Macroeconomics II - Problem Set 2

Gualtiero Azzalini

Question 1

Part (1)

The following part of my code imports the dataset, note that I divided the returns by 100 in order not to work with percentages:

```
% Import Dataset
data    = xlsread('data_homework_nyu.xls');
years   = data(:,1);
C_ND    = data(:,2);
P_ND    = data(:,3);
exr     = data(:,4)/100;
smb     = data(:,5)/100;
hml     = data(:,6)/100;
rf      = data(:,7)/100;
s1      = data(:,8)/100;
s2      = data(:,9)/100;
s3      = data(:,10)/100;
b1      = data(:,11)/100;
b2      = data(:,12)/100;
b3      = data(:,13)/100;
T       = length(years);
```

Then the following part of the code first computes the Market Return as $r_t = exr_t + rf_t$, i.e. as the sum between the excess return and the risk-free return, then I compute consumption growth (I use $log\,(C_t/C_{t-1})$) and inflation (I use $log\,(P_t/P_{t-1})$) . Finally, I coumpute the real returns by subtracting inflation (π) from the original returns, i.e. $real\,x_t = x_t - \pi_t$ where x_t is any return that I need.

```
%% Question 1
r = exr+rf; % Market Return
% Compute C growth and Pi
c_g = log(C_ND(2:end,1)./C_ND(1:end-1,1));
pi  = log(P_ND(2:end,1)./P_ND(1:end-1,1));
% Real returns
rf_r = rf(1:end-1) - pi;
r_r  = r(1:end-1) - pi;
```

```
s1_r = s1(1:end-1) - pi;
s2_r = s2(1:end-1) - pi;
s3_r = s3(1:end-1) - pi;
b1_r = b1(1:end-1) - pi;
b2_r = b2(1:end-1) - pi;
b3_r = b3(1:end-1) - pi;
```

Part (2)

In the following part of the code I compute the required means, standard deviations and autocorrelations:

```
% Mean, sd, autocorrelation
means = [mean(c_g),mean(rf_r),mean(r_r),mean(s1_r),mean(s2_r),...
              mean(s3_r),mean(b1_r),mean(b2_r),mean(b3_r)];
stds  = [std(c_g),std(rf_r),std(r_r),std(s1_r),std(s2_r),...
              std(s3_r),std(b1_r),std(b2_r),std(b3_r)];
autocorrs = [sacf(c_g,1),sacf(rf_r,1),sacf(r_r,1),...
                  sacf(s1_r,1),sacf(s2_r,1),sacf(s3_r,1),...
                  sacf(b1_r,1),sacf(b2_r,1),sacf(b3_r,1)];

Q1Part2_mean = table(means);
Q1Part2_std  = table(stds);
Q1Part2_autocorr = table(autocorrs);
```

Table 1 reports the values:

Table 1: Mean, St.Dev, Autocorrelation

	Mean	St.Dev	Autocorrelation (1 lag)
Consumption Growth	0.0243	0.0257	0.3283
Real Risk-Free Rate	0.0087	0.0530	0.6645
Real Market Return	0.0862	0.1948	0.0116
Real S1	0.0796	0.3452	-0.0402
Real S2	0.1251	0.3259	-0.0173
Real S3	0.1896	0.3968	0.1021
Real B1	0.0520	0.2101	-0.0325
Real B2	0.0723	0.2316	-0.0248
Real B3	0.0918	0.2744	-0.1085

Looking at Table 1 it clearly emerges the mean-variance trade-off: the higher the mean of the return, the higher its variability. This of course regards also the market return and the risk-free rate, for which we get values closed to the stylized facts we saw during class: 0.8% average return and 5% st.dev for the risk-free rate and 8.6% and 19.5% for the market return. Moreover, we get high persistence of the risk-free rates and low correlation for the market return. The mean-variance trade-off is present also in the 6 portfolios that also show low (and negative except in one case) autocorrelation.

From comparing S and B we also get the two Fama-French factors: S stocks have higher mean than B (size premium) and low book-to-market is associated with lower returns (value premium).

Part (3)

Let x_t be any return. We have already divided it by 100 when importing the data so we have the level value. The log-return is defined as $\log(1+x_t)$. The following code computes what required:

```
% Mean, sd of Log-returns
lmeans = [mean(log(1+rf_r)),mean(log(1+r_r)),mean(log(1+s1_r)),...
            mean(log(1+s2_r)),mean(log(1+s3_r)),mean(log(1+b1_r)),...
            mean(log(1+b2_r)),mean(log(1+b3_r))];
lstds  = [std(log(1+rf_r)),std(log(1+r_r)),std(log(1+s1_r)),...
            std(log(1+s2_r)),std(log(1+s3_r)),std(log(1+b1_r)),...
            std(log(1+b2_r)),std(log(1+b3_r))];

Q1Part3_mean = table(lmeans);
Q1Part3_std  = table(lstds);
```

Table 2 reports the results:

Table 2: Mean, St.Dev of Log-Returns

	Mean	St.Dev
Real Risk-Free Rate	0.0073	0.0527
Real Market Return	0.0656	0.1900
Real S1	0.0254	0.3272
Real S2	0.0767	0.2923
Real S3	0.1235	0.3177
Real B1	0.0289	0.2167
Real B2	0.0463	0.2213
Real B3	0.0568	0.2525

Part (4)-(5)

In this following part of the code we run these two series of regressions:

$$r_t^i = \alpha^i + \beta_{mkt}^i r_t^m + \varepsilon_t \tag{1}$$

$$r_t^i = \alpha^i + \beta_{cons}^i c_t + \varepsilon_t \tag{2}$$

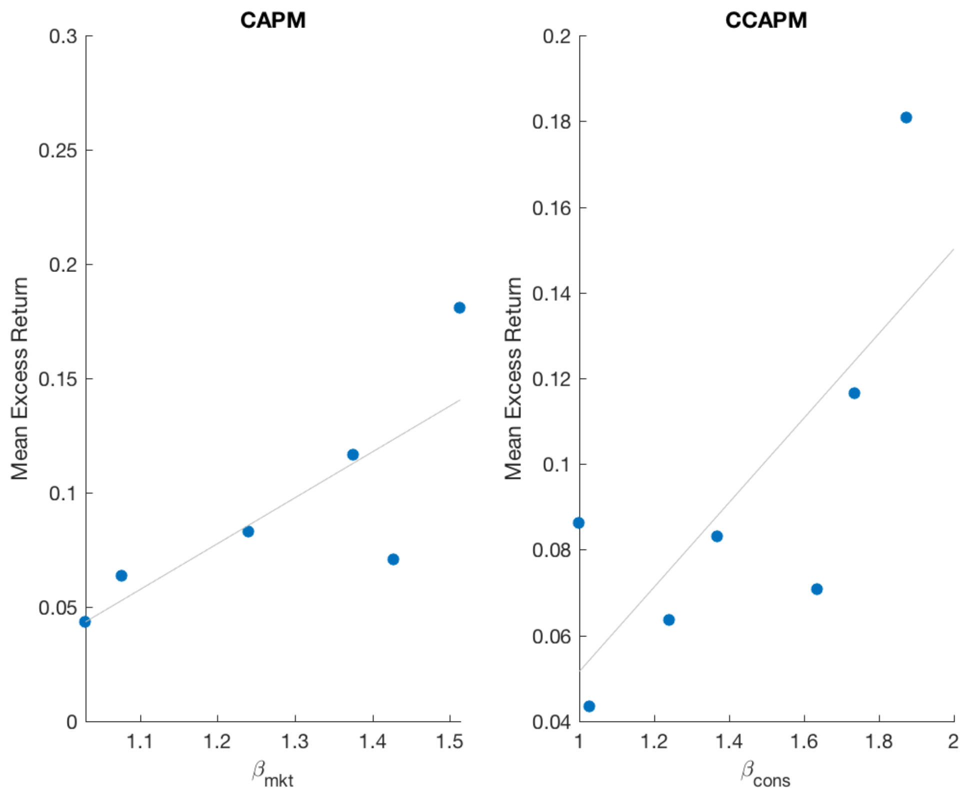
In equation (1) we have $i =$*S1,S2,S3,B1,B2,B3* while in equation (2) $i =$*S1,S2,S3,B1,B2,B3,Market Return* and c_t is consumption growth. Hence the first case correspond to CAPM while the second to CCAPM. We then plot the resulting values for β in the two cases against the Mean Excess Return. We clearly see from Figure 1 that higher $\hat{\beta}_{mkt}$ predicts higher Mean Excess Return as stated by the CAPM model. Similarly, also the chart on the right (where the βs have been normalized by β_{cons}^{mkt}, i.e. the value for the Market Return for comparability) shows that the results are in accordance with the predictions of the CCAPM model as higher $\hat{\beta}_{cons}$ predicts higher Mean Excess Return. Both theories predicts that investors should get higher expected return only if they buy riskier assets (i.e. higher β) and this is exactly what I find in the data.

```
% Computing the Betas portfolio = [s1_r s2_r s3_r b1_r b2_r b3_r];
port_mean = [mean(s1_r-rf_r),mean(s2_r-rf_r),mean(s3_r-rf_r),...
          mean(b1_r-rf_r),mean(b2_r-rf_r),mean(b3_r-rf_r)]';
X1 = [ones(length(r_r),1) r_r];
X2 = [ones(length(c_g),1) c_g];

for j=1:6
beta_market(j,:) = regress(portfolio(:,j),X1)';
beta_cons(j,:)   = regress(portfolio(:,j),X2)';
end
beta_cons(7,:) = regress(r_r,X2)';
port_mean_c = [port_mean; mean(r_r)];

figure(1)
subplot(1,2,1)
scatter(beta_market(:,2),port_mean,'filled'); title('CAPM');
ylabel('Mean Excess Return'); xlabel('\beta_{mkt}'); lsline;
axis([min(beta_market(:,2)) max(beta_market(:,2)) 0 0.3]);
subplot(1,2,2)
scatter(beta_cons(:,2)./beta_cons(7,2),port_mean_c,'filled'); title('CCAPM');
ylabel('Mean Excess Return'); xlabel('\beta_{cons}'); lsline;
saveas(figure(1),'figure1.png');
```

Figure 1: Beta and Mean Excess Return, CAPM (left) CCAPM (right)



Question 2

Part (1)

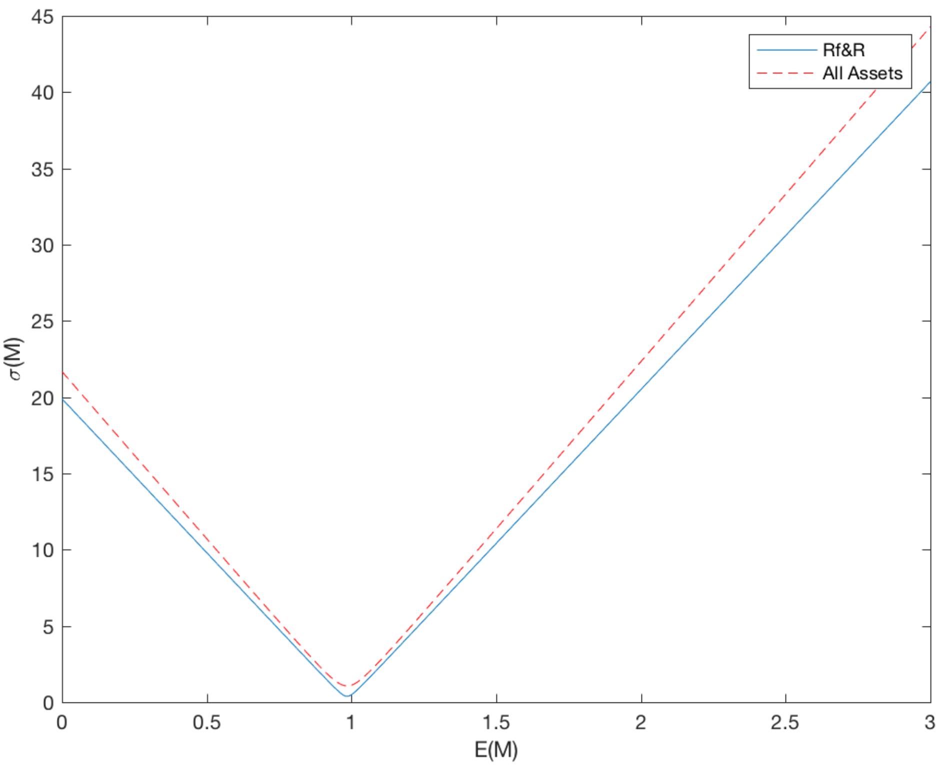
The following part of the code does the job (it just computes the formula in the text):

```
%% Question 2
% H-J Bounds
R1 = [(rf_r+1) (r_r+1)]; ER1 = mean(R1)'; VR1 = cov(R1);
R2 = [(s1_r+1) (b3_r+1) (rf_r+1) (r_r+1) (s2_r+1) (b2_r+1) (s3_r+1) (b1_r+1)];
ER2 = mean(R2)'; VR2 = cov(R2);
p=1; for j=0:.01:3
VM1(p,1) = sqrt((ones(length(ER1),1)-j*ER1)'*inv(VR1)*(ones(length(ER1),1)-j*ER1));
VM2(p,1) = sqrt((ones(length(ER2),1)-j*ER2)'*inv(VR2)*(ones(length(ER2),1)-j*ER2));
p=p+1;
end
figure(2)
plot(0:.01:3,VM1);
hold on plot(0:.01:3,VM2,'r--');
ylabel('\sigma(M)');xlabel('E(M)');
legend('Rf&R','All Assets');
```

```
saveas(figure(2),'figure2.png');
hold off
```

The results for the two portfolios are plotted in Figure 2. It is clear that the H-J bound with only two assets is lower than the other. Intuitively this happens because including more risky stocks in the portfolio increases the volatility for a given value of $E(M)$.

Figure 2: H-J Bounds for Different Portfolios (2 and 8 assets)



Part (2)

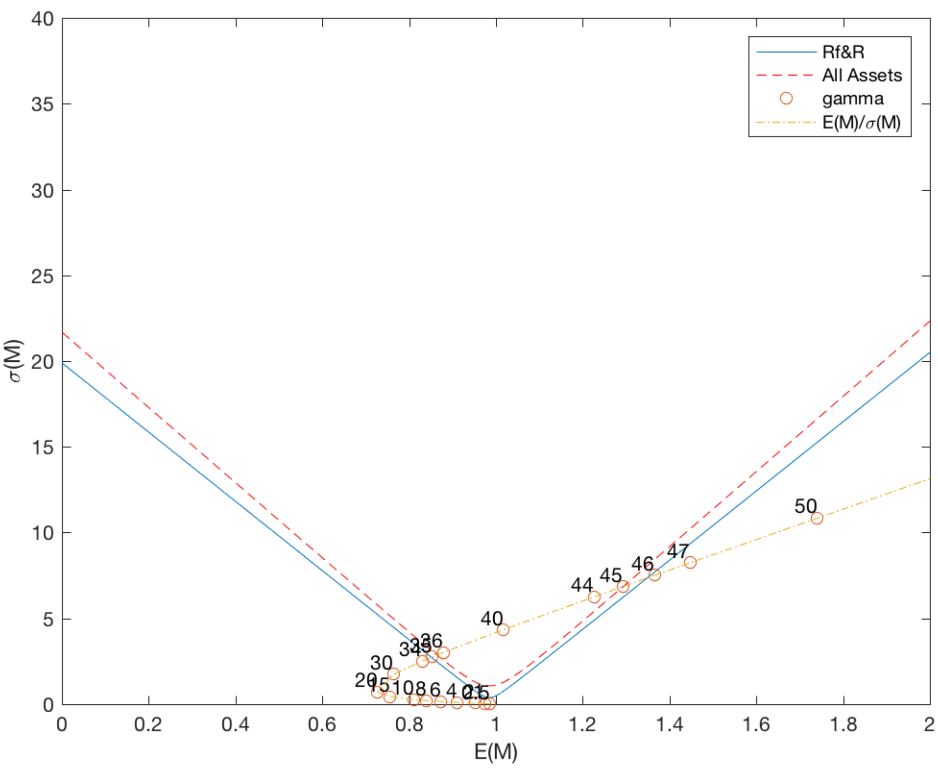
The following part of the code does the job. Basically I computed $M_{t+1} = \left(\frac{C_{t+1}}{C_t}\right)^{-\gamma}$, the implied SDF of the CRRA preferences for different values of γ, namely [0.5 1 2 4 6 8 10 15 20 30 34 35 36 40 44 45 46 47 50 60 70 90 100 150 200] - some of them are not plotted as they don't fit into the picture - and then calculated its mean and st.dev. As no value for β is given I use the same value as the one given in Question 3, i.e. $\beta = 0.998$.

```
% CRRA
beta = 0.998;
gamma = [0.5 1 2 4 6 8 10 15 20 30 34 35 36 40 44 45 46 47 50 60 70 90 100 150 200]';
for jj = 1:length(gamma)
for tt=1:T-1
M(tt,1) = beta*(C_ND(tt+1,1)/C_ND(tt,1))^(-gamma(jj,1));
end
```

```
E_SDF(jj,1) = mean(M);
DEV_SDF(jj,1) = std(M);
end
figure(3) plot(0:.01:3,VM1);
hold on plot(0:.01:3,VM2,'r--');
ylabel('\sigma(M)');xlabel('E(M)');
hold on
scatter(E_SDF,DEV_SDF);
labelpoints(E_SDF, DEV_SDF, gamma);
plot(E_SDF,DEV_SDF,':');
axis([0 2 0 40]);
legend('Rf&R','All Assets','gamma','E(M)/\sigma(M)');
saveas(figure(3),'figure3.png'); hold off
```

The results of this exercise are reported in Figure 3: the dots represent the values I obtained for different values of γ. It's visible that for values below $\gamma = 35$ the points are below the bounds, at $\gamma = 35$ we are on the bounds, while for $\gamma \in [36, 44]$ the points are above and the H-J bounds are violated. For $\gamma = 45$ the higher bound is again satisfied and from $\gamma = 47$ onwards we are again below both the bounds.

Figure 3: H-J Bounds for Different Portfolios (2 and 8 assets) and Values from SDF with different γ with CRRA preferences



Part (3)

From the given habit model, after taking first order conditions with respect to c_t we get the following SDF:

$$M_{t+1} = \beta\theta \left(\frac{c_{t+1} - \theta h_{t+1}}{c_t - \theta h_t} \right)^{-\gamma}$$

with:

$$h_t = (1-\delta)h_{t-1} + \delta c_{t-1}$$

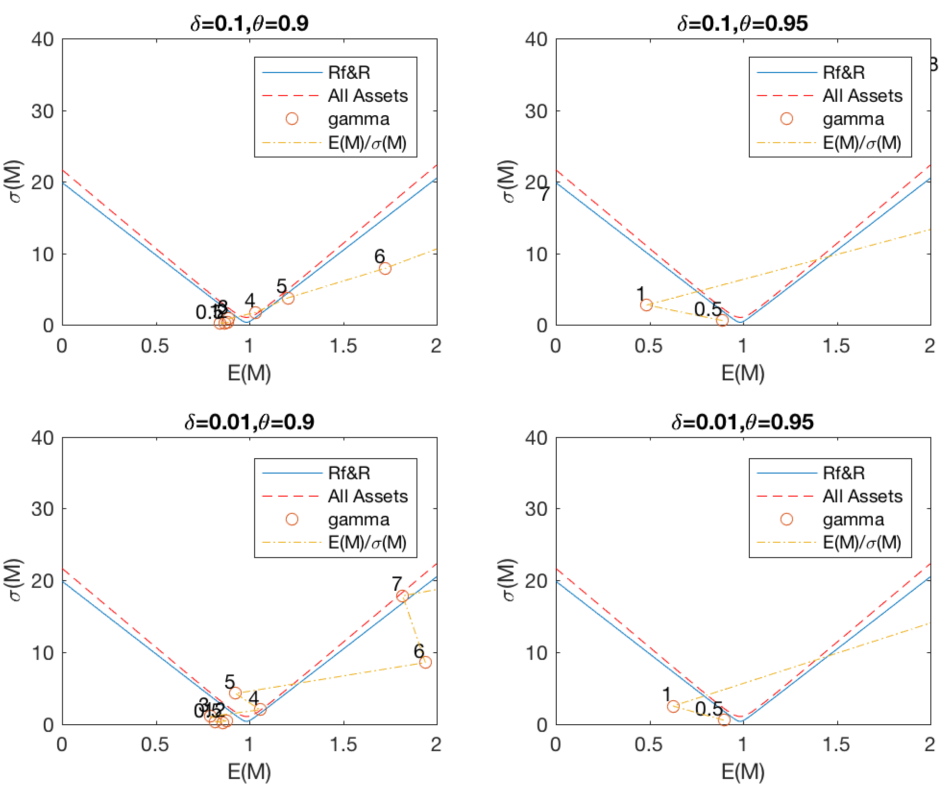
I assume that the first value of consumption in the dataset corresponds to the first value of the habit.

```
% Habit model
theta = [0.9 0.95]';
delta = [0.1 0.01]';
number = 1;
H(1,1) = C_ND(1,1);
gamma1 = [0.5 1 2 3 4 5 6]';
for kk = 1:length(delta)
for k = 1:length(theta)
for ttt=2:T
H(ttt,1) = (1-delta(kk,1))*H(ttt-1,1)+delta(kk,1)*C_ND(ttt-1,1);
end
for jj = 1:length(gamma1)
for tt=1:T-1
M_h(tt,1) = beta*theta(k,1)*((C_ND(tt+1,1)-theta(k,1)*H(tt+1,1))/...
(C_ND(tt,1)-theta(k,1)*H(tt,1)))^(-gamma1(jj,1));
end
E_SDF_h(jj,1) = mean(M_h);
DEV_SDF_h(jj,1) = std(M_h);
end
figure(4)
subplot(2,2,number)
plot(0:.01:3,VM1);
hold on
plot(0:.01:3,VM2,'r--');
ylabel('\sigma(M)');xlabel('E(M)');
hold on
scatter(E_SDF_h,DEV_SDF_h);
labelpoints(E_SDF_h, DEV_SDF_h, gamma1);
title(['\delta=',num2str(delta(kk,1)),',','\theta=',num2str(theta(k,1))]);
plot(E_SDF_h,DEV_SDF_h,':');
axis([0 2 0 40]);
legend('Rf&R','All Assets','gamma','E(M)/\sigma(M)');
```

```
number = number+1;
end
end
saveas(figure(4),'figure4.png');
hold off
```

Figure 4 plots the results for different values of θ, δ and γ, respectively, $\{0.9, 0.95\}$,$\{0.1, 0.01\}$ and $\{0.5, 1, 2, 3, 4, 5, 6\}$. For $\delta = 0.1, \theta = 0.9$ (top-left) we get a similar shape as before; however, the range of values that is above the bounds changes: indeed, from around $\gamma = 3$ to $\gamma = 5$ the points are above the bounds. These values are more credible than those obtained above. As θ increases (top-right) we get that already for some $\gamma \in (1, 2)$ the bounds are not satisfied (there are no values for $\gamma > 1$ because they fall outside the region plotted - but they are below the bounds). If δ decreases (bottom-left) the results are similar to the top-left panel even thoung we get a more erratic behavior. Finally, if both are increased we get similar results as top-right panel, which implies that the effect of θ prevails (even though the $E(M)$ is lower for points below H-J bounds on the left side).

Figure 4: H-J Bounds for Different Portfolios (2 and 8 assets) and Values from SDF with different γ, δ, θ with Habit model



Question 3

I report first all the results and then I separately comment them. Figure 5 plots the criterion functions as functions of γ, for all the cases Figure 6 does the same under optimal weighting (only in the last three cases as in the first three there is only one equation and one parameter to be estimated). Table

3 and 4 report the numbers we are asked to provide. In general, we can see the following results: the minimization works well when two assets are included (smaller CI and more credible estimate values) then it comes the 8 assets case and finally the one asset cases. In general, also, the over-id test is always rejected.

Figure 5: Criterion Functions in the six different specifications

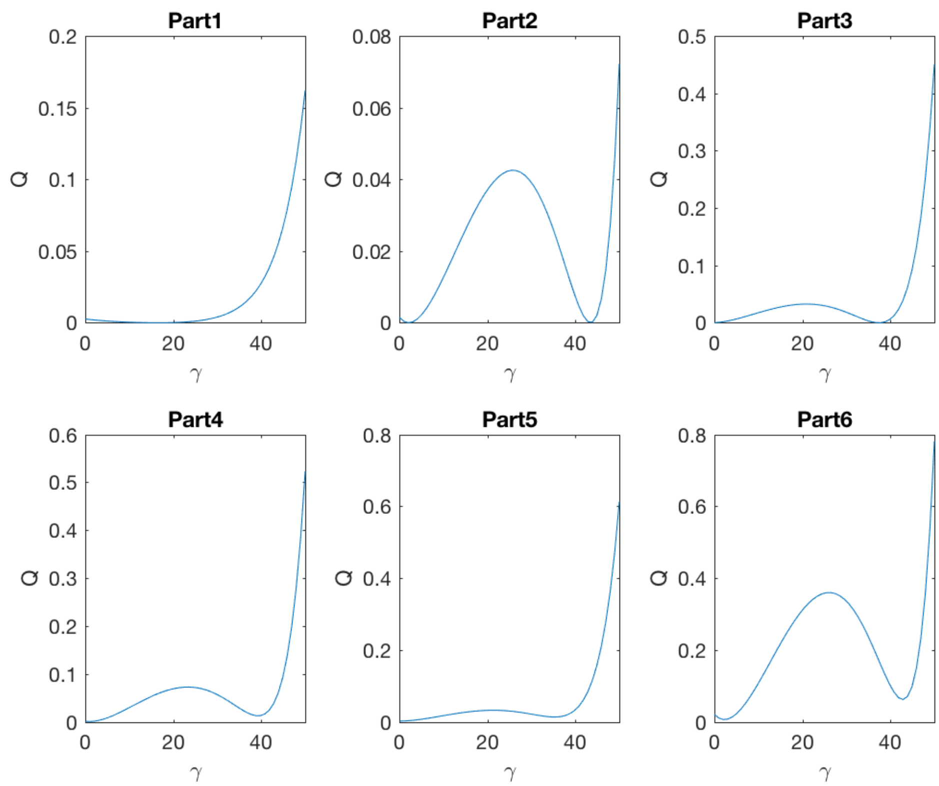


Figure 6: Criterion Functions under optimal weighting - last three specifications

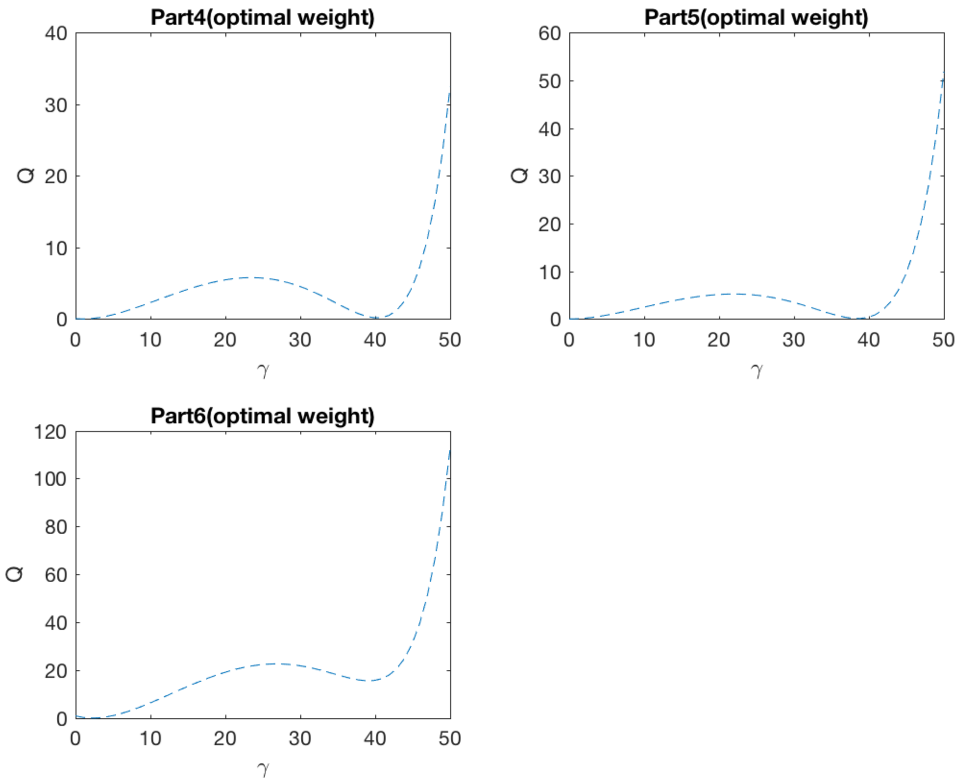


Table 3: Estimates for γ, First and Second Stages

	γ First Stage	γ Second Stage
Part 1	17.169	NA
Part 2	3.0972	NA
Part 3	0.2795	NA
Part 4	1.9196	1.9465
Part 5	0.96344	1.0312
Part 6	3.242	3.2228

Table 4: Average Errors, Confidence Intervals, J-Tests

	Average Error	95% CI	(J,P-Value,Outcome)
Part 1	-5.4031e-08	[-27.015;61.353]	NA
Part 2	-4.7136e-07	[-5.2821;11.476]	NA
Part 3	3.6946e-07	[-8.5833;9.1423]	NA
Part 4	[-0.0378,0.0297]	[1.2587;2.6344]	(2.1871,0.13917,NR)
Part 5	[-0.0175,0.0721]	[0.33116;1.7312]	(2.249,0.1337,NR)
Part 6	[-0.0646,-0.0031,-0.0149,0.0265,0.0850,-0.0350,-0.0181,-0.0011]	[2.857;3.5885]	(2.6159,0.91812,NR)

Part (1)

Criterion Function

From the panel (1,1) in Figure 5 we can see that the criterion function (plotted as a function of γ) is a parabola with a minimum at around 20.

Estimates

In all the problems I minimize $0.5g_T' W g_T$ where $g_T = \frac{1}{T}\sum_{t=1}^{T} g_t(w_t,\gamma) = \frac{1}{T}\sum_{t=1}^{T}\left[\beta\left(\frac{C_{t+1}}{C_t}\right)^{-\gamma} R_{t+1}\right] - 1$. Note that for the excess return there is no minus one at the end (because it's the difference between market and risk-free return). For the other points the minimization is done in the same fashion, changing the relevant R_{t+1}. Hence g_T is a KxT matrix where K is the number of assets included and T the time-series length. Finally, W is the identity matrix in all first stages and the optimally weighted long-run matrix (Newey-West) in the second stage $W = \left(\frac{1}{T}\sum_{t=1}^{T} g_t(w_t, \hat{\gamma}_{GMM,1}) g_t(w_t, \hat{\gamma}_{GMM,1})'\right)$. From Table 3 we see that $\hat{\gamma}_{GMM,1} = 17.169$. As there is only one equation and one parameter, there is no optimal weighting so we only have the first stage estimate.

Average Error

In these case there is only one asset so the average error, computed as explained in the text is: -5.4031e-08.

Confidence Interval

In all cases we know that under optimal weighting the estimate is asymptotically normal with asymptotic variance $(G'S^{-1}G)^{-1}$. We know that a consistent estimator for this is $(\hat{G}'\hat{S}^{-1}\hat{G})^{-1}$ where $\hat{G}$ is the matrix of derivatives of g computed at the GMM estimate with respect to the parameter and $\hat{S}^{-1} = \hat{W}$ where $\hat{W}$ is the optimally weighted matrix. Hence, computing these number we get the following 95% CI: [-27.015;61.353].

J-test

There is one parameter to be estimated and one equation so we can't perform the J-test.

Part (2)

Criterion Function

From the panel (1,2) in Figure 5 we can see that the criterion function (plotted as a function of γ) has two local minima. We shall check which of them is the global.

Estimates

From Table 3 we see that $\hat{\gamma}_{GMM,1} = 3.0972$. As there is only one equation and one parameter, there is no optimal weighting so we only have the first stage estimate. Since the function here has a lower value than in the other local minima (1.11e-13<2.24e-13) this is the global minimum.

Average Error

In these case there is only one asset so the average error, computed as explained in the text is: -4.7136e-07.

Confidence Interval

By the same procedure explained in Part (1) I get the following 95% CI: [-5.2821;11.476].

J-test

There is one parameter to be estimated and one equation so we can't perform the J-test.

Part (3)

Criterion Function

From the panel (1,3) in Figure 5 we can see that the criterion function (plotted as a function of γ) has two local minima. We shall check which of them is the global.

Estimates

From Table 3 we see that $\hat{\gamma}_{GMM,1} = 0.2795$. As there is only one equation and one parameter, there is no optimal weighting so we only have the first stage estimate. Since the function here has a lower value than in the other local minima (2.41e-14<6.83e-13) this is the global minimum.

Average Error

In these case there is only one asset so the average error, computed as explained in the text is: 3.6946e-07

Confidence Interval

By the same procedure explained in Part (1) I get the following 95% CI: [-8.5833;9.1423]

J-test

There is one parameter to be estimated and one equation so we can't perform the J-test.

Part (4)

Criterion Function

From the panel (2,1) in Figure 5 we can see that the criterion function (plotted as a function of γ) has one minima near to zero. When plotting the same function under optimal weighting in panel (1,1) in Figure 6 we see that there are two local minima so we should check which one is global.

Estimates

From Table 3 we see that $\hat{\gamma}_{GMM,1} = 1.9196$ and $\hat{\gamma}_{GMM,2} = 1.9465$. We have also checked (not reported) that in both cases these are global minima.

Average Error

There are two assets so the two average errors, respectively, for the real risk-free and the real market return are [-0.0378,0.0297].

Confidence Interval

By the same procedure explained in Part (1) - but evaluating the $\hat{G}$ matrix at the second step GMM estimate - I get the following 95% CI: [1.2587;2.6344].

J-test

The J-stat is 2.1871. Evaluating at a χ_1 (2 equations-1 parameter) the p-value is 0.13917 so we can't reject the null hypothesis that the moment restrictions hold at usual significance levels.

Part (5)

Criterion Function

From the panel (2,1) in Figure 5 we can see that the criterion function (plotted as a function of γ) has one minima near to zero. When plotting the same function under optimal weighting in panel (1,1) in Figure 6 we see that there are two local minima so we should check which one is global.

Estimates

From Table 3 we see that $\hat{\gamma}_{GMM,1} = 0.96344$ and $\hat{\gamma}_{GMM,2} = 1.0312$. We have also checked (not reported) that in both cases these are global minima.

Average Error

There are two assets so the two average errors, respectively, for the real risk-free and the real excess market return are [-0.0175,0.0721].

Confidence Interval

By the same procedure explained in Part (1) - but evaluating the $\hat{G}$ matrix at the second step GMM estimate - I get the following 95% CI: [0.33116;1.7312].

J-test

The J-stat is 2.249. Evaluating at a χ_1 (2 equations-1 parameter) the p-value is 0.1337, so we can't reject the null hypothesis that the moment restrictions hold at usual significance levels.

Part (6)

Criterion Function

From the panel (2,1) in Figure 5 we can see that the criterion function (plotted as a function of γ) has one minima near to zero. When plotting the same function under optimal weighting in panel (1,1) in Figure 6 we see that there is one minimum that is near zero.

Estimates

From Table 3 we see that $\hat{\gamma}_{GMM,1} = 3.242$ and $\hat{\gamma}_{GMM,2} = 3.2228$. We have also checked (not reported) that in both cases these are global minima.

Average Error

There are 8 assets so the 8 average errors, respectively, for the real risk-free and the real market return, s1-s3 and b1-b3 are [-0.0646,-0.0031,-0.0149,0.0265,0.0850,-0.0350,-0.0181,-0.0011]

Confidence Interval

By the same procedure explained in Part (1) - but evaluating the $\hat{G}$ matrix at the second step GMM estimate - I get the following 95% CI: [2.857;3.5885].

J-test

The J-stat is 2.6159. Evaluating at a χ_7 (8 equations-1 parameter) the p-value is 0.91812, so we can't reject the null hypothesis that the moment restrictions hold at usual significance levels.

Code

Please find it at the end of the pset.

Question 4

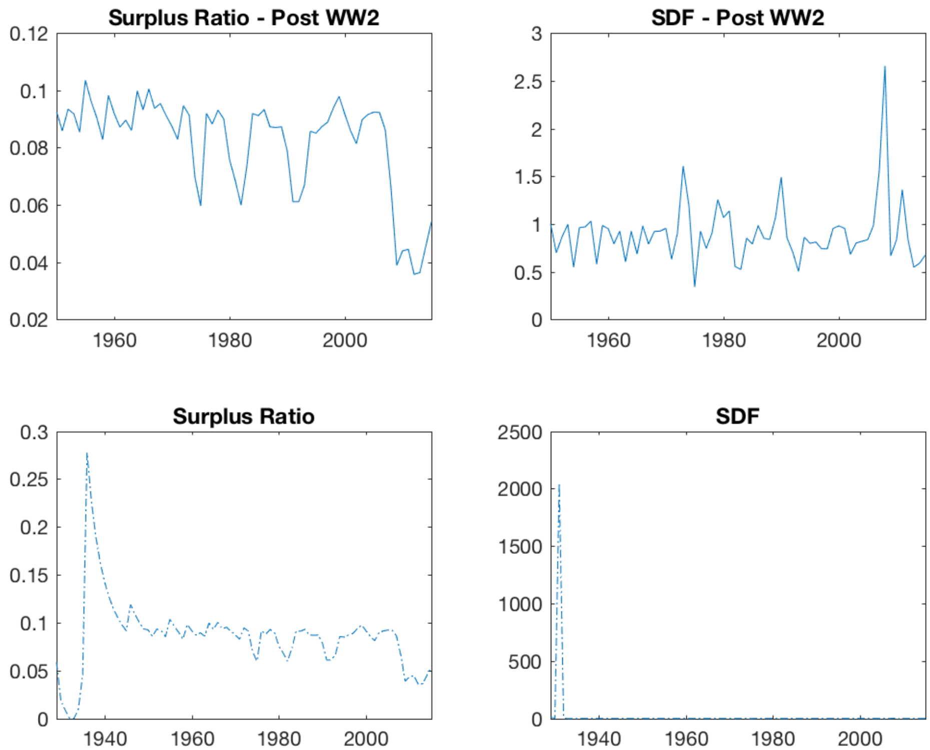
Part (a)

Following the same steps as in the slides we get that the SDF is $M_{t+1} = \beta \left(\frac{S_{t+1}}{S_t} \frac{C_{t+1}}{C_t} \right)^{-\gamma}$. Hence I use the following code:

```
%% Question 4
% Part a
sigma = 1.5/100; gamma = 2; phi    = 0.87; g       = 1.89/100; beta   = 0.89;
sbar   = sigma*sqrt(gamma/(1-phi));
lsbar = log(sbar); smax   = lsbar + (1-sbar^2)/2;
s(1,1)  = lsbar;
for j=1:length(C_ND)-1
 if s(j,1)<smax || s(j,1)==smax              l
ambda(j,1) = (sbar)^(-1)*sqrt(1-2*(s(j,1)-lsbar))-1;
 else              lambda(j,1) = 0;
end
s(j+1,1) = (1-phi)*lsbar+phi*s(j,1)+lambda(j,1)*(log(C_ND(j+1,1)/C_ND(j,1))-g);
end
S       = exp(s);
SDF_CC = beta*((S(2:end,1)./S(1:end-1,1)).*...
                          (C_ND(2:end,1)./C_ND(1:end-1,1))).^(-gamma);
```

To get the following series for S and M - note that I report the full sample and only post WW2 as the Great Recession gets huge values and does not let us see the behavior of the rest of the series:

Figure 7: S and M



Part (b)

Table 5 and Table 6 report the required summary statistics for the post war sample. We can see that while S shows autocorrelation and has low st.dev, the SDF has instead high variability and lower autocorrelation. From Table 6 we can also see that both are negatively correlated with the returns.

Table 5: Mean, St.Dev, Autocorrelation

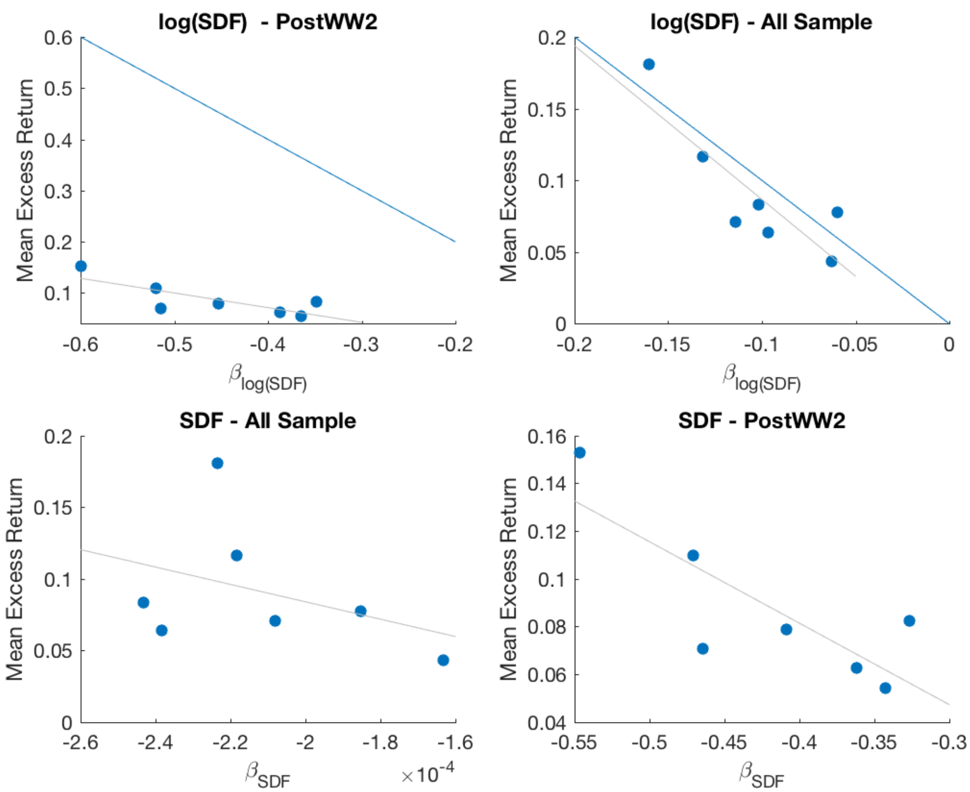
	Mean	St.Dev	Autocorrelation (1 lag)
S_t	0.081422	0.016969	0.83423
M_{t+1}	0.90165	0.32274	0.17514

Table 6: Covariances

	exr	rf_r	r_r	s1	s2	s3	b1	b2	b3
S_t	-0.00028	-3.1336e-06	-0.00027	-0.00075	-0.00071	-0.0007	-0.0007	-0.00053	-0.00046
M_{t+1}	-0.033142	-0.00084512	-0.033987	-0.048404	-0.049057	-0.056889	-0.03566	-0.037683	-0.042555

From Figure 8 it emerges the negative correlation between the betas we found and the assets, and we can also see that except for the all sample case with log(SDF) we are far away from the 45 degrees line, hence except for this case the SDF does not seem to price the assets correctly.

Figure 8: S and M



Code

```
% Part b
summary=table(mean(SDF_CC),mean(S),std(SDF_CC),std(S),sacf(SDF_CC,1),sacf(S,1))
close all
C1=cov(S,exr);C1SDF=cov(SDF_CC,exr(1:end-1));
C2=cov(S(1:end-1),rf_r);C2SDF=cov(SDF_CC,rf_r);
C3=cov(S(1:end-1),r_r);C3SDF=cov(SDF_CC,r_r);
C4=cov(S(1:end-1),s1_r);C4SDF=cov(SDF_CC,s1_r);
C5=cov(S(1:end-1),s2_r);C5SDF=cov(SDF_CC,s2_r);
C6=cov(S(1:end-1),s3_r);C6SDF=cov(SDF_CC,s3_r);
C7=cov(S(1:end-1),b1_r);C7SDF=cov(SDF_CC,b1_r);
C8=cov(S(1:end-1),b2_r);C8SDF=cov(SDF_CC,b2_r);
C9=cov(S(1:end-1),b3_r);C9SDF=cov(SDF_CC,b3_r);
covS=table(C1(1,2),C2(1,2),C3(1,2),C4(1,2),..
+C5(1,2),C6(1,2),C7(1,2),C8(1,2),C9(1,2))
covSDF=table(C1SDF(1,2),C2SDF(1,2),C3SDF(1,2),C4SDF(1,2),...
C5SDF(1,2),C6SDF(1,2),C7SDF(1,2),C8SDF(1,2),C9SDF(1,2))
% Computing the Betas
%All sample
portfolio_s = [ r_r s1_r s2_r s3_r b1_r b2_r b3_r];
```

```
port_mean_s = [ mean(r_r-rf_r) mean(s1_r-rf_r),...
mean(s2_r-rf_r),mean(s3_r-rf_r),...
mean(b1_r-rf_r),mean(b2_r-rf_r),mean(b3_r-rf_r)]';
Y1 = [ones(length(SDF_CC),1) SDF_CC];
Y2 = [ones(length(SDF_CC),1) log(SDF_CC)];
for j=1:7
beta_s(j,:) = regress(portfolio_s(:,j),Y1)';
beta_sdf(j,:)    = regress(portfolio_s(:,j),Y2)'; end

%Post WW2
portfolio_s = [ r_r(22:end) s1_r(22:end) s2_r(22:end)...
s3_r(22:end) b1_r(22:end) b2_r(22:end) b3_r(22:end)];
port_mean_s = [mean(r_r(22:end)-rf_r(22:end))...
mean(s1_r(22:end)-rf_r(22:end)),mean(s2_r(22:end)-rf_r(22:end)),...
mean(s3_r(22:end)-rf_r(22:end)),mean(b1_r(22:end)-rf_r(22:end)),...
mean(b2_r(22:end)-rf_r(22:end)),mean(b3_r(22:end)-rf_r(22:end))]';
Y1 = [ones(length(SDF_CC(22:end)),1) SDF_CC(22:end)];
Y2 = [ones(length(SDF_CC(22:end)),1) log(SDF_CC(22:end))];
for j=1:7
beta_s(j,:) = regress(portfolio_s(:,j),Y1)';
beta_sdf(j,:)    = regress(portfolio_s(:,j),Y2)';
end
```

Part (c)

I run the following set of equations: $\log(R_{t+1,t+j}) = \alpha^i + \beta^i \log(S_t)$ where $R_{t+1,t+j}$ is the cumulative real market return between periods $t+1$ and $t+j$ for j up to 7, in the post WWII sample only.

```
% Part c
post=22;
er1 = log((1+r_r(post+1:end)));
er2 = er1(2:end,1)+er1(1:end-1,1);
er3 = er1(3:end,1)+er2(1:end-1,1);
er4 = er1(4:end,1)+er3(1:end-1,1);
er5 = er1(5:end,1)+er4(1:end-1,1);
er6 = er1(6:end,1)+er5(1:end-1,1);
er7 = er1(7:end,1)+er6(1:end-1,1);
for k=1:7
X  = [ones(length(S(22:end-k-1,1)),1) log(S(22:end-k-1,1))];
y  = eval(['er',num2str(k)]);
[b,bint,r,rint,stats]=regress(y,X);
eval(['b_',num2str(k),'=b(2,1)',';']);
eval(['bint_',num2str(k),'=bint',';']);
eval(['R2_',num2str(k),'=stats(1,1)',';']);
end
```

```
betas=table(b_1,b_2,b_3,b_4,b_5,b_6,b_7)
R2=table(R2_1,R2_2,R2_3,R2_4,R2_5,R2_6,R2_7)
```

And I obtain the following values:

Table 7: β and R^2 for different horizons

HORIZON	β	R^2
1	-0.12314	0.03087
2	-0.21895	0.044386
3	-0.35201	0.072717
4	-0.55957	0.11052
5	-0.7801	0.1369
6	-0.97575	0.14539
7	-1.4871	0.18527

The betas are negative and this match the negative covariances we found before and increase with the horizon. Moreover, predictive power increases with the horizon. Negative betas imply lower returns when surplus ratio is high.

Part (d)

According to the model the following equation holds:

$$\mathbb{E}_t\left[\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}\frac{P_{t+1}+D_{t+1}}{P_t}\right]=\frac{P_t}{D_t}$$

$$\Rightarrow \mathbb{E}_t\left[\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}R_{t+1}\right]=\frac{P_t}{D_t}$$

Taking unconditional expectations:

$$\mathbb{E}\left[\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}R_{t+1}-\frac{P_t}{D_t}\right]=0$$

by using the fact that this equation should hold in the model for every return (i.e the market and the Fama-French in this case), we can check if the Campbell-Cochrane SDF is able to match the P/D ratio by looking if this equation holds in the data (we have series for S - generated by the model, for C,R and P/D so we just have to plug in the values in the data and see if this holds).

Part(e)

I firstly run the code that was provided to us to generate the artificial series I need in the following parts.

Part (i)

Using a similar code as before, I compute the required statistics. As Table 8 shows, the model replicates pretty well the mean, st.dev and autocorrelation in the data - except the autocorrelation for the SDF that turns out to be negative in the model - at least in the post WW2 sample. However, it fails to return the negative covariances we find in the data (even though they are small).

Table 8: Mean, St.Dev, Autocorrelation

	Mean		St.Dev		Autocorrelation (1 lag)		Covariance with Exret	
	Data	*Model*	*Data*	*Model*	*Data*	*Model*	*Data*	*Model*
S_t	0.081422	1.066	0.016969	0.024995	0.83423	0.85565	-0.00028	0.00064494
M_{t+1}	0.90165	0.85833	0.32274	0.04543	0.17514	-0.036774	-0.033142	0.00016373

Part (ii)

Using the same code as before, I report the results in Table 9. The returns are much more predictable than in the true data as the R^2 is always higher. We still get negative betas but their size is way bigger than before.

Table 9: β and R^2 for different horizons

Horizon	β		R^2	
	Data	*Model*	*Data*	*Model*
1	-0.12314	-1.9689	0.03087	0.11303
2	-0.21895	-3.6577	0.044386	0.20749
3	-0.35201	-5.1189	0.072717	0.28671
4	-0.55957	-6.4342	0.11052	0.35489
5	-0.7801	-7.5897	0.1369	0.41169
6	-0.97575	-8.5719	0.14539	0.45782
7	-1.4871	-9.4468	0.18527	0.49595

Part (iii)

Figure 3 in the Cochrane-Campbell paper shows that it does not matter, because they get very similar series for P/C and P/D even when correlationis low.

Part (f)

According to the model we used (and to the parameters that were given in part (a)), the set of parameters that has to be estimated is $(\sigma, \gamma, \phi, g, \beta)$. In order to estimate them, we can adopt a GMM procedure. In the model we have the following Euler Equation (I write unconditional expectation because I can apply the law of iterated expectations):

$$\mathbb{E}\left[\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}\frac{P_{t+1}+D_{t+1}}{P_t}-\frac{P_t}{D_t}\right]=0$$

$$\Rightarrow \mathbb{E}\left[\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}R_{t+1}-\frac{P_t}{D_t}\right]=0$$

Then using a set of instruments K z_t (variables known at t), we can apply GMM on:

$$\mathbb{E}\left[\left(\beta\left(\frac{S_{t+1}}{S_t}\right)^{-\gamma}\left(\frac{C_{t+1}}{C_t}\right)^{1-\gamma}R_{t+1}-\frac{P_t}{D_t}\right)\otimes z_t\right]=0$$

This last passage means that we have K equations that we can use to apply GMM and estimate the parameters in a similar fashion as we did in Question 3. Alternatively, for g, ϕ, σ we could also match the moments in real data (even though this is a calibration rather than estimation approach).