

Today's lecture we will see how to create MATLAB scripts, import data, create dates and times, download data from datafeed and plot time series.

Working with scripts

Import the data, check if you set the right path. The time series [TABSNNCB] and [TABSNNB] are retrieved from Federal Reserve Economic Data (FRED).

```
% Load the Data
load Totasset
aggregate = US_Business_TotalAssets;
corpBusiness = US_corpBusiness_TotalAssets;
noncorpBusiness = US_NoncorpBusiness_TotalAssets;
%Lets compute the logarithmic first diff, mean and standard deviation
aggreggrowthrate = diff(log(aggregate));
mu = mean(aggregate);
sigma = std(aggregate);

corpassetgrowthrate = diff(log(corpBusiness));
mu2 = mean(corpassetgrowthrate);
sigma2 = std(corpassetgrowthrate);

noncorpassetgrowthrate = diff(log(noncorpBusiness));
mu3 = mean(noncorpassetgrowthrate);
sigma3 = std(noncorpassetgrowthrate);
```

Comments and sections

We use %% to divide the script into sections, we use % to add comments

```
%% Section
% Comments
```

Now use Publish and see what happens.

Click Publish, then Edit Publishing Options. In the Edit Configurations window, specify the output format(pdf, LaTeX, html)

Plot the data

```
figure
plot(aggreggrowthrate)

figure
plot(corpassetgrowthrate)

figure
plot(noncorpBusiness)
```

Datetime function

%Create a 3-by-1 column vector named d of type datetime as shown below.

```
% 01-Feb-1999
```

```
% 01-Feb-2000
```

```
% 01-Feb-2001
```

```
d= datetime([1999;2000;2011],2,1)
```

```
d =
```

```
    01-Feb-1999
```

```
    01-Feb-2000
```

```
    01-Feb-2011
```

```
% adjust date format
```

```
% Create a datetime d representing the date January 11th 1989 using the format yyyy-MMM-dd.
```

```
d = datetime(1989, 1, 11, 'Format', 'yyyy-MMM-dd')
```

```
d =
```

```
1989-Jan-11
```

```
% Creat Duration variables
```

```
% The three inputs represent, in order, the number of hours, minutes, and seconds.
```

```
t = duration(12,0,2)
```

```
t =
```

```
12:00:02
```

```
% Create a duration t representing 1 hour, 23 minutes and 16 seconds.
```

```
t = duration(1, 23, 16)
```

```
t =
```

```
01:23:16
```

```
% Creat Calendar Duration variables
```

```
% Create a calendar duration t representing 6 calendar months.
```

```
t = calendarDuration(0, 6, 0)
```

```
t =
```

```
6mo
```

Use datetime to plot time series

```
% time space
startdate = datenum('Q1-2000','QQ-yyyy');
enddate = datenum('Q2-2018','QQ-yyyy');
dt = linspace(startdate,enddate,74);
% convert
dtDates = datetime(dt,'ConvertFrom','datenum');
% plot
figure('color','w')
plot(dtDates,US_corpBusiness_TotalAssets,'Color','r');hold on
plot(dtDates,US_NoncorpBusiness_TotalAssets,'Color','b');hold on
title('US Businesses Total Assets','FontSize',10);
legend({'Corp Businesses','NonCorp Businesses'],'FontSize',10,'Location','best')
hold off
```

Datafeed Basics

```
%
url = 'https://fred.stlouisfed.org/';
c = fred(url);

% The data series required : Real Gross Domestic Product (GDPC1)
% the start and end dates (specified as datetime variables).
startdate = datenum('Q1-2004','QQ-yyyy');
enddate = datenum('Q2-2018','QQ-yyyy');
d = fetch(c,'GDPC1',startdate,enddate)

% time space
dt = linspace(startdate,enddate,58);
% convert
dtDates = datetime(dt,'ConvertFrom','datenum');
% plot real GDP
figure('color','w')
plot(dtDates,d.Data(:,end),'Color','r');hold on % you can also extract the data with d.Data(:,2)
title('US Real GDP','FontSize',10);
hold off
```

Merge two datasets

```
x1Data =[736390      16865;
         736391      16899;
         736392      16944;
         736393      17007;
         736396      17074]

z1Data =[ 736391      16747;
         736392      16960;
         736393      17015;
         736396      16911;
         736397      16783;
```

736398 16642]

```
% CONVERT
x1Dates = datetime(x1Data(:, 1), ...
                  'ConvertFrom', 'datetime');

z1Dates = datetime(z1Data(:, 1), ...
                  'ConvertFrom', 'datetime');

% USE intersect function
[commonDates, idxx1, idxz1] = intersect(x1Dates, z1Dates);

% extract
commonx1 = x1Data(idxx1, 2);
commonz1 = z1Data(idxz1, 2);
% check the size
size(commonx1)
size(commonz1)
% Merge
commonData = [commonx1, commonz1];
```