APS502 Project 2

Di Tong Liu

1002196394

Part 1

a.

Mean and standard deviation table:

	Expected return	standard deviation
SPY	**0.008616**	**0.03472**
GOVT	**0.002625**	**0.01109**
EEMV	**0.002052**	**0.03459**

Variance and covariance table (red = variance, black = covariance):

	SPY	GOVT	EEMV
SPY	**0.001205**	**-0.0001061**	**0.00076**
GOVT	**-0.0001061**	**0.000123**	**0.000008884**
EEMV	**0.00076**	**0.000008884**	**0.001196**

b.

Decision variables:

Let w_1 = weight of SPY fund, w_2 = weight of GOVT fund, and w_3 = weight of EEMV fund

Objective function:

Minimize $0.001205w_1^2 + 0.000123w_2^2 + 0.001196w_3^2 – 0.0002122w_1w_2 + 0.00152w_1w_3 + 0.00001777w_2w_3$

Constraints:

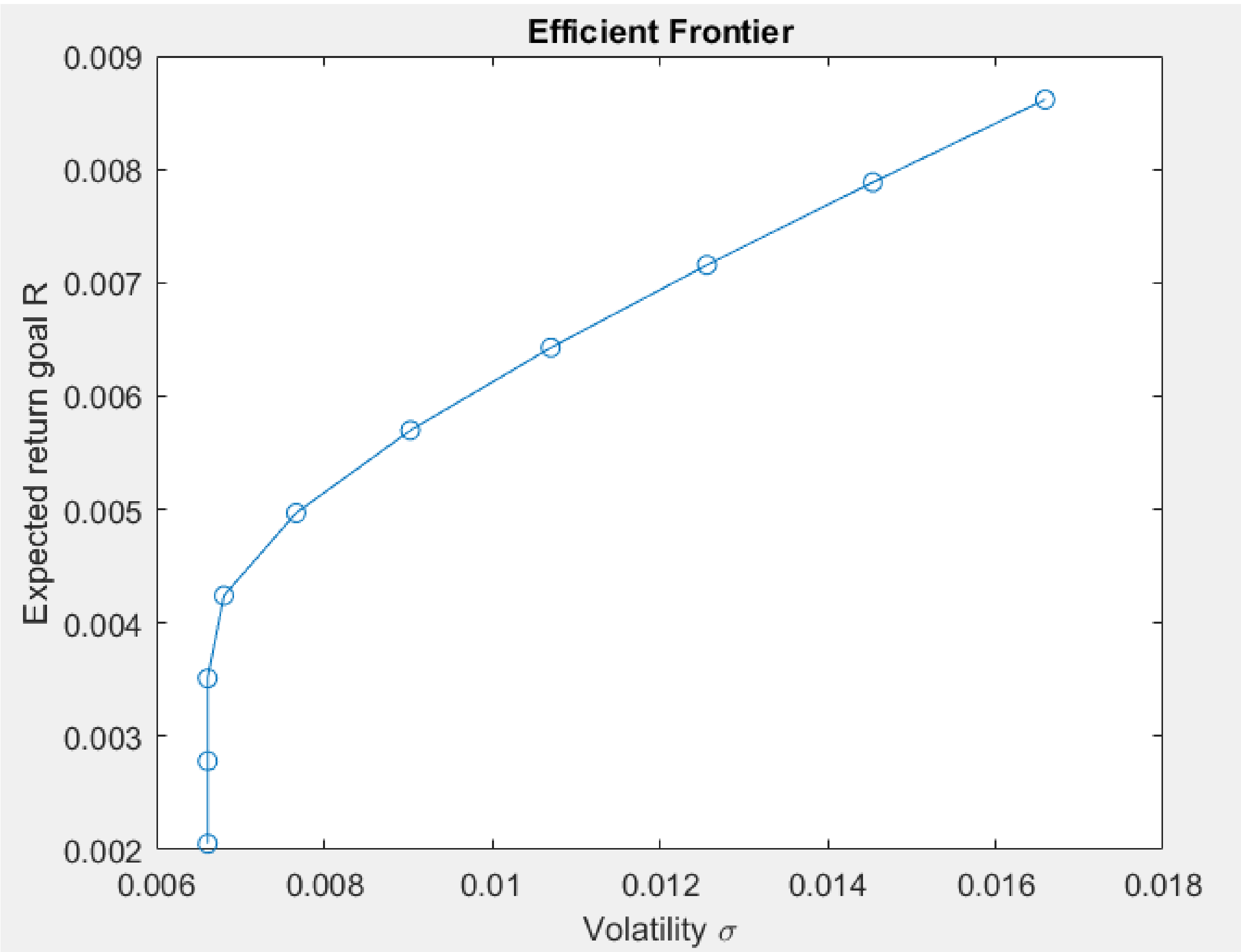
1. $0.008616w_1 + 0.002625w_2 + 0.002052w_3 >= R$, where R = expected return goal
2. $w_1 + w_2 + w_3 = 1$

I assumed that short selling is allowed.

After sweeping R through 10 different values, the following results were obtained:

R value	**Optimal weights of assets (w_1, w_2, w_3)**	**Portfolio variance (σ^2)**
0.002053	0.1785, 0.8683, -0.0468	0.00004372
0.002782	0.1785, 0.8683, -0.0468	0.00004372
0.003511	0.1785, 0.8683, -0.0468	0.00004372
0.00424	0.2589, 0.8529, -0.1117	0.00004632
0.004969	0.3719, 0.8312, -0.203	0.00005878
0.005698	0.4848, 0.8095, -0.2943	0.00008151
0.006427	0.5977, 0.7879, -0.3856	0.0001145
0.007156	0.7107, 0.7662, -0.4769	0.0001578
0.007885	0.8236, 0.7446, -0.5682	0.0002114
0.008615	0.9367, 0.7229, -0.6596	0.0002753

Plot of efficient frontier:



Matlab code:

```
format long g

R_array = [0.002053 0.002782 0.003511 0.00424 0.004969
0.005698 0.006427 0.007156 0.007885 0.008615]; % R values
used
Q = [0.001205, -0.0001061, 0.00076; -0.0001061, 0.000123,
0.000008884;0.00076, 0.000008884, 0.001196;];
c = [0 0 0]';
A = -[0.008616, 0.002625, 0.002052];
Aeq = [1 1 1];
beq = [1];
ub = [inf; inf; inf;];
lb = [-inf; -inf; -inf;];
variance_array = zeros(10,1);
weights_matrix = zeros(10,3);

for i = 1:10 % for loop to sweep through R values
   b = R_array(i)*-1;
   [x, fval] = quadprog(Q, c, A, b, Aeq, beq, lb, ub); %
each iteration of for loop calls quadprog with a new R
value
   x = round(x,10);
   fval = round(fval,10);
   variance_array(i,:) = fval;
   weights_matrix(i,:) = x;
end

variance_array % portfolio variance values
weights_matrix % weights of assets matrix (w1, w2, w3)
std_array = sqrt(variance_array); % portfolio std values

h = plot(std_array, R_array, '-o') % efficient frontier
plot
title('Efficient Frontier')
xlabel('Volatility \sigma')
ylabel('Expected return goal R')
ax = gca
ax.YAxis.Exponent = 0;
```

Matlab output:

```
variance_array =

              4.37199e-05
              4.37199e-05
              4.37199e-05
              4.63221e-05
              5.87781e-05
              8.15089e-05
             0.0001145217
             0.0001578224
             0.0002113866
             0.0002753316


weights_matrix =

             0.1785188519              0.8682607352             -0.0467795872
             0.1785180021              0.8682608982             -0.0467789003
             0.1785186461              0.8682607747             -0.0467794208
             0.2588870281              0.8528519148             -0.1117389429
             0.3718507036              0.8311936281             -0.2030443317
             0.4847857052              0.8095408391             -0.2943265442
              0.597736422               0.787885037              -0.385621459
             0.7107070815              0.7662254113             -0.4769324928
             0.8236398775              0.7445730451             -0.5682129226
             0.9367438014              0.7228878689             -0.6596316704
```

c.

Expected return table March 2020:

	Expected return
SPY	**-0.1485**
GOVT	**0.03244**
EEMV	**-0.1379**

Minimum variance portfolio from b (portfolio #1):

w_1 = 0.1785, w_2 = 0.8683, w_3 = - 0.0468, portfolio expected return = **0.008114**

Equal weighted portfolio using March 2020 returns (portfolio #2):

w_1 = 1/3, w_2 = 1/3, w_3 = 1/3, portfolio expected return = **-0.08465**

Portfolio using March 2020 returns with 60% in SPY, 30% in GOVT, and 10% in EEMV (portfolio #3):

w_1 = 0.6, w_2 = 0.3, w_3 = 0.1, portfolio expected return = **-0.09316**

Portfolios 2 and 3 both have negative returns, thus **portfolio 1 performs the best out of the three**. Portfolio 2 performs slightly better than portfolio 3 since it has a slightly higher return compared to Portfolio 3.

The reason for this is because in March 2020, GOVT is the only one of the three assets that has a positive return. Portfolio 1 has 87% of its weight put into GOVT, while portfolio 2 only has 33% put into GOVT and portfolio 3 only has 30% put into GOVT. Naturally, portfolio 1 will perform much better than portfolios 2 and 3 because of this.

Part 2

Mean and standard deviation table:

	Expected return	standard deviation
SPY	**0.008616**	**0.03472**
GOVT	**0.002625**	**0.01109**
EEMV	**0.002052**	**0.03459**
CME	**0.01691**	**0.04885**
BR	**0.01617**	**0.05451**
CBOE	**0.01187**	**0.0594**
ICE	**0.01162**	**0.04742**
ACN	**0.01307**	**0.05087**

Variance and covariance table:

	SPY	GOVT	EEMV	CME	BR	CBOE	ICE	ACN
SPY	0.0012	-0.000106	0.00076	0.000307	0.000941	0.000165	0.000607	0.0012
GOVT	-0.000106	0.00012	0.00000888	-0.0000564	0.0000402	0.0000468	-0.0000832	-0.000088
EEMV	0.00076	0.00000888	0.0012	-0.000217	0.00059	-0.000152	-0.0000175	0.000536
CME	0.000307	-0.0000564	-0.000217	0.00239	0.000912	0.00116	0.001287	0.00053
BR	0.000941	0.0000402	0.00059	0.000912	0.00297	0.0004	0.000718	0.00135
CBOE	0.000165	0.0000468	-0.000152	0.00116	0.0004	0.00353	0.00082	0.000438
ICE	0.000607	-0.0000832	-0.0000175	0.00129	0.000718	0.00082	0.00225	0.000907
ACN	0.0012	-0.000088	0.000536	0.00053	0.00135	0.000438	0.000907	0.00259

Decision variables:

Let w_1 = weight of SPY fund, w_2 = weight of GOVT fund, w_3 = weight of EEMV fund, w_4 = weight of CME fund, w_5 = weight of BR fund, w_6 = weight of CBOE fund, w_7 = weight of ICE fund, and w_8 = weight of ACN fund

Objective function:

Minimize

$0.0012w_1^2 - 0.000212w_1w_2 + 0.00152w_1w_3 + 0.000614w_1w_4 + 0.001882w_1w_5 + 0.00033w_1w_6 + 0.001214w_1w_7 + 0.0024w_1w_8$

$+ 0.0012w_2^2 + 0.00001776w_2w_3 - 0.0001128w_2w_4 + 0.0000804w_2w_5 + 0.0000936w_2w_6 - 0.0001664w_2w_7 - 0.000176w_2w_8$

$+ 0.0012w_3^2 - 0.000434w_3w_4 + 0.00118w_3w_5 - 0.000304w_3w_6 - 0.000035w_3w_7 + 0.001072w_3w_8$

$+ 0.00239w_4^2 + 0.001824w_4w_5 + 0.00232w_4w_6 + 0.002574w_4w_7 + 0.00106w_4w_8$

$+ 0.00297w_5^2 + 0.0008w_5w_6 + 0.001436w_5w_7 + 0.0027w_5w_8$

$+ 0.00353w_6^2 + 0.00164w_6w_7 + 0.000876w_6w_8$

$+ 0.00225w_7^2 + 0.001814w_7w_8$

$+ 0.00259w_8^2$

Constraints:

1. $0.008616w_1 + 0.002625w_2 + 0.002052w_3 + 0.01691w_4 + 0.01617w_5 + 0.01187w_6 + 0.01162w_7 + 0.01307w_8 >= R$, where R = expected return goal
2. $w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 + w_8 = 1$

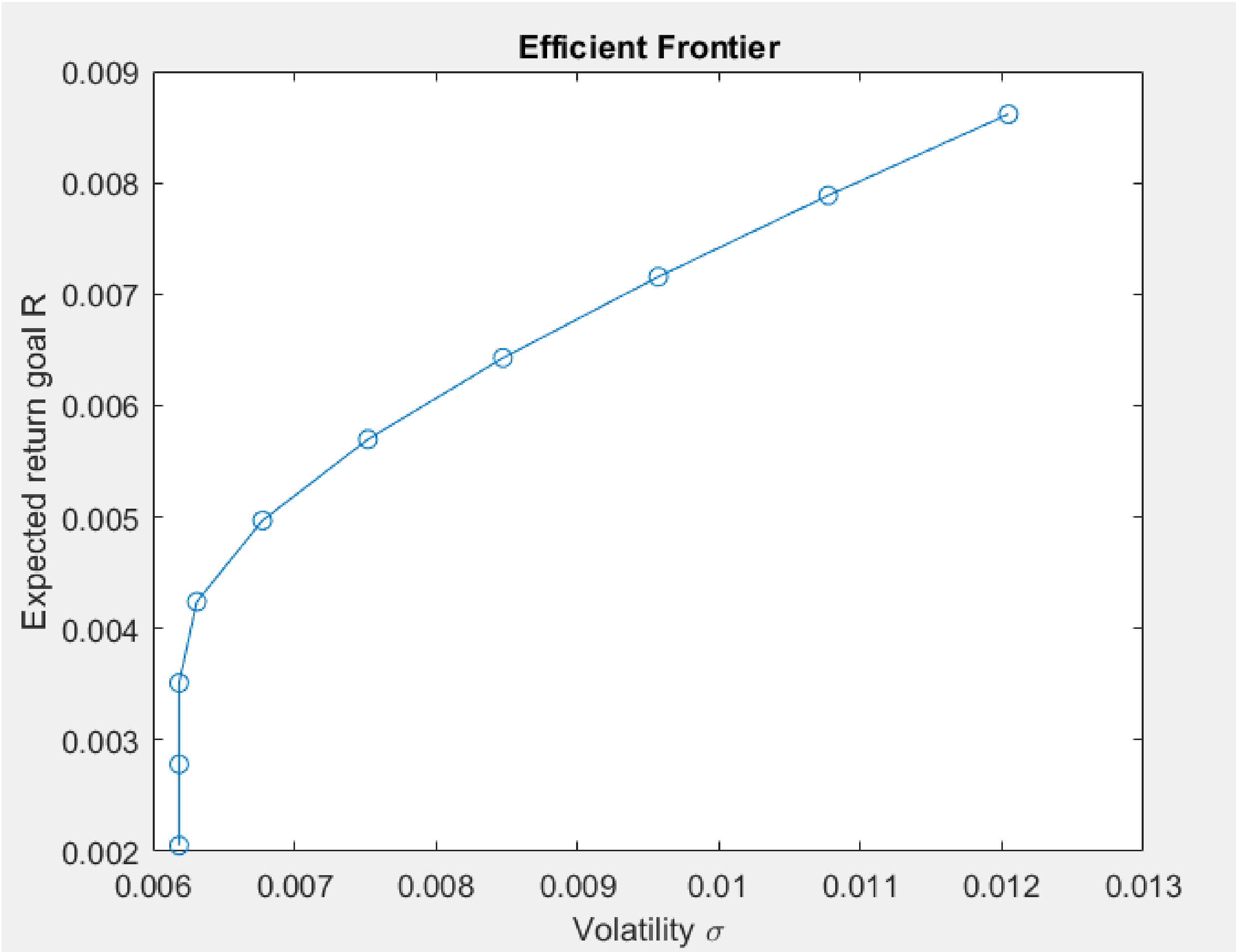
I assumed that short selling is allowed.

After sweeping R through 10 different values, the following results were obtained:

(Note: These are the same R values that were used in part 1)

R value	**w_1**	**w_2**	**w_3**	**w_4**	**w_5**	**w_6**	**w_7**	**w_8**	**σ^2**
0.002053	0.1861	0.85	-0.0242	0.04606	-0.05215	-0.01045	0.01215	-0.00752	0.00003826
0.002782	0.1861	0.85	-0.02421	0.04607	-0.05215	-0.01045	0.01215	-0.00752	0.00003826
0.003511	0.1862	0.8497	-0.02442	0.04627	-0.05199	-0.01041	0.01209	-0.00745	0.00003826
0.00424	0.1963	0.8205	-0.04381	0.06494	-0.03735	-0.00681	0.00693	-0.00067	0.00003982
0.004969	0.2086	0.7847	-0.06762	0.08786	-0.01938	-0.0024	0.00058	0.00766	0.00004589
0.005698	0.2209	0.7489	-0.09144	0.1108	-0.00141	0.00202	-0.00576	0.01598	0.00005655
0.006427	0.2333	0.7131	-0.1152	0.1337	0.01656	0.00644	-0.01211	0.02431	0.00007179
0.007156	0.2456	0.6773	-0.1391	0.1566	0.03453	0.01085	-0.01845	0.03263	0.00009162
0.007885	0.2579	0.6415	-0.1629	0.1795	0.0525	0.01527	-0.02479	0.04096	0.000116
0.008615	0.2703	0.6056	-0.1867	0.2025	0.07049	0.01969	-0.03115	0.04929	0.0001451

Plot of efficient frontier:



The table below provides the differences in portfolio variance between the efficient frontier above and the efficient frontier in part 1:

R value	**σ^2 in part 2**	**σ^2 in part 1**	**Difference (σ^2 in part 2 - σ^2 in part 1)**
0.002053	0.00003826	0.00004372	**-0.00000546**
0.002782	0.00003826	0.00004372	**-0.00000546**
0.003511	0.00003826	0.00004372	**-0.00000546**
0.00424	0.00003982	0.00004632	**-0.0000065**
0.004969	0.00004589	0.00005878	**-0.00001289**
0.005698	0.00005655	0.00008151	**-0.00002496**
0.006427	0.00007179	0.0001145	**-0.00004271**
0.007156	0.00009162	0.0001578	**-0.00006618**
0.007885	0.000116	0.0002114	**-0.0000954**
0.008615	0.0001451	0.0002753	**-0.0001302**

In comparison to the efficient frontier from part 1, the part 2 portfolio's efficient frontier provides a lower portfolio variance compared to the part 1 portfolio's efficient frontier for all 10 R values that were tested. A decrease in portfolio variance while retaining the same expected return value means that the portfolios in part 2 are less risky than the portfolios in part 1 while providing the same expected return. **Therefore, I can make the conclusion that including the 5 stocks from part 2 does in fact lead to better portfolios due to this decrease in portfolio variance.**

Matlab code:

```
format long g

R_array = [0.002053 0.002782 0.003511 0.00424 0.004969
0.005698 0.006427 0.007156 0.007885 0.008615];
Q = [0.0012, -0.000106, 0.00076, 0.000307, 0.000941,
0.000165, 0.000607, 0.0012;
    -0.000106, 0.00012, 0.00000888, -0.0000564, 0.0000402,
0.0000468, -0.0000832, -0.000088;
    0.00076, 0.00000888, 0.0012, -0.000217, 0.00059, -
0.000152, -0.0000175, 0.000536;
    0.000307, -0.0000564, -0.000217, 0.00239, 0.000912,
0.00116, 0.001287, 0.00053;
    0.000941, 0.0000402, 0.00059, 0.000912, 0.00297,
0.0004, 0.000718, 0.00135;
    0.000165, 0.0000468, -0.000152, 0.00116, 0.0004,
0.00353, 0.00082, 0.000438;
    0.000607, -0.0000832, -0.0000175, 0.00129, 0.000718,
0.00082, 0.00225, 0.000907;
    0.0012, -0.000088, 0.000536, 0.00053, 0.00135,
0.000438, 0.000907, 0.00259;
];
c = [0 0 0 0 0 0 0 0]';
A = -[0.008616, 0.002625, 0.002052, 0.01691, 0.01617,
0.01187, 0.01162, 0.01307];
Aeq = [1 1 1 1 1 1 1 1];
beq = [1];
ub = [inf; inf; inf; inf; inf; inf; inf; inf;];
lb = [-inf; -inf; -inf; -inf; -inf; -inf; -inf; -inf;];
variance_array = zeros(10,1);
weights_matrix = zeros(10,8);

for i = 1:10
   b = R_array(i)*-1;
   [x, fval] = quadprog(Q, c, A, b, Aeq, beq, lb, ub);
   x = round(x,5);
   fval = round(fval,10);
   variance_array(i,:) = fval;
   weights_matrix(i,:) = x;
end

variance_array
weights_matrix
```

```
std_array = sqrt(variance_array);

h = plot(std_array, R_array, '-o')
title('Efficient Frontier')
xlabel('Volatility \sigma')
ylabel('Expected return goal R')
ax = gca
ax.YAxis.Exponent = 0
ax.XAxis.Exponent = 0;
```

Matlab output:

```
variance_array =

             3.82628e-05
             3.82628e-05
              3.8263e-05
             3.98185e-05
             4.58893e-05
             5.65467e-05
             7.17923e-05
             9.16211e-05
             0.000116038
            0.0001450845
```

```
weights_matrix =

  Columns 1 through 4

                0.18611                     0.85                  -0.0242                  0.04606
                0.18611                  0.84999                 -0.02421                  0.04607
                0.18622                  0.84968                 -0.02442                  0.04627
                0.19626                  0.82051                 -0.04381                  0.06494
                 0.2086                  0.78471                 -0.06762                  0.08786
                0.22093                  0.74891                 -0.09144                  0.11077
                0.23326                   0.7131                 -0.11525                  0.13369
                0.24559                   0.6773                 -0.13906                   0.1566
                0.25793                   0.6415                 -0.16287                  0.17952
                0.27028                  0.60564                 -0.18672                  0.20247

  Columns 5 through 8

               -0.05215                 -0.01045                  0.01215                 -0.00752
               -0.05215                 -0.01045                  0.01215                 -0.00752
               -0.05199                 -0.01041                  0.01209                 -0.00745
               -0.03735                 -0.00681                  0.00693                 -0.00067
               -0.01938                  -0.0024                  0.00058                  0.00766
               -0.00141                  0.00202                 -0.00576                  0.01598
                0.01656                  0.00644                 -0.01211                  0.02431
                0.03453                  0.01085                 -0.01845                  0.03263
                 0.0525                  0.01527                 -0.02479                  0.04096
                0.07049                  0.01969                 -0.03115                  0.04929
```