Table of Contents

Introduction

The aim of this livescript is to provide an example of how to estimate a SDF from a predictive model. The script is relatively short given that the lectures has not touched time-series predictabiltity yet. In the live-script, we will focus on the classical predictive model boing back to Fama (1984). You can easily replace the predictive model later on with both linear models and ML models that we will cover later.

The script follows the procedure of Chernov, Dahlquist and Lochstoer (2022), and we will blindly incorporate their estimate for covariance matrix and the most simply model that they incorporate. You should note that the model we focus on, is not paticular good, so you really have room for inprovement.

We will focus on the period from 2000 and forward due to simplication with the transition from DM to Euro.

Enjoy the ride!

Load currency data

In this section, we load the currency data. The data is from Thomson Reuters, which is the standard source for currency data. The MAT file, "fx_data" contains risk premia and log forward discounts of a large cross-section of currencies. The risk premia is defined as

$$RX_{t,t+1,i} = \frac{S_{t+1,i} - F_{t,t+1,i}}{S_{t,i}} = \frac{S_{t+1,i} - S_{t,i}}{S_{t,i}} + \frac{S_{t,i} - F_{t,t+1,i}}{S_{t,i}},$$

where F is the forward rate and S is the spot rate. The excess return is, hence, given as the spot rate return plus the forward discount, which under the covered interest rate parity is equal to interest rate difference.

```
% Housecleaning
clear
clc
% load data
load('fx_data');

% Defining cross-section as G11 countries
fx_use=[3, 8, 12, 26,31,32,44,45,49,50];
 RP=currencyPremia(end-251:end, fx_use);
fd=logForwardDiscount(end-251:end, fx_use);
```

Estimate covariance matrix

This section estimates the condtional covariance matrix. In the first step, we estimate the covariance matrix from daily spot changes using the Shrinkage method of Ledoit and Wolf (2020). This is done in the function analytical_shrinkage.m, directly stolen from the appendix of their paper. The output is given as Σ_t

Afterwards, Chernov et. al (2022) apply a simple exponential weighted average:

$$\widetilde{V}_t = (1-\lambda)\Sigma_t + \lambda V_{t-1}$$

where λ is set to 0.94 (riskmetrics). Now we have the condtional covariance matrix of spot changes, but we are interested in the covariance matrix of excess returns. To estimate that we pre- and post-multiply $\widetilde{V}_t$ with a diagonal matrix with $\frac{S_{t,i}}{F_{t,t+1,i}}$ along the diagonal.

```
for i=50:size(currencyPremia,1)
    idx=find(dailySpotChanges(:,1)==currencyPremia(i,1) &dailySpotChanges(:,2)==curren
    data_used=dailySpotChanges(idx(1):idx(end),fx_use+1)./100;
    if sum(isnan(data_used(:, end))>0)
        data_used(:,end)=dailySpotChanges(idx(1):idx(end),17)./100;% Replace Euro with
        data_used(isnan(data_used))=0;
    end


       sigmatilde=analytical_shrinkage(data_used);

    if i==50
        omega(i,:,:)=sigmatilde;
    else
        omega(i,:,:)=(0.94*squeeze(omega(i-1,:,:))+0.06*sigmatilde);
        omega_til(i,:,:)=diag(1./(-(forwardDiscount(i,fx_use)./100-1)))*(squeeze(omega



    end
end
omega_til=omega_til(end-251:end,:,:);
```

Construct forecasts

In this section, we construct forecasts for the predictive model, and utilize the forecasts for estimating UMVE weights. As predictive model, we consider;

$$RX_{t,t+1,i} = \alpha_i + \beta_i(s_{t,i} - f_{t,t+1,i}) + \varepsilon_{t+1}$$

where $s_{t,i} - f_{t,t+1,i}$ is the log forward discount. As alternative we consider the historical average to examine the effect of the predictive model. Our object is to construct the maximum sharpe ratio portfolio which is our only evaluation measure for now.

```
for i=36:size(RP,1)-1
    for j=1:size(RP,2)
    res=nwRegress(RP(2:i,j)./100, [fd(1:i-1,j)],1,6);
    bv(:,j)=res.bv;
    end

   mu(i,:)=sum([ones(1,size(fd,2)); fd(i,:)].*bv(:,:),1);
   weight(i,:)=1/(1+mu(i,:)/squeeze(omega_til(i,:,:))*mu(i,:)')*inv(squeeze(omega_til
   RP_stra(i+1,1)=RP(i+1,:)*weight(i,:)';


    res=nwRegress(RP(2:i,:)./100, ones(size(RP(2:i,1),1),1), 0,6);
     mu2(i,:)=res.bv;

   weight2(i,:)=1/(1+mu2(i,:)/squeeze(omega_til(i,:,:))*mu2(i,:)')*inv(squeeze(omega_t
   RP_stra_HA(i+1,1)=RP(i+1,:)*weight2(i,:)';


end

mean(RP_stra(37:end))./std(RP_stra(37:end))*sqrt(12)
```

```
ans = 0.7284
```

```
mean(RP_stra_HA(37:end))./std(RP_stra_HA(37:end))*sqrt(12)
```

```
ans = -0.0176
```

So the predictive model delivers a substantial higher sharpe ratio.