

A Quick Guide to EViews 4.0

1	Introduction	2
1.1	Where To Go For Help	2
2	EViews Basics	3
2.1	The EViews Window	3
2.2	Objects	3
2.3	EViews Workfiles	4
2.4	Creating a New Workfile	4
2.5	Importing Data	5
2.6	Working with Series (Variables)	6
2.7	The Workfile Sample	7
3	Descriptive Statistics	8
4	Ordinary Least Square Regression	9
5	Maximum Likelihood Estimation	11
6	Pooled Time Series, Cross -Section Data	13
7	Some EViews Commands	15
	Data Handling	15
	Descriptive Statistics	15
	Graphs	16
	Estimation	16
	Tests	17
8	Important Operators	18

1 Introduction

EViews provides data analysis, estimation, and forecasting tools. The immediate predecessor of EViews was MicroTSP, first released in 1981.

EViews takes advantage of the visual features of modern Windows software. You can use your mouse to guide the operation with standard Windows menus and dialogs. Results appear in windows and can be manipulated with standard Windows techniques. Alternatively, you may use EViews' command and batch processing language. You can enter and edit commands in the command window. You can create and store the commands in programs that document your research project for later execution.

1.1 Where To Go For Help

The *User's Guide* explains the visual approach to using EViews. The *Command and Programming Reference*, provides systematic information about the details of EViews commands. A large portion of the EViews documentation may be called up from within EViews by using the help system. Select **Help** in the main menu. You will find on the EViews web site a Frequently Asked Questions (FAQ) section: <http://www.eviews.com>.

2 EViews Basics

2.1 The EViews Window

You should see the EViews window when you launch the program. Familiarise yourself with the following areas in the EViews Window:

Just below the title bar is the *main menu*. Below the menu bar is an area called the *command window*. EViews commands may be typed in this window. The command is executed as soon as you hit **Enter**. The contents of the command area may be saved directly into a text file for later use: make certain that the command window is active, and then select **File/Save As...**. Some important EViews Commands are listed in Section 5.

The area in the middle of the window is the work area where EViews will display the workfile window the various other windows that it creates.

2.2 Objects

Information in EViews is stored in objects. Each object consists of a collection of information related to a particular area of analysis. Associated with each type of object is a window and a set of views and procedures which can be used in association with the information contained in the object. Views are tabular and graphical windows that provide various ways of looking at the data in an object. If you double click on a object, you open the object window. EViews will display the view that was displayed the last time the object was opened.

The most fundamental objects in EViews are series and equation objects. A *series* object contains the data on a particular variable, information on how it was generated, descriptive statistics, the histogram and other graphs.

An *equation object* contains the coefficients obtained from estimation of the equation, but also a description of the specification, the variance-covariance matrix of the coefficient estimates, and a variety of statistics associated with the estimates. There are, however, a number of different types of objects, each of which serves a unique function.

2.3 EViews Workfiles

EViews stores the data as well as estimation specifications and estimation results in a *workfile*. Workfiles can be saved with **File/Save As...** or **File/Save...**. You can use **File/Open/Workfile...** to bring back a previously saved workfile.

After you have opened a new or previously save workfile, you will see the workfile window. In normal display mode, all named objects, such as variables and estimations are listed in the workfile window. There are two objects every workfile: a vector of coefficients, *C*, and a series of residuals, *RESID*.

2.4 Creating a New Workfile

One way to create a workfile is to click **File/New/Workfile...** and then to provide the necessary dialog information. Select the appropriate frequency and enter the information for the workfile range. The *Start date* is the earliest date or observation you plan to use in the project and the *End date* is the latest date or observation. You can expand the workfile range later. EViews describes data the following way: year (e.g. 1981); year:quarter (e.g. 1992:1); year:month. (e.g. 1990:11); month:day:year (e.g. 8:10:97). Cross-sectional data is stored as *undated or irregular*.

2.5 Importing Data

You can read data directly from files created by other programs. Data may be in standard ASCII form or in either Lotus (.WKS, .WK1 or .WK3) or Excel (.XLS) spreadsheet formats. First make certain that you have an open EViews workfile to receive the contents of the data import. Next, click on Procs/Import/Read Text-Lotus-Excel... You will see a standard File dialog box asking you to specify the type and name of the file.

EViews will prompt you for additional information about the import procedure. The dialog will differ depending on the file type. Most fields of the dialog box are reasonably self-explaining. You need to tell whether the data are ordered by observation or by series. "By observation" means that variables are arranged in columns while "by row" implies that all of the observations for a variable are in a single row.

Enter the names of the series that you wish to read into the edit box. Alternatively, if the names that you wish to use for your series are contained in the file, you can simply provide the number of series to be read. The names must be adjacent to your data. If the data are organised by row and the starting cell is B2, then the names must be in column A, beginning at cell A2. If the data are organised by column beginning in B2, then the names must be in row 1, starting in cell B1. If, in the course of reading the data, EViews encounters an invalid cellname, it will automatically assign the next unused name with the prefix SER, followed by a number (e.g., SER01, SER02, etc.).

2.6 Working with Series (Variables)

To create or modify a variable (called *series* in EViews), select Quick/Generate Series.... EViews prompts you for additional information. Alternatively, series are generated with the `genr` or `series` Command. You can type the series name, followed by an equals sign and then an expression. For every element of the sample, EViews will evaluate the expression on the right-hand side of the equality, and assign the value to the destination series on the left-hand side, creating the series if necessary. All observations *not* in the current sample will be unchanged. EViews fills an observation with Na (not available) if it does not assign a value. There is a variety of commands for mathematical, logical, time lag operators, etc.. See section 8 for important operators.

Some examples:

<code>series y</code>	creates a new Y series; all observations are filled with Na
<code>series y = 2*x + 37*z</code>	creates the Y series as a function of the X and Z series
<code>series y = log(y)</code>	creates the Y series as the natural logarithm of the Y series itself
<code>series y = x(- 4)</code>	creates the Y series as the fourth lag of the X series
<code>series y = x(2)</code>	creates the Y series as the second lead of the X series
<code>series y = @movav(x,6)</code>	creates the Y series as a 6-period moving average of the X series
<code>series y = x < 3</code>	the Y dummy takes value 1 if X is less than 3 and 0 otherwise

2.7 The Workfile Sample

The sample is the set of observations in the workfile to be included in displays and in statistical procedures. Samples may be specified using ranges of observations and/or conditions that observations must satisfy to be included. The sample is set initially to be the entire range of the workfile. You can always tell the current workfile sample of observations by looking at the top of your workfile window.

To change the sample, select Quick/Sample... from the menu or use the sample Command. EViews displays a Sample dialog prompting you for additional information. In the upper edit window you will enter one or more *pairs of dates*. The lower part of the sample window allows you to add *conditions* to the sample specification. For example,

Upper window: 1958:1 1998:4

includes all observations between the first quarter of 1958 and the last quarter of 1998.

Lower window: income >= 4000

includes observations, where the series INCOME is greater than or equal to 4000.

Upper window: 50 100 200 250

Lower window: (ed>=6 and ed<=13) or earn<@mean(earn)

includes undated workfile observations 50 through 100 and 200 through 250, where the value of the variable ED falls between 6 and 13, or where the value of the variable EARN is lower than its mean.

Upper window: @all

Lower window:

sets the sample to the entire workfile.

3 Descriptive Statistics

EViews provides both descriptive statistics of a single series (variable) or of a group of selected series.

To access descriptive statistics of *a single series*, open the series window by double clicking on the series name in the workfile. Select View/Descriptive Stats/ Histogram and Stats. A complement of standard descriptive statistics are displayed along with the histogram. All of the statistics are calculated using observations in the current sample. Alternatively, you can access descriptive statistics with the menu option Quick/Series Statistics/Histogram and Stats or the histcommand.

To access descriptive statistics of *a group of series*, select Quick/Group Statistics/ Descriptive Statistics and enter the names of the series to be included. There are two alternatives to deal with missing observations: Common Sample computes the statistics using observations for which there are no missing values in any of the series in the group. Individual Samples computes the statistics using all nonmissing observations for each series. Alternatively, you can use the stats command.

4 Ordinary Least Square Regression

Single equation regression estimation in EViews is performed using the equation object. To create an equation object in EViews: select Objects/New Object/Equation or Quick/Estimate Equation from the main menu, or simply type the keyword `equation` in the command window. Next, you will specify your equation in the Equation Specification dialog box that appears, and select an estimation method.

The simplest way to specify a linear equation is to provide a list of variables that you wish to use in the equation. First, include the name of the *dependent variable*, followed by a list of *explanatory variables*. For example, to specify a linear consumption function, CS regressed on a constant, the variable INC and the variable R, type the following in the upper field of the equation specification dialog:

```
cs c inc r
```

Alternatively, type the least square command (`ls`) followed by the equation specification in the command window:

```
ls cs c inc r
```

Note the presence of the series name C in the list of regressors. This is a built-in EViews series that is used to specify a constant in a regression. EViews does not automatically include a constant in a regression.

You may include transformed variables in a regression. For example,

```
ls log(cs) c log(inc)
```

specifies a regression of the natural logarithm of CS on a constant and the natural logarithm of INC.

Another way to specify the equation is using a formula. Simply enter the

expression in the dialog in place of the list of variables. EViews will add an implicit additive disturbance to this equation. For example,

```
ls cs = c(1) + c(2)*inc + c(3)*r
```

is the same regression as above.

Having specified your equation, you now need to choose an estimation method. Standard, single-equation regression is performed using the LS - Least Squares method. EViews will use the current workfile sample if you do not specify it in the sample edit box. EViews provides a number of estimation options. These options allow you to weight the estimating equation, to compute heteroskedasticity and auto-correlation robust covariances, and to control various features of your estimation algorithm.

The estimation results are displayed in the stats view of the equation window. EViews reports the estimated *coefficients*, the estimated *standard errors* of the coefficient estimates, the *t-statistic* of the hypothesis that a coefficient is equal to zero, and the *probability* of drawing a t-statistic as extreme as the one actually observed, under the assumption that the errors are normally distributed. You can view the whole covariance matrix choosing View/Covariance Matrix. The residuals of the previously performed estimation are stored in the series RESID.

Following estimation, EViews holds the estimated coefficients and their covariance matrix, the residuals, and some summary statistics in the equation object. These results are available for use in a variety of subsequent computations, including Specification and Diagnostic Tests, Forecasting, and Model Simulation. You may use the Name button to name your equation object. The equation will be saved with the workfile when the latter is saved. Untitled equations will be overwritten by the subsequent estimation and will not be saved with the workfile.

5 Maximum Likelihood Estimation

EViews provides you with a tool to estimate a wide variety of specific actions through the log likelihood (LogL) object. To create a likelihood object, choose Objects/New Object.../LogL or type `logl` in the command window. The likelihood window will open with a blank specification view. The specification view is a text window into which you enter a list of statements which describe your statistical model, and in which you set options which control various aspects of the estimation procedure.

Each likelihood specification must contain a control statement which provides the name of the series which is used to contain the likelihood contributions. The format of this statement is

```
@logl series_name
```

where `series_name` is the name of the series which will contain the contributions. This control statement may appear anywhere in the LogL specification. If you would like to remove one or more of the series used in the specification after evaluation, you can use the `@temp` statement:

```
@temp series_name1 series_name2 ...
```

An Example: ML-Estimation of a linear regression model

```
@logl logLi
@temp e
e = y - c(1) - c(2)*x1 - c(3)*x2
logLi = log(1/c(4)*@dnorm(e/c(4)))
```

After estimating the above example, the elements C(1), C(2), C(3) and C(4) will contain ML-estimates of the constant, two slope coefficients and the standard deviation of the error term.

The choice of *starting values* is important. For problems in which the likelihood function is globally concave, it will influence how many iterations are taken for estimation to converge. For problems where the likelihood function is not concave, it may determine which of several local maxima is found. In some cases, estimation will fail unless reasonable starting values are provided. By default, EViews uses the values stored in the coefficient vector or vectors prior to estimation. You can arbitrarily set a parameter starting value using a statement in the command line:

```
c(4) = 1
```

An alternative method of initializing the parameters to known values is to include a `@param` statement in the likelihood specification. For example,

```
@param c(4) 1
```

In the example above, we used the coefficients C(1) to C(4) as names for our unknown parameters. More generally, any element of a named coefficient vector which appears in the specification will be treated as a parameter to be estimated. You can create your own coefficient vectors by the following commands in the command line,

```
coef(1) alpha
coef(2) beta
coef(1) sigma
```

You could then write the likelihood specification as

```
@logl logLi
@temp e
e = y - alpha(1) - beta(1)*x1 - beta(2)*x2
logLi = log(1/sigma(1)*@dnorm(e/sigma(1)))
@param sigma(1) 1
```

6 Pooled Time Series, Cross-Section Data

EViews provides a number of specialized tools to help you work with pooled data. The EViews object that manages time series/cross-section data is called a *pool*.

Your first step will be to *create a workfile* to hold your pooled data. There is nothing special about the creation of this workfile. Simply select File/New/Workfile... and enter the frequency and dates for your workfile. The range of your workfile should represent the earliest and latest dates that you wish to consider for any of the cross-section units.

At the heart of a pool object is a list of (short) names that you will use to refer to cross-section members. For example, in a cross-country study, you might use “USA” to refer to the United States, “CAN” to refer to Canada and so on. It is important to note that a pool object is simply a description of the underlying structure of your data and does not contain the data.

To *create a pool object*, select Objects/New Object/Pool... and enter the identifiers in the edit window that will open automatically, e.g.:

```
Cross Section Identifiers: (Enter identifiers below this line)
_USA
...
_CAN
```

Even though you will be working with pooled data, all of your data will be held in ordinary EViews series. The key to using series with pools is to *name your series* using a combination of a base name and the cross-section identifier. The cross-section identifier may be embedded at an arbitrary location in the series name, so long as this is done consistently across identifiers. Consider the above example and that you have time series measurements on GDP for each of the cross-section units. We will use GDP as the

base name for our series and name the series GDP_USA, ... GDP_CAN. See the EViews Help on how to import time series, cross-section data.

The key to work with a set of series is the concept of a *pool series*. A pool series is a group of series defined by a base name and the entire list of cross-section identifiers. Pool series are specified using the base name, and a “?” character for the cross-section identifier. If your series are named GDP_USA, ... the corresponding pool series will be referred to as GDP?. You can view your pool series by selecting View/Spreadsheet View, and listing the pool series name. You may compute descriptive statistics of a pool series with View/Descriptive Statistics. You can generate or modify pool series using the PoolGenr procedure. Click on PoolGenr on the pool toolbar and enter your formula, e.g. REALGDP? = GDP?/DEFL?.

Estimation in the pool object allows the following options (always use the pool series names, e.g. GDP?):

- There are two types of explanatory variables: Common coefficients are for pool series that are to have the same coefficient across all cross-section members of the pool; Cross-section specific coefficients are for pool series with different coefficients for each member of the pool.
- You can choose between no intercepts, common identical intercept for all pool members, fixed effects for different intercepts for each cross-section unit and random effects for a cross-section specific part of the error term.
- There are three estimation methods: No weighting uses OLS; Cross section weights uses FGLS with estimated cross-section residual variances; SUR is an analogue to Seemingly Unrelated Regression and uses FGLS with estimated cross-section residual covariance matrix. Models with random effects model are always estimated using a FGLS procedure.

7 Some EViews Commands

Data Handling

open (open a file)

open file_name

series (declare a series object)

series ser_name

series ser_name = expression

genr (generate series)

genr ser_name = expression

smpl (set sample range)

smpl start1 end1 start2 end2 ... if_condition

sort (sort the workfile)

sort(d) ser1 ser2 ...

d Sort in descending order

group (declare a group of series)

group group_name ser1 ser2 ser3 ...

na (not available code)

rnd (uniform random number generator)

Generates (pseudo) random draws from a uniform distribution on (0,1).

nrnd (normal random number generator)

Generates (pseudo) random draws from a standard normal distribution.

Descriptive Statistics

stats (descriptive statistics)

stats ser1 ser2 ser3 ...

hist (histogram and descriptive statistics of a series)

hist series_name

cov (covariance matrix)

cov ser1 ser2 ser3 ...

cor (correlation matrix)

cor ser1 ser2 ser3 ...

Graphs

line, plot (line graph)

line ser_name

plot ser_name

bar (bar graph of series or each column of a vector/matrix)

bar ser1 ser2 ser3 ...

scat (scatter diagram)

scat(r,m) ser1 ser2 ser3 ...

r draw a linear regression line for each series on the vertical axis

m make multiple scatter plots for more than two series

Estimation

ls (estimation by linear or nonlinear least squares regression)

ls(h,n) y x1 x2 x3

ls(h,n) y=c(1)*x1+c(2)*x2+c(3)*x3

h White's heteroskedasticity consistent standard errors.

n Newey-West heteroskedasticity and autocorrelation consistent (HAC) standard errors.

probit (estimation of binary dependent variable models with normal errors)

probit(q,r,b) y x1 x2 x3

q Use quadratic hill climbing for maximization algorithm.

r Use Newton-Raphson for maximization algorithm.

b Use Berndt-Hall-Hall-Hausman (BHHH) for maximization algorithm.

logit (estimate binary models with logistic errors)

logit(q,r,b) y x1 x2 x3

censored (estimation of censored and truncated models)

censored(l=number,r=number,q,r,b) y x1 x2 x3

l=number Set value for the left censoring limit.

r=number Set value for the right censoring limit.

Tests

wald (Wald coefficient restriction test)

wald restrictions

restrictions e.g.: c(2)=0, c(3)=c(4)

chow (Chow test for stability)

chow(options) obs1 obs2 obs3

obs1,... breakpoints (dates or observation numbers) to be tested

white (White's test for heteroskedasticity)

white(c)

c Include all possible nonredundant cross-product terms in the test

uroot (unit root tests)

uroot(lags,c,t,n,) series_name

lags specify the number of lags

c include a constant in the test equation.

t include a constant and a linear time trend in the test equation.

n do not include a constant or time trend.

8 Important Operators

+	add	$x+y$ adds the contents of X and Y
-	subtract	$x-y$ subtracts the contents of Y from X
*	multiply	$x*y$ multiplies the contents of X by Y
/	divide	x/y divides the contents of X by Y
^	raise to the power	x^y raises X to the power of Y
>	greater than	$x>y$ takes the value 1 if X exceeds Y, and 0 otherwise
<	less than	$x<y$ takes the value 1 if Y exceeds X, and 0 otherwise
=	equal to	$x=y$ takes the value 1 if X and Y are equal, and 0 otherwise
<>	not equal to	$x<>y$ takes the value 1 if X and Y are not equal, and 0 if they are equal
<=	less than or equal to	$x<=y$ takes the value 1 if X does not exceed Y, and 0 otherwise
>=	greater than or equal to	$x>=y$ takes the value 1 if Y does not exceed X, and 0 otherwise
and	logical and	x and y takes the value 1 if both X and Y are nonzero, and 0 otherwise
or	logical or	x or y takes the value 1 if either X or Y is nonzero, and 0 otherwise
log(x)	natural logarithm	ln(X)
sqr(x)	square root	$\text{sqr}(4)=2$
abs(x)	absolute value	$\text{abs}(-3)=3$
exp(x)	exponential	$\text{exp}(1)=2.71813$
@cos(x)	cosine	argument in radians: $\text{@cos}(\pi)=-1$
@sin(x)	sine	argument in radians: $\text{@sin}(\pi)=0$
@tan(x)	tangent	argument in radians: $\text{@tan}(1)=1.5574$
d(x)	first difference	$(1-L)X_t = X_t - X_{t-1}$
d(x,n)	n-th difference	$(1-L)^n X_t$
@trend	time trend	0, 1, 2, 3, 4, ..., T-2, T-1