

MFE MATLAB Companion Course
Michaelmas and Hilary 2019–2020

Kevin Sheppard
www.kevinsheppard.com

Tuesday 10th September, 2019

©2019 Kevin Sheppard

Contents

Introduction	5
Installing MATLAB	5
Add the MFE Toolbox to the Path	5
Getting Started	6
Importing Data into MATLAB	8
Functions	9
Accessing Elements in Matrices and Tables	11
Program Flow	12
Logical Operators	13
Graphics	14
Data and Simulation	16
Data Set Construction	16
Simulation	16
Expectations	17
Estimation and Inference	18
Method of Moments	18
Maximum Likelihood	18
Bias and Verification of Standard Errors	19
Linear Regression	20
Basic Linear Regression	20
Rolling and Recursive Regressions	21
Model Selection and Cross-Validation	21
Time-series Modeling	22
ARMA Estimation	22
ARMA Model Selection	22
ARMA Residual Diagnostics	22
ARMA Forecasting	23
Unit Root Testing	23
Volatility Modeling	24
ARCH Model Estimation	24
ARCH Model Selection	24
ARCH Model Forecasting	24

Value-at-Risk Modeling	25
Value-at-Risk using Historical Simulation	25
Value-at-Risk using Filtered Historical Simulation	25
Value-at-Risk Forecast Evaluation	25
Vector Autoregressions	26
Vector Autoregression (VAR) Estimation	26
VAR Model Order Selection	26
Granger Causality Testing	26
Impulse Responses	27
Cointegration Testing using Engle-Granger	27

Introduction

Installing MATLAB

MATLAB is available to install on your local PC or Mac. It can be downloaded from OUCS at

<http://help.it.ox.ac.uk/sls/matlab#div.2>

You will need your single sign-on name in order to download MATLAB. You will also require the MFE toolbox during the course, which is available at available at

http://www.kevinsheppard.com/MFE_Toolbox

The toolbox can be installed using the function `addToPath` which you will find after unzipping the files on your hard drive. Finally, in order to complete the tutorial on your own PC, you will need the zipped data files available at

https://www.kevinsheppard.com/MFE_MATLAB

Add the MFE Toolbox to the Path

Extract the contents of the MFE toolbox somewhere on your computer and then use the GUI too located under Home, Environment, Set Path to add these directories to your MATLAB path.

To verify that you were successful, **close** and **reopen** MATLAB, then run the following command

```
which acf -all
```

The output should be

```
PATH\WHERE\YOU\PUT\THE\TOOLBOX\timeseries\acf.m
```

If you see `'acf'` not found, something has gone wrong.¹

If you get an error about not being able to save the path, enter `edit startup.m` in the command window, and then type the following into the editor window

```
pd = pwd
cd PATH\WHERE\YOU\PUT\THE\TOOLBOX\
addToPath -silent
cd(pd)
```

This will add the MFE toolbox to your path each time you open MATLAB.

¹PATH\WHERE\YOU\PUT\THE\TOOLBOX\ is the location where you extracted the files. For example, on Windows, it may be something like C:\users\username\document\MFEToolbox\ or on OSX it might be /Users/username/MFEToolbox/ .

Getting Started

This lesson covers:

- Launching MATLAB
- Launching the editor
- Creating a startup file

Launching MATLAB

To launch MATLAB, run MATLAB R2016b.² When MATLAB opens, a window similar to figure 1.3 should be present, although the contents of the panes may vary.

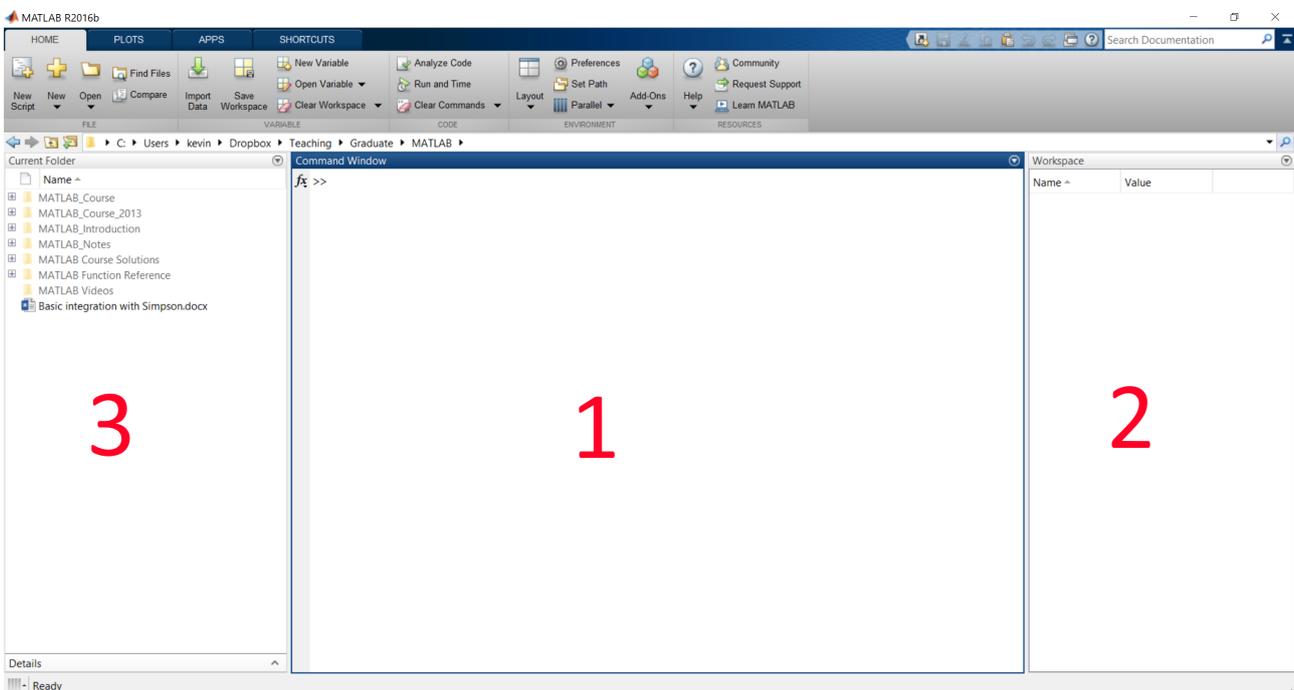


Figure 1.1: Basic MATLAB Window. The standard setup has three panes. 1: The command window, 2: Workspace and 3: Current Directory

Problem 1 (Launching MATLAB). Open MATLAB on your computer.

Launch the Editor

Once MATLAB is up and running, launch the editor. There are two methods to accomplish this task

- Enter `edit` in the command window
- Use the menu via Home>New>Script.

The editor should appear similar to figure 1.2.

²The version may be different depending on what OUCS is distributing.

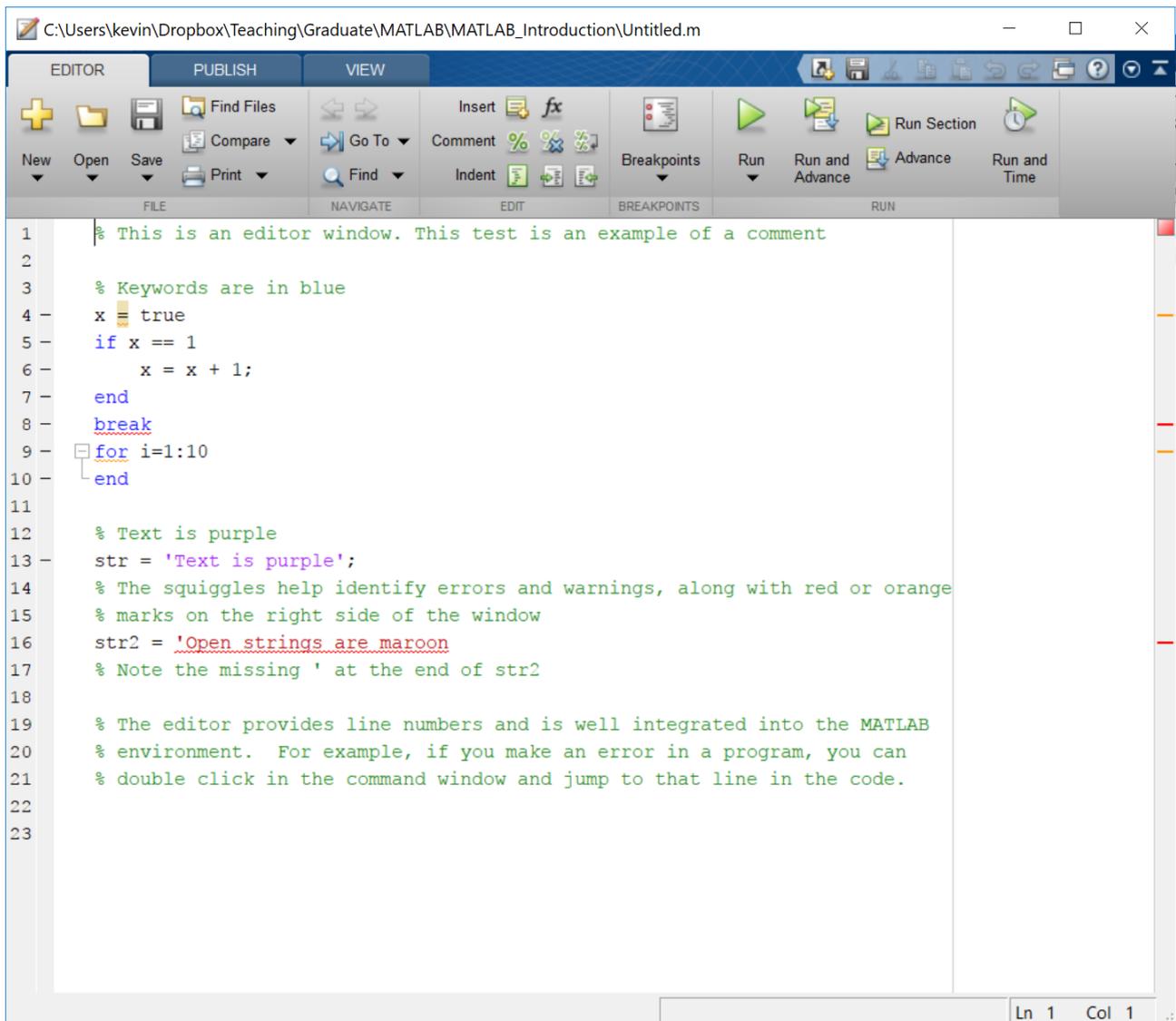


Figure 1.2: The editor.

Problem 2 (Launch the Editor). Open the editor using one of the methods above. Once the editor is open, create a file with the following contents:

```
x = exp(1)
y = log(x)
```

and save it as `lesson1.m`. Go to the command window and enter `lesson1`. The command window should show

```
x = 2.7183
y = 1
```

Problem 3 (Edit an Existing File). Next, close the editor by either clicking on the “X” in the top right corner. A specific file can be opened in the editor by entering `edit filename` in the command window. To verify this works, try the following:

```
>> edit lesson1
```

Problem 4 (Explore the Desktop). Explore alternative views of the MATLAB desktop by changing the configuration using the menu under Home > Layout.

Importing Data into MATLAB

This lesson covers:

- Preparing data for import
- Importing data
- Converting dates

Begin by clearing all data from the workspace by running the command

```
>> clear all
```

The Excel file required for this lesson can be downloaded from

https://www.kevinsheppard.com/MFE_MATLAB

Problem 5 (Using the import wizard). Use the import wizard to import the data contained in excel.xlsx into MATLAB.

- Import into column vectors. Save the imported data as column_data.mat.
- Import into a table as table_data.mat.
- Import into a numeric matrix as numeric_data.mat.

Problem 6 (Read into a table). Use `readtable` to import the data to a MATLAB table.

- Convert the column DATE to a `datetime`.

Problem 7 (Formatting Data in Excel for Import). Format the return data contained in excel.xlsx for import into MATLAB and save the file as both excel_for_import.xlsx and excel_for_import.csv.

Problem 8 (Importing Data). Import the file created in the previous step using both `csvread` and `xlsread` and save the data to excel_imported.mat.

Problem 9 (Converting Dates). Convert the dates imported in the previous step using `x2mdate`.

Exercises

Exercise 1 (Practice). Getting data into and out of MATLAB is very important for your success in the computing portion of the course. Download a year of data containing the price of the SPY ETF from google finance, <https://www.google.com/finance/historical?q=NYSEARCA:SPY>.

- Import it into a table with datetimes
- Import the columns into a numeric array
- Convert the dates to MATLAB dates

Functions

This lesson covers:

- Calling function with more than one input and output
- Calling functions when some inputs are not used
- Writing a custom function

Begin by clearing all data from the workspace and loading the data in `deciles.mat`

```
>> clear all
>> load deciles
```

This data set contains 4 years of data on the 10 CRSP decile portfolios from 2004-2007. The variables are named `decXX` where `XX` ranges from 01 (smallest) to 10 (largest). Also included is a variable named `date` that contains the date in MATLAB date format.

Problem 10 (Calling Functions). Get used to calling functions by computing the `mean`, `std`, `kurtosis`, `max`, and `mean` of the 10 portfolios in `deciles.mat`. Also, explore the help available for calling functions using the `help` and `doc` commands. For example,

```
>> help std
>> doc std
```

Problem 11 (Calling Functions with 2 Outputs). Some useful functions return 2 or more outputs. One example is `sort` which will sort a vector or matrix (column-by-column) and returns the sorted data in the first output and the indices of the elements in the second element. Sort the elements of `dec01` and `dec10`. What is the index of the largest and the smallest elements?

```
>> sort(dec10)
>> [dec10sorted,dec10indices] = sort(dec10)
```

Problem 12 (Calling Functions with 2 Inputs). *Many* functions take two or more inputs. Like outputs, the inputs are simply listed in order separated by commas. Use `linspace` to produce a series of 11 points evenly spaced between 0 and 1. The help for `linspace` is listed below.

```
>> help linspace
linspace Linearly spaced vector.
    linspace(X1, X2) generates a row vector of 100 linearly
    equally spaced points between X1 and X2.

    linspace(X1, X2, N) generates N points between X1 and X2.
    For N = 1, linspace returns X2.

Class support for inputs X1,X2:
    float: double, single

See also linspace, colon.

Reference page for linspace
Other functions named linspace
```

Problem 13 (Calling Functions with Empty Inputs and Outputs). On occasion it is necessary to call a function where some, but not all, inputs are required. One such example is `std`, which takes the data as its first argument, has a flag

for a bias correction as its second argument, and takes the dimension as its third. Compute the standard deviation of each decile, then compute the standard deviation across the individual days (Hint: Down is dim 1, across is dim 2).

```
>> help std
std Standard deviation.
For vectors, Y = std(X) returns the standard deviation. For matrices,
Y is a row vector containing the standard deviation of each column. For
N-D arrays, std operates along the first non-singleton dimension of X.

std normalizes Y by (N-1), where N is the sample size. This is the
sqrt of an unbiased estimator of the variance of the population from
which X is drawn, as long as X consists of independent, identically
distributed samples.

Y = std(X,1) normalizes by N and produces the square root of the second
moment of the sample about its mean. std(X,0) is the same as std(X).

Y = std(X,FLAG,DIM) takes the standard deviation along the dimension
DIM of X. Pass in FLAG==0 to use the default normalization by N-1, or
1 to use N.

std(..., MISSING) specifies how NaN (Not-A-Number) values are treated.
The default is 'includenan':

'includenan' - the standard deviation of a vector containing NaN
values is also NaN.
'omitnan' - elements of X or W containing NaN values are ignored.
If all elements are NaN, the result is NaN.

Example: If X = [4 -2 1; 9 5 7]
then std(X,0,1) is [3.5355 4.9497 4.2426] and std(X,0,2) is [3.0; 2.0]

Class support for input X:
float: double, single

See also cov, mean, var, median, corrcoef.

Reference page for std
Other functions named std
```

Problem 14 (Writing a Custom Function). Custom functions will play an important role later in the course when estimating parameters. Construct a custom function that takes two arguments, μ and σ^2 and computes the likelihood function of a normal random variable

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

Save this function as `normal_likelihood.m` and compute the likelihood of $x = 0, \mu = 0, \sigma^2 = 1$.

Problem 15 (Custom Function). Extend your function written as part of 14 to allow for vector or scalar values of x , μ , and σ^2 (all the same size). Test your function with

```
>> x=[-3; 0; 1];
>> mu = [-3; 0; 2];
```

```
>> sigma2=[9; 4; 1];
```

Exercises

Exercise 2 (Custom Function). Write a function named `summarystats` that will take a single matrix input, `x`, and return a table with 4 columns and as many rows as there were columns in the original data where the columns contain the mean, standard deviation, skewness and kurtosis of `x`.

Accessing Elements in Matrices and Tables

This lesson covers:

- Accessing specific elements in vectors and matrices
- Accessing columns in a [table](#)

Accessing elements in a vector or matrix is a common task. To begin this lesson, clear the workspace, set up a two vectors, `y` and `z`, and a 5×5 matrix `x` using the following commands:

```
>> clear all
>> x=1:25;
>> x=reshape(x,5,5)
x =

     1     6    11    16    21
     2     7    12    17    22
     3     8    13    18    23
     4     9    14    19    24
     5    10    15    20    25
>> y = 1:5
y =
     1     2     3     4     5
>> z = (1:5)'
z =

     1
     2
     3
     4
     5
```

These vectors and matrix will make it easy to determine which elements are selected by a command.

Problem 16 (Picking an Element out of a Matrix). Select the third element of all three, `x`, `y` and `z`. Next select the 11th element of `x`. Finally, using double index notation, select the (1,3) and the (3,1) element of `x` such as in

```
>> x(1,3)
>> x(3,1)
```

Which index is rows and which index is columns? Does MATLAB count across first then down or down first then across?

Problem 17 (Selecting Entire Rows). Select the 2nd row of `x` using the colon (`:`) operator. Select the 2nd column of `z` then `y` using the same syntax. What happens?

Problem 18 (Selecting Entire Columns). Select the 2nd column of `x` using the colon (`:`) operator.

Problem 19 (Selecting Specific Rows or Columns). Select the 2nd and 3rd columns of `x` using the colon (`:`) operator. Select the 2nd and 4th rows of `x`. How can these be combined to select columns 2 and 3 and rows 2 and 4?

Working with tables

Create some tables by entering

```
>> t1 = table(x(:,1), x(:,2), x(:,3), x(:,4), x(:,5))
>> t2 = table(x(:,1), x(:,2), x(:,3), x(:,4), x(:,5), ...
    'VariableNames',{'Alpha','Beta','Delta','Gamma','Epsilon'})
>> nums = [3.14,2.71,1.61]' % Name sure column
>> dates = datetime({'1990-01-01','2000-01-01','2010-01-01'})
>> strings = {'Wilson Phillips','Faith Hill','Ke$ha'}
>> t3 = table(nums, dates, strings,...
    'VariableNames', {'number','date','str'})
```

Problem 20 (Picking an Element out of a Table). Select the element in column 2, row 3 of all three, `t1`, `t2` and `t3` using braces (`{}`), parentheses (`()`) and dot-notation. For the tables with variable names, use both location and names to select the elements.

Problem 21 (Picking a Column from a Table). Select the column 2 of all three tables, `t1`, `t2` and `t3` using braces (`{}`), parentheses (`()`) and dot-notation. For the tables with variable names, use both location and names to select the elements.

Problem 22 (Picking a Table from a Table). Select the columns 1 and 3 of all three tables, `t1`, `t2` and `t3` using braces (`{}`), parentheses (`()`) and dot-notation. For the tables with variable names, use both location and names to select the elements. Note: braces (`{}`) will not work for all tables. Why?

Exercises

Exercise 3 (Selecting Data by Date). Load the data in `deciles.mat` and construct a matrix containing all of the vectors. How can all returns on a particular day be selected? How can all returns for a particular decile be selected?

Exercise 4 (Selecting Data by Date). Load the data in `deciles.mat` and construct a table containing all of the vectors. How can all returns on a particular day be selected? How can all returns for a particular decile be selected?

Program Flow

This lesson covers:

- `for` loops
- Nested loops

Problem 23 (Basic For Loops). Construct a `for` loop to sum the numbers between 1 and N for any N . A `for` loop that does nothing can be written

```
N = 10;
for i=1:N
end
```

Problem 24 (Compute a compound return). The compound return on a bond that pays interest annually at rate r is given by

$$c r_t = \prod_{i=1}^T (1 + r) = (1 + r)^T$$

Use a `for` loop compute the total return for £100 invested today for 1,2,...,10 years when r is 6.3%. Store this variable in a 10 by 1 vector `cr`.

Problem 25 (Simulate a random walk). (Pseudo) Normal random variables can be simulated using the command `randn(N, M)` where N and M are the dimensions of the desired random numbers. Simulate 100 normals in a 100 by 1 vector and name the result e . Initialize a vector p containing zeros using the function `zeros`. Add the 1st element of e to the first element of p . Use a `for` loop to simulate a process

$$y_i = y_{i-1} + e_i$$

When finished plot the results using

```
>> plot(y)
```

Problem 26 (Simulate a random walk with stochastic volatility). (Pseudo) Normal random variables can be simulated using the command `randn(N, M)` where N and M are the dimensions of the desired random numbers. Simulate 100 normals in a 250 by 1 vector and name the result e . Initialize a vector p containing zeros using the function `zeros`. Add the 1st element of e to the first element of p . Use a `for` loop to simulate a process

$$y_i = y_{i-1} + \sigma_i e_i$$

$$\ln \sigma_i = 0.99 \ln \sigma_{i-1} + 0.11 u_i$$

Where u_i and e_i are independent normals. Assume $\sigma_0 = 1$. When finished plot both the simulated price series and the simulated standard deviations (not the log std deviations) the results using

```
>> subplot(2,1,1)
>> plot(1:100, y)
>> subplot(2,1,2)
>> plot(1:100, sigma)
```

Problem 27 (Nested Loops). Begin by clearing the workspace and loading `deciles.mat`. Begin by adding 1 to the returns to produce *gross returns*.³ Use two loops to loop both across time and across the 10 deciles to compute the total compound return. For example, if only interested in a single series, this

```
cr=zeros(size(dec01));
gr = 1 + dec01;
cr(1) = 1+dec01(1);
T=10;
for t=2:T
    cr(t)=cr(t-1)*gr(t);
end
```

would compute the cumulative return. When finished, plot the cumulative returns using `plot(cr)`. After finishing this assignment, have a look at `doc cumsum` and `doc cumprod`.

Exercises

Exercise 5. Simulate a 1000 by 10 matrix consisting of 10 standard random walks using both nested loops and `cumsum`. Plot the results. If you rerun the code, do the results change? Why?

Logical Operators

This lesson covers:

- Basic logical operators

³A gross return is the total the value in the current period of £1 invested in the previous period. A net return subtracts the original investment to produce the net gain or loss.

- Compound operators
- Mixing logic and loops
- `all` and `any`

Begin by clearing all data and loading the data in `deciles.mat`

Problem 28 (Basic Logical Statements). For decile 1 and decile 10, count the number of elements that are < 0 , ≥ 0 and exactly equal to 0. Next count the number of times that the returns in decile 5 are greater, in absolute value, that 2 times the standard deviation of the returns in that decile.

Problem 29 (Compound Statements). Count the number of times that the returns in *both* decile 1 and decile 10 are negative. Next count the number of times that the returns in deciles 1 and 10 are *both* greater, in absolute value, that 2 times their respective standard deviations.

Problem 30 (Logical Statements and for Loops). Use a `for` loop along with an `if` statement to simulate an asymmetric random walk of the form

$$y_i = y_{i-1} + e_i + I_{[e_i < 0]} e_i$$

where $I_{[e_i < 0]}$ is known as an indicator variable that takes the value 1 if the statement in brackets is true and where e_i is standard normal. Plot y .

Problem 31 (Selecting Elements using Logical Statements). For decile 1 and decile 10, select the elements that are < 0 , ≥ 0 and exactly equal to 0. Next select the elements where both deciles are less than 0.

Problem 32 (Using `find`). Use `find` to select the index of the elements in decile 5 that are negative. Next, use the `find` command in its two output form to determine which elements of the decile return matrix are less than -2%.

Exercises

Exercise 6 (`all` and `any`). Use `all` to determine the number of days where all of the decile returns were negative. Use `any` to compute the number of days with at least 1 negative return and with no negative returns (Hint: use negation (`~`)).

Graphics

This lesson covers:

- Basic plotting
- Editing plots
- Subplots
- Histograms

Begin by clearing all data from the workspace and loading the data in `hf.mat`. This data set contains high-frequency price for IBM and MSFT on a single day and times in MATLAB format. It also contains two tables containing the same data.

Exercises

Exercise 7 (Basic Plotting). Plot the series labeled `IBMprice` which contains the price of IBM. Add a title and label the axes. Use the interactive tool to add markers and remove the line.

Exercise 8 (Basic Plotting of Tables). Plot the series labeled `price` in the table `IBM` which contains the price of IBM using red dots in a single command.

Exercise 9 (Subplot). Create a 2 by 1 subplot with the price of IBM in the top subplot and the price of MSFT in the bottom subplot using the price data in the tables.

Exercise 10 (Plot with Dates). Plot the price of IBM against the series `IBMtimes`. Use `datetick` to reformat the x-axis.

Exercise 11 (Plot with Dates using data in a table). Plot the price of IBM against the series `time` using the data in `IBM`.

Exercise 12 (Histogram). Produce a histogram of MSFT returns (Hint: you have to produce the Microsoft returns first).

Data and Simulation

Data Set Construction

Note: In all cases, convert the dates to both MATLAB date format and YYYYMMDD.

Functions

`csvread`, `xlsread`, `weekday`, `textscan`, `str2double`, `c2mdate`, `diff`

Exercise 13. Download all daily data for the S&P 500 and the FTSE 100 from Yahoo! Finance.

- Import both data sets into MATLAB, converting dates to MATLAB date format.
- Construct weekly price series from each, using Tuesday prices (less likely to be a holiday).
- Construct monthly price series from each using last day in the month.

Exercise 14. Write a function which will return month-end prices. The function signature should be

```
function [month_end_price, month_end_date] = month_end_prices(price,date)
```

Exercise 15. Import the Fama-French benchmark portfolios as well as the 25 sorted portfolios at both the monthly and daily horizon.

http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

- Import daily FX rate data for USD against AUD, Euro, JPY and GBP from the Federal Reserve Economic Database (FRED)
<http://research.stlouisfed.org/fred2/categories/94>

Note: `c2mdate` is in the MFE toolbox.

Simulation

Functions

`randn`, `trnd`, `rng`

Exercise 16. Simulate 100 standard Normal random variables

Exercise 17. Simulate 100 random variables from a $N(.08, .2^2)$

Exercise 18. Simulate 100 random variables from a Student's t with 8 degrees of freedom

Exercise 19. Simulate 100 random variables from a Students t with 8 degrees of freedom with a mean of 8% and a volatility of 20%. Note: $V[X] = \frac{v}{v-2}$ when $X \sim t_v$.

Exercise 20. Simulate two identical sets of 100 standard normal random variables by resetting the random number generator.

Exercise 21. Repeat exercise 18 using *only* `randn`.

Expectations

Functions

`randn`, `trnd`, `chi2rnd`, `exp`, `mean`, `std`, `integral`, `quadl`

Exercise 22. Compute $E[X]$, $E[X^2]$, $V[X]$ and the kurtosis of X using Monte Carlo integration when X is distributed:

- Standard Normal
- $N(0.08, 0.2^2)$
- Students t_8
- χ_5^2

Exercise 23. Compute $E[\exp(X)]$ when $X \sim N(0.08, 0.2^2)$. Compare this to the analytical result for a Log-Normal random variable.

Exercise 24. Explore the role of uncertainty in Monte Carlo integration by using $4 \times$ as many simulations as in the base case.

Exercise 25. Compute the expectation in exercise 23 using quadrature. Note: This requires writing a function which will return $\exp(x) \times \phi(x)$ where $\phi(x)$ is the probability form the pdf.

Exercise 26. [Much more challenging] Suppose log stock market returns are distributed according to a Students t with 8 degrees of freedom, mean 8% and volatility 20%. Utility maximizers hold a portfolio consisting of a risk-free asset paying 1% and the stock market. Assume that they are myopic and only care about next period wealth, so that

$$U(W_{t+1}) = U(\exp(r_p) W_t)$$

and that $U(W) = \frac{W^{1-\gamma}}{1-\gamma}$ is CRRA with risk aversion γ . The portfolio return is $r_p = w r_s + (1-w) r_f$ where s is for stock market and f is for risk-free. A 4th order expansion of this utility around the expected wealth next period is

$$E_t[U(W_{t+1})] \approx \phi_0 + \phi_1 \mu'_1 + \phi_2 \mu'_2 + \phi_3 \mu'_3 + \phi_4 \mu'_4$$

where

$$\phi_j = (j!)^{-1} U^{(j)}(E_t[W_{t+1}]), \quad U^{(j)} = \frac{\partial^j U}{\partial W^j} \quad \text{and} \quad \mu'_k = E_t[(r - \mu_p)^k]$$

and

$$\mu = E_t[r_p].$$

Use Monte Carlo integration to examine how the weight in the stock market varies as the risk aversion varies from 1.5 to 10. Note that when $\gamma = 1$, $U(W) = \ln(W)$. Use $W_t = 1$ without loss of generality since the portfolio problem is homogeneous of degree 0 in wealth.

Estimation and Inference

Method of Moments

Functions

`mean`, `sum`, `subplot`, `plot`

Exercise 27. Estimate the mean, variance, skewness and kurtosis of the S&P 500 and FTSE 100 using the method of moments using monthly data.

Exercise 28. Estimate the asymptotic covariance of the mean and variance of the two series (separately, but not the skewness and kurtosis).

Exercise 29. Estimate the Sharpe ratio of the two series and compute the asymptotic variance of the Sharpe ratio. See Chapter 2 of the notes for more on this problem.

Exercise 30. Plot rolling estimates of these using 120 months of consecutive data using a 4 by 1 subplot against the dates.

Maximum Likelihood

Functions

`log`, `gamma`, `gammaln`, `normcdf`, `erf`, `fminunc`, `fmincon`, `trnd`, `var`, `std`, `normpdf`

Exercise 31. Simulate a set of i.i.d. Student's t random variables with degree of freedom parameter $\nu = 10$. Standardize the residuals so that they have unit variance using the fact that $V[x] = \frac{\nu}{\nu-2}$. Use these to estimate the degree of freedom using maximum likelihood. Note that the likelihood of a standardized Student's t is

$$f(x; \nu, \mu, \sigma_t^2) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)} \frac{1}{\sqrt{\pi(\nu-2)}} \frac{1}{\sigma_t} \frac{1}{\left(1 + \frac{(x-\mu)^2}{\sigma_t^2(\nu-2)}\right)^{\frac{\nu+1}{2}}}$$

where $\Gamma()$ is known as the gamma function.

Exercise 32. Repeat the previous exercise using daily, weekly and monthly S&P 500 and FTSE 100 data. Note that it is necessary to remove the mean and standardize by the standard deviation error before estimating the degree of freedom. What happens over longer horizons?

Exercise 33. Repeat the previous problem by estimating the mean and variance simultaneously with the degree of freedom parameter.

Exercise 34. Simulate a set of Bernoulli random variables y_i where $p_i = \Phi(x_i)$ where $X_i \sim N(0, 1)$. (Note: p_i is the probability of success and $\Phi()$ is the standard Normal CDF). Use this simulated data to estimate the Probit model where

$$p_i = \Phi(\alpha_0 + \alpha_1 x_i)$$

using maximum likelihood.

Exercise 35. Estimate the asymptotic covariance of the estimated parameters in exercise 34.

Bias and Verification of Standard Errors

Functions

`sum`, `mean`, `cov`, `bsxfun`, `fminunc`, `normcdf`, `normpdf`, `norminv`, `pltdens`, `chi2rnd`, `rand`, `plot`, `title`, `axis`, `legend`

Exercise 36. Simulate a set of i.i.d. χ_5^2 random variables and use the method of moments to estimate the mean and variance.

Exercise 37. Compute the asymptotic variance of the method of moment estimator.

Exercise 38. Repeat 36 and 37 a total of 1000 times. Examine the finite sample bias of these estimators relative to the true values.

Exercise 39. Repeat 36 and 37 a total of 1000 times. Compare the covariance of the estimated means and variance (1000 of each) to the asymptotic covariance of the parameters (use the average of the 1000 estimated variance-covariances). Are these close? How does the sample size affect this?

Exercise 40. In the previous problem, for each parameter, form a standardized parameter estimate as

$$z_i = \frac{\sqrt{n} (\hat{\theta}_i - \theta_{i,0})}{\sqrt{\hat{\Sigma}_{ii}}}$$

where $\sqrt{n} (\hat{\theta} - \theta_0) \xrightarrow{d} N(0, \Sigma)$ so that $\hat{\Sigma}$ is the estimated asymptotic covariance. What percent of these z_i are larger in absolute value than 10%, 5% and 1% 2-sided critical values from a normal?

Exercise 41. Produce a density plot of the z_i standardized parameters and compare to a standard normal.

Exercise 42. Repeat the same exercise for the Bernoulli problem from the previous question.

Linear Regression

Basic Linear Regression

Functions

`mean`, `bsxfun`, `repmat`

Exercise 43. Write a function to estimate the parameters of a linear regression and compute related quantities. The function signature should match

`[b, tstat, s2, VCV, VCV_white, R2, Rbar, yhat] = ols(y, x, c)` where the inputs are

- `y`: A n by 1 vector of dependent variables
- `x`: A n by K vector of independent variables
- `c`: A scalar, 1 if the model should include a constant. If not, then no constant will be added to the
- `x` matrix.
and where the outputs are
- `b`: The estimated coefficients
- `tstat`: The t-stats of the estimated coefficients
- `s2`: The estimated variance of residual
- `VCV`: The K by K covariance matrix of the estimated parameters
- `VCVwhite`: The K by K heteroskedasticity robust covariance matrix of the estimated parameters
- `R2`: The R^2 of the regression
- `Rbar`: The \bar{R}^2 of the regression
- `yhat`: The fit values \hat{y} .

Exercise 44. Use the OLS function to estimate the coefficients of the Fama-French portfolios (monthly data) on the market, size and value factors. Use only the four extremum portfolios – that is the 1-1, 1-5, 5-1 and 5-5 portfolios.

Exercise 45. Are the parameter standard errors similar using the two covariance estimators? If not, what does this mean?

Exercise 46. How much of the variation is explained by these three regressors?

Rolling and Recursive Regressions

Functions

`ols`, `title`, `datetick`, `legend`, `axis`, `subplot`, `plot`, `figure`

Exercise 47. For the same portfolios in the previous exercise, compute rolling β s using 60 consecutive observations.

Exercise 48. For each of the four *market* β s, produce a plot containing four series:

- A line corresponding to the constant β (full sample)
- The β estimated on the rolling sample
- The constant β plus $1.96 \times$ the variance of a 60-observation estimate of β .¹
- The constant β minus $1.96 \times$ the variance of a 60-observation estimate of β .

Exercise 49. Do the market exposures appear constant?

Exercise 50. What happens if only the market is used as a factor (repeat the exercise excluding SMB and HML).

Exercise 51. In problems 1 and 2, is there any evidence of time-variation in the SMB or HML loadings?

Model Selection and Cross-Validation

Functions

`randperm`, `ols`, `setdiff`, `norminv`, `linspace`, `mean`

Exercise 52. For these portfolios, and considering all 8 sets of regressors which range from no factor to all 3 factors, which model is preferred by AIC, BIC, GtS and StG?

Exercise 53. Cross-validation is a method of analyzing the in-sample forecasting ability of a cross-sectional model by using $\alpha\%$ of the data to estimate the model and then measuring the fit using the remaining $100 - \alpha\%$. The most common forms are 5- and 10-fold cross-validation which use $\alpha = 20\%$ and 10% , respectively. k -fold cross validation is implemented by randomly grouping the data into k -equally-sized groups, using $k - 1$ of the groups to estimate parameters, and then evaluating using the bin that was held out. This is then repeated so that each bin is held out.

- Implement cross-validation using the 5- and 10-fold methods for all 8 models.
- For each model, compute the full-sample sum of squared errors as well as the sum-of-squared errors using the held-out sample. Note that all data points will appear exactly once in both of these sum of squared errors. What happens to the cross-validated R^2 when computed on the held out sample when compared to the full-sample R^2 ? (k -fold cross validated SSE by the TSS).

¹The 60-month covariance can be estimated using a full sample VCV and rescaling it by $T/60$ where T is the length of the full sample used to estimate the VCV. Alternatively, the VCV could be estimated by first estimating the 60-month VCV for each sub-sample and then averaging these.

Time-series Modeling

ARMA Estimation

Functions

ARMAX, [armaxfilter](#)

Exercise 54. Download data on 1 year and 10 year US government bond rates from FRED, and construct the term premium as the different in yields on 10 year and 1 year bonds.

Exercise 55. Estimate an AR(1) on the term premium, and compute standard errors for the parameters.

Exercise 56. Estimate an MA(5) on the term premium, and compute standard errors for the parameters.

Exercise 57. Estimate an ARMA(1,1) on the term premium, and compute standard errors for the parameters.

ARMA Model Selection

Functions

[armaxfilter](#)

Exercise 58. Perform a model selection exercise on the term premium using

- a. General-to-Specific
- b. Specific-to-General
- c. Minimizing an Information Criteria

Note: When comparing models with different AR orders, it is important to use the holdback input.

ARMA Residual Diagnostics

Functions

[aramaxfilter](#), [plot](#), [lmtest1](#), [ljungbox](#)

Exercise 59. Compute the residuals from your preferred model from the previous exercise, as well as a random-walk model.

- a. Plot the residuals

- b. Is there evidence of autocorrelation in the residuals?
- c. Compute the Q statistic from both sets of residuals. Is there evidence of serial correlation?
- d. Compute the LM test for serial correlation. Is there evidence of serial correlation?

ARMA Forecasting

Functions

`arma_forecaster`, `cov_nw`, `ols`

Exercise 60. Produce 1-step forecasts from your preferred model in the previous exercise, as well as a random-walk model.

- a. Are the forecasts objectively accurate?
- b. Compare these forecasts to the random walk models using MSE and MAE.

Exercise 61. Produce 3-step forecasts from the models selected in the previous exercises as well as a random walk model.

- a. Compare these forecasts to the random walk models using MSE and MAE.

Note: Use 50% of the sample to estimate the model and 50% to evaluate it.

Unit Root Testing

Functions

`augdf`, `augdfautolag`

Exercise 62. Download data on the AAA and BAA yeilds (Moody's) from FRED and construct the default premium.

- a. Test the default premium for a unit root.
- b. If you find a unit root, test the change.

Exercise 63. Download data on consumer prices in the UK from the ONS.

- a. Test the log of CPI for a unit root.
- b. If you find a unit root, test inflation for one.

Volatility Modeling

ARCH Model Estimation

Functions

`tarch`

Exercise 64. Download 10 years of data for the S&P 500 and the EUR/USD exchange rate from FRED.

Exercise 65. Estimate a GARCH(1,1) and a GJR-GARCH(1,1,1) to the returns of both series.

Exercise 66. Comment on the asymmetry.

- Compare robust and non-robust standard errors.
- Plot the fit variance and fit volatility.
- Plot the standardized residuals.

ARCH Model Selection

Functions

`tarch`, `egarch`

Exercise 67. Which model is selected for the S&P 500 among the classes:

- TARCH
- GJR-GARCH
- EGARCH

ARCH Model Forecasting

Functions

`tarch`, `egarch`

Exercise 68. Use 50% of the sample to estimate your preferred GARCH model for returns to the S&P 500 and the EUR/USD rate, and construct forecasts for the remaining period.

Exercise 69. Evaluate the accuracy of the forecasts.

Exercise 70. Evaluate the accuracy of forecasts from a 2-year backward moving average variance.

Exercise 71. Compare the ARCH-model forecasts to a naive 2-year backward looking moving average using QLIKE.

Value-at-Risk Modeling

Value-at-Risk using Historical Simulation

Functions

`rand`

Exercise 72. Compute the historical simulation VaR for the S&P 500 and the EUR/USD rate.

Note: Start the historical simulation at 25% of the data, and then build the additional forecasts using a recursive scheme.

Value-at-Risk using Filtered Historical Simulation

Functions

`tarch`

Exercise 73. Use a GARCH(1,1) model to construct filtered historical VaR for the S&P 500 and the EUR/USD exchange rate, using 10 years of data.

Exercise 74. Compare this VaR to the HS VaR in the previous example.

Note: For simplicity, estimate the model on the full sample, but start the historical simulation at 25% of the data, and then build the additional forecasts using a recursive scheme.

Value-at-Risk Forecast Evaluation

Functions

`ols`

Exercise 75. Evaluate the FHS and HS VaR forecasts constructed in the previous exercises using:

- a. HIT tests
- b. The Bernoulli test for unconditionally correct VaR
- c. Christoffersen's test for conditionally correct VaR

Vector Autoregressions

Vector Autoregression (VAR) Estimation

Functions

`vectorar`, `eig`

Exercise 76. Download data on 10-year interest rates, 1-year interest rates and the GDP deflator from FRED.

Exercise 77. Transform the GDP deflator to be percent returns (e.g. $\Delta \ln gdp_t$).

Exercise 78. Estimate a first-order VAR on the spread between the 10-year and 1-year (*spread*), the one-year, and the growth rate of the GDP deflator.

Exercise 79. What are the “own” effects?

Exercise 80. What are the cross effects between these?

Exercise 81. How could you get a sense of the persistence of this system?

VAR Model Order Selection

Functions

`vectorar`

Exercise 82. Using the same data as in the previous exercise, determine the optimal VAR order using:

- a. AIC
- b. HQIC
- c. BIC
- d. Likelihood-ratio testing using General-to-Specific

Granger Causality Testing

Functions

`grangercause`

Exercise 83. Using the data and the models selected in the previous exercise, is there evidence of Granger Causality between the series?

Exercise 84. What if the 10 year and the 1 year are both used, but the spread is omitted?

Impulse Responses

Functions

`impulseresponse`

Exercise 85. Plot the impulse responses from both a first order model and the model selected in the order selection exercise.

Exercise 86. Which covariance factor makes sense for the impulse responses?

Exercise 87. What happens when you re-order the series and use the Choleski factor in the impulse response?

- a. Which series should be first?
- b. Which should be last?

Cointegration Testing using Engle-Granger

Functions

`augdf`, `augdfautolag`

Exercise 88. Download data on *CAY* from Sydney Ludvigson's site.

Exercise 89. Is there evidence that these three series are cointegrated in the entire sample?

Exercise 90. What about in the post-Volker era (start in 1981)?