
highfval=0;
for i=1:1
    iniValues=randn(1,No_of_Asset).*0.1';
    [x,fval] =fmincon(Quadraticfun,iniValues',[],[],...
    Aeq,beq,LB',UB',[],opts);
     if fval<=highfval
         highfval=fval;
         optX=x;
     end
end
Weight(:,1)=optX;
k=1;
while(StopNumber+1+Period*k<No_of_Day)
    Old=Period_Return(StopNumber+Period*k-Period+1-PP+1:...
    StopNumber+Period*k-Period+1,:);
    correlation_old=corrcoef(Old);
    mean_old=mean(Old);
    std_old=std(Old);
    H = ones([1,No_of_Asset]);
    %iniValues=H*1/No_of_Asset;
    LB = zeros([1,No_of_Asset]);
    UB =ones([1,No_of_Asset]);
    Quadraticfun = @(x)(-mean_old*x)/sqrt((std_old'.*x)'...
    *correlation_old*(std_old'.*x));
    highfval=0;
    for i=1:1
        iniValues=randn(1,No_of_Asset).*0.1';
        [x,fval] =fmincon(Quadraticfun,iniValues',[],[],...
```

```
            Aeq,beq,LB',UB',[],opts);
            if fval<=highfval
                highfval=fval;
                optX=x;
            end
    end
    Weight(:,k+1)=optX;
    k=k+1;
end
Wealth(1)=1;
for j=1:k-1
    new_wealth=sum(prod(1+AllReturn(StopNumber+1+Period*(j-1):...
    StopNumber+1+Period*j,:))'.*Weight(:,j)*Wealth(j));

    change=abs(Weight(:,j+1)*new_wealth-prod(1+AllReturn...
    (StopNumber+1+Period*(j-1):StopNumber+1+Period*j,:))'...
        .*Weight(:,j)*Wealth(j));
    Transaction_cost(j)=0.0004*sum(change);
    Wealth(j+1)=new_wealth-Transaction_cost(j);
    Delta(j,:)=0.0004*(change);
end
% Wealth
% Transaction_cost
% Weight
% Delta=Delta'/.0004


% Truely_Price is the value of the portfolio each day from the beginning of
% the testing data.(i.e. the first day after Day StopNumber)
Truely_Price(1)=Wealth(1);
Truely_Price(2)=sum(Wealth(1)*(1+AllReturn(StopNumber,:))'.*Weight(:,j));
temp=Truely_Price(1);
No=1;
for j=3:No_of_Day-StopNumber+1
    Truely_Price(j)=sum(temp*Weight(:,No).*prod(1+AllReturn(StopNumber+...
    (No-1)*Period:StopNumber+j-2,:),1)');
    if((j+1)/Period==floor((j+1)/Period))
        Truely_Price(j)=Wealth(No+1);
        No=No+1;
        temp=Truely_Price(j);
    end
end

Portfolio_Return(1)=Truely_Price(SR_interval)/Wealth(1)-1;
for j=1:No_of_Day-StopNumber-SR_interval+1
    Portfolio_Return(j)=Truely_Price(SR_interval+j)/Truely_Price(j)-1;
end
```

```
MU=mean(Portfolio_Return);
STD=std(Portfolio_Return);
SR=MU/STD;
```

A.3 Equallyweighted.m

```
function [SR,Truely_Price]=Equallyweighted(SR_interval,StopNumber)
AllReturn=xlsread('Fields_Mitac_MapleridgeDataSet.csv');
[m,n]=size(AllReturn);
No_of_Asset=n;
No_of_Day=m;
digits(10)

weight=ones(35,1)/35;
Truely_Price(1)=1;
for i=2:(m-StopNumber+2)
    Truely_Price(i)=sum(Truely_Price(1)*weight.*prod(1+...
    AllReturn(StopNumber:StopNumber+i-2,:),1)');
end
for j=1:No_of_Day-StopNumber-SR_interval+1
    Portfolio_Return(j)=Truely_Price(SR_interval+j)/Truely_Price(j)-1;
end
MU=mean(Portfolio_Return);
STD=std(Portfolio_Return);
SR=MU/STD;
```

A.4 one_over_sigma.m

```
function [SR,Truely_Price]=one_over_sigma(SR_interval,StopNumber)
AllReturn=xlsread('Fields_Mitac_MapleridgeDataSet.csv');
[m,n]=size(AllReturn);
No_of_Asset=n;
No_of_Day=m;
% StopNumber=1500;
Old=AllReturn(1:StopNumber-1,:);
correlation_old=corrcoef(Old);
mean_old=mean(Old);
std_old=std(Old);
for i=1:35
    x3(i)=(1./std_old(i))/sum(1./std_old);
end
weight=x3';

Truely_Price(1)=1;
for i=2:(m-StopNumber+2)
    Truely_Price(i)=sum(Truely_Price(1)*weight.*...
    prod(1+AllReturn(StopNumber:StopNumber+i-2,:),1)');
end
for j=1:No_of_Day-StopNumber-SR_interval+1
```

```
    Portfolio_Return(j)=Truely_Price(SR_interval+j)/Truely_Price(j)-1;
end
MU=mean(Portfolio_Return);
STD=std(Portfolio_Return);
SR=MU/STD;
```

A.5 Opone.m

```
function [SR,Truely_Price]=Opone(SR_interval,StopNumber)
AllReturn=xlsread('Fields_Mitac_MapleridgeDataSet.csv');
[m,n]=size(AllReturn);
No_of_Asset=n;
No_of_Day=m;
digits(10)
PP=500;
Period=60;


for i=1:No_of_Day-Period
    % The return for each strategy between the period
    Period_Return(i,:)=prod(1+AllReturn(i:i+Period-1,:))-1;
end
Old=Period_Return(StopNumber-Period+1-PP+1:StopNumber-Period+1,:);
correlation_old=corrcoef(Old);
mean_old=mean(Old);
std_old=std(Old);

H = ones([1,No_of_Asset]);
%iniValues=H*1/No_of_Asset;
LB = zeros([1,No_of_Asset]);
UB =ones([1,No_of_Asset]);
Quadraticfun = @(x)(-mean_old*x)/sqrt((std_old'.*x)'...
*correlation_old*(std_old'.*x));
opts = optimset('fmincon');
opts.LargeScale='off';
opts.Display='off';
Aeq=H;
beq=[1];
highfval=0;
for i=1:1
    iniValues=randn(1,No_of_Asset).*0.1';
    [x,fval] =fmincon(Quadraticfun,iniValues',[],[],...
    Aeq,beq,LB',UB',[],opts);
     if fval<=highfval
         highfval=fval;
         optX=x;
     end
end
```

```
weight=optX;
Truely_Price(1)=1;
for i=2:(m-StopNumber+2)
    Truely_Price(i)=sum(Truely_Price(1)*weight.*...
    prod(1+AllReturn(StopNumber:StopNumber+i-2,:),1)');
end
for j=1:No_of_Day-StopNumber-SR_interval+1
    Portfolio_Return(j)=Truely_Price(SR_interval+j)/Truely_Price(j)-1;
end
MU=mean(Portfolio_Return);
STD=std(Portfolio_Return);
SR=MU/STD;
```

A.6 Second_Trans.m

```
function [SR,Truely_Price]=Second_Trans(Period,SR_interval,StopNumber)
% Input:
% Period:  reallocate period, unit: Number of days!
% SR_interval;%Sharp ratio interval, has to be less than StopNumber
% (default 500!), unit: Number of days!
% StopNumber:Default 500, the number of days we used as historical
% data, unit: Number of days!
% Output:
% SR: sharp ratio of Period we need
%$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$


% Period=256;% reallocate period 1 yr
% SR_interval=125;%Sharp ratio interval, has to be less
% than StopNumber/2 (default 500!)
AllReturn=xlsread('Fields_Mitac_MapleridgeDataSet.csv');
[m,n]=size(AllReturn);
No_of_Asset=n;
No_of_Day=m;
% StopNumber=500;
for i=1:No_of_Day-Period
    % The return for each strategy between the period
    Period_Return(i,:)=prod(1+AllReturn(i:i+Period-1,:))-1;
end


k=1;
while(StopNumber+1+Period*k<No_of_Day)
    k=k+1;
end
Weight=ones(35,k)/35;
Wealth(1)=1;
```

```
for j=1:k-1
    new_wealth=sum(prod(1+AllReturn(StopNumber+1+Period*(j-1):...
    StopNumber+1+Period*j,:))'.*Weight(:,j)*Wealth(j));

    change=abs(Weight(:,j+1)*new_wealth-prod(1+AllReturn(...
    StopNumber+1+Period*(j-1):StopNumber+1+Period*j,:))'...
        .*Weight(:,j)*Wealth(j));
    Transaction_cost(j)=0.0004*sum(change);
    Wealth(j+1)=new_wealth-Transaction_cost(j);
    Delta(j,:)=0.0004*(change);
end
Wealth
Transaction_cost
Weight
Delta=Delta'/.0004
% Truely_Price is the value of the portfolio each day from the beginning of
% the testing data.(i.e. the first day after Day StopNumber)
Truely_Price(1)=Wealth(1);
Truely_Price(2)=sum(Wealth(1)*(1+AllReturn(StopNumber,:))'.*Weight(:,j));
temp=Truely_Price(1);
No=1;
for j=3:No_of_Day-StopNumber+1
    Truely_Price(j)=sum(temp*Weight(:,No).*prod(1+AllReturn(...
    StopNumber+(No-1)*Period:StopNumber+j-2,:),1)'));
    if((j+1)/Period==floor((j+1)/Period))
        Truely_Price(j)=Wealth(No+1);
        No=No+1;
        temp=Truely_Price(j);
    end
end

Portfolio_Return(1)=Truely_Price(SR_interval)/Wealth(1)-1;
for j=1:No_of_Day-StopNumber-SR_interval+1
    Portfolio_Return(j)=Truely_Price(SR_interval+j)/Truely_Price(j)-1;
end
MU=mean(Portfolio_Return);
STD=std(Portfolio_Return);
SR=MU/STD;
```